R&S®RTE
Oscilloscope
User Manual

1326103202
Version 19

ROHDE&SCHWARZ
Make ideas real

This manual describes the following R&S®RTE models with firmware version 4.80:

- R&S®RTE1022: 200 MHz, 2 channels (1326.2000K22 and 1317.2500K22)
- R&S®RTE1024: 200 MHz, 4 channels (1326.2000K24 and 1317.2500K24)
- R&S®RTE1032: 350 MHz, 2 channels (1326.2000K32 and 1317.2500K32)
- R&S®RTE1034: 350 MHz, 4 channels (1326.2000K34 and 1317.2500K34)
- R&S®RTE1052: 500 MHz, 2 channels (1326.2000K52 and 1317.2500K52)
- R&S®RTE1054: 500 MHz, 4 channels (1326.2000K54 and 1317.2500K54)
- R&S®RTE1102: 1 GHz, 2 channels (1326.2000K62 and 1317.2500K02)
- R&S®RTE1104: 1 GHz, 4 channels (1326.2000K64 and 1317.2500K04)
- R&S®RTE1152: 1,5 GHz, 2 channels (1326.2000K72)
- R&S®RTE1154: 1,5 GHz, 4 channels (1326.2000K74)
- R&S®RTE1202: 2 GHz, 2 channels (1326.2000K82)
- R&S®RTE1204: 2 GHz, 4 channels (1326.2000K84)

Mühldorfstr. 15, 81671 München, Germany
Phone: +49 89 41 29 - 0
Email: info@rohde-schwarz.com
Internet: www.rohde-schwarz.com

1326.1032.02 | Version 19 | R&S®RTE

Throughout this manual, products from Rohde & Schwarz are indicated without the ® symbol , e.g. R&S®RTE is indicated as R&S RTE.

Contents

1 Preface....................9

1.1 Safety and Regulatory Information....................9

1.2 Key Features....................10

1.3 Documentation Overview....................11

1.4 Options Described in this Document....................12

1.5 Conventions Used in the Documentation....................13

2 Getting Started....................15

2.1 Preparing for Use....................15

2.2 Instrument Tour....................21

2.3 Trying Out the Instrument....................35

2.4 Operating the Instrument....................67

3 Instrument Setup....................94

3.1 System Setup....................94

3.2 Screen Setup....................99

3.3 Frontpanel Setup....................101

3.4 Display Configuration....................104

3.5 External Application....................120

3.6 Self-alignment....................120

3.7 Self-test....................122

3.8 Firmware Update....................123

3.9 Options....................123

4 Acquisition and Waveform Setup....................128

4.1 Basics....................128

4.2 Horizontal Settings....................138

4.3 Vertical Settings....................147

4.4 High Definition Mode....................151

4.5 Probes....................155

4.6 R&S RT-ZVC Probe....................179

4.7 Differential Signals....................189

4.8 Digital Filter Setup....................192

4.9 Horizontal Accuracy.. 193
4.10 Setting Up the Waveform...196

5 Triggers..200
5.1 Basics of Triggering...200
5.2 Setting Up the Trigger...201
5.3 Trigger Types...203
5.4 Holdoff..226
5.5 Noise Reject...227
5.6 Control / Action... 229
5.7 Sequence... 232
5.8 External Trigger Input... 235
5.9 Acquisition Info... 239

6 Waveform Analysis... 240
6.1 Zoom... 240
6.2 Reference Waveforms..252
6.3 Mathematics.. 257
6.4 History..273
6.5 XY-Diagram..280

7 Measurements...284
7.1 Cursor Measurements... 284
7.2 Automatic Measurements.. 294
7.3 Quick Measurements... 356

8 Spectrum Analysis...359
8.1 FFT Analysis...359
8.2 Spectrum Analysis (Option R&S RTE-K18).. 378

9 Mask Testing...383
9.1 About Mask Testing... 383
9.2 Mask Test Settings...385
9.3 Working with Masks...397

10 Search Functions...406
10.1 Overview: Search Definition and Results...406

10.2 Search Setup... 408

10.3 Search Gate... 422

10.4 Result Presentation.. 424

10.5 Noise Reject...428

11 Data and File Management..430

11.1 Instrument Settings.. 430

11.2 Waveform Data and Results..438

11.3 Autonaming... 459

11.4 Screenshots...461

11.5 Reports..467

11.6 Preset Setup.. 469

11.7 File Selection Dialog..471

12 Protocol Analysis..473

12.1 Basics of Protocol Analysis...473

12.2 I²C (Option R&S RTE-K1)..482

12.3 SPI Bus (Option R&S RTE-K1)... 501

12.4 UART/RS-232/RS-422/RS-485 (Option R&S RTE-K2)..513

12.5 CAN and CAN FD (Options R&S RTE-K3 and -K9)..523

12.6 LIN (Option R&S RTE-K3)...561

12.7 FlexRay (Option R&S RTE-K4)...579

12.8 Audio Signals (Option R&S RTE-K5)...598

12.9 MIL-1553 (Option R&S RTE-K6)... 620

12.10 ARINC 429 (Option R&S RTE-K7).. 641

12.11 Ethernet 10BASE-T and 100BASE-TX (Option R&S RTE-K8)............................ 656

12.12 Ethernet 100BASE-T1 (Option R&S RTE-K57)...675

12.13 SENT (Option R&S RTE-K10)... 696

12.14 Custom: Manchester / NRZ (Option R&S RTE-K50)..733

12.15 MDIO (Option R&S RTE-K55)... 768

12.16 USB (Option R&S RTE-K60)... 783

12.17 USBPD (Option R&S RTE-K63)...815

12.18 SpaceWire (Option R&S RTE-K65).. 831

12.19 CXPI (Option R&S RTE-K76).. 846

13 Mixed Signal Option (MSO, R&S RTE-B1)....................................... 867
13.1 Digital Channels and Parallel Buses.. 867
13.2 Display.. 876
13.3 Trigger.. 878
13.4 Measurements on Digital Channels... 890
13.5 Data Export... 890
13.6 Mathematics... 891
13.7 Search.. 892

14 Waveform Generator (Option R&S RTE-B6).................................... 893
14.1 Setup of the Waveform Generator... 893
14.2 Setup of the Pattern Generator.. 912
14.3 Coupling and Sync Settings.. 916
14.4 Configuring the Waveform Generator... 917
14.5 DC Offset Alignment... 921

15 Power Analysis (Option R&S RTE-K31).. 922
15.1 Power Measurement Selection... 922
15.2 Overview of Power Measurement Setup... 933
15.3 Power Quality... 936
15.4 Inrush Current.. 941
15.5 Current Harmonic... 944
15.6 Modulation Analysis.. 949
15.7 Dynamic on Resistance.. 953
15.8 Slew Rate.. 957
15.9 Safe Operating Area (S.O.A.).. 961
15.10 Turn On/Off... 965
15.11 Switching Loss... 970
15.12 Power Efficiency... 974
15.13 Output Ripple.. 977
15.14 Transient Response 983
15.15 Output Spectrum.. 987

16 Network and Remote Operation... 991
16.1 Operating System.. 991

16.2 Setting Up a Network (LAN) Connection.....996
16.3 Web Interface.....1001
16.4 Remote Desktop Connection.....1007
16.5 Remote Control Interfaces and Protocols.....1009
16.6 Remote Settings.....1013
16.7 Starting and Stopping Remote Control.....1015

17 Remote Control Commands.....1016
17.1 Conventions used in Remote Command Description.....1016
17.2 Finding the Appropriate Command.....1017
17.3 Programming Examples.....1018
17.4 Frequently Used Parameters and Suffixes.....1035
17.5 Common Commands.....1039
17.6 General Remote Settings.....1044
17.7 Instrument Setup.....1048
17.8 Acquisition and Setup.....1069
17.9 Trigger.....1127
17.10 Waveform Analysis.....1173
17.11 Cursor Measurements.....1201
17.12 Automatic Measurements.....1212
17.13 Spectrum Analysis.....1278
17.14 Mask Testing.....1294
17.15 Search.....1314
17.16 Data Management.....1352
17.17 Protocols.....1381
17.18 Mixed Signal Option (MSO, R&S RTE-B1).....1863
17.19 Waveform Generator (Option R&S RTE-B6).....1886
17.20 Power Analysis (Option R&S RTE-K31).....1906
17.21 Maintenance.....1945
17.22 Status Reporting.....1948
17.23 Remote Trace.....1952
17.24 Deprecated Commands.....1956

18 Maintenance and Support.....1959
18.1 Cleaning.....1959

18.2 Contacting Customer Support....1959
18.3 Information for Technical Support....1960
18.4 Data Security....1960
18.5 Storing and Packing....1961
18.6 Maintenance Information....1961

Annex....1963

A Menu Overview....1963
A.1 File Menu....1963
A.2 Horizontal Menu....1964
A.3 Trigger Menu....1964
A.4 Vertical Menu....1964
A.5 Math Menu....1965
A.6 Cursor Menu....1965
A.7 Meas Menu....1966
A.8 Masks Menu....1966
A.9 Analysis Menu....1966
A.10 Display Menu....1967
A.11 WaveGen Menu (with Option R&S RTE-B6)....1968

B Remote Control - Basics....1969
B.1 Messages....1969
B.2 SCPI Command Structure....1971
B.3 Command Sequence and Synchronization....1980
B.4 General Programming Recommendations....1982

C Remote Control - Status Reporting System....1983
C.1 Structure of a SCPI Status Register....1983
C.2 Hierarchy of Status Registers....1984
C.3 Contents of the Status Registers....1986
C.4 Application of the Status Reporting System....1994
C.5 Reset Values of the Status Reporting System....1996

List of commands....1998

Index....2057

1 Preface

1.1 Safety and Regulatory Information

The R&S RTE oscilloscope is designed for measurements on circuits that are only indirectly connected to the mains or not connected at all. It is not rated for any measurement category.

The instrument is rated for pollution degree 2 - for indoor, dry location use where only non-conductive pollution occurs. Temporary conductivity caused by condensation is possible.

The instrument is intended for use in industrial areas. When used in residential areas, radio disturbances caused by the instrument can exceed given limits. Additional shielding can be required.

The instrument must be controlled by personnel familiar with the potential risks of measuring electrical quantities. Observe applicable local or national safety regulations and rules for the prevention of accidents.

Safety information is part of the product documentation. It warns you about the potential dangers and gives instructions how to prevent personal injury or damage caused by dangerous situations. Safety information is provided as follows:

- The "Basic Safety Instructions" in different languages are delivered as a printed brochure with the instrument.
- Throughout the documentation, safety instructions are provided when you need to take care during setup or operation.

Safety instructions

To prevent electric shock, personal injury or fire, follow these rules:

- Do not open the instrument casing.
- Do not use the instrument if you detect or suspect any damage of the instrument or accessories.
- Do not operate the instrument in wet, damp or explosive atmospheres.
- Make sure that the instrument is properly grounded.
- Do not use the instrument to ascertain volt-free state.
- Do not exceed the voltage limits given in Chapter 2.2.1.1, "Input Connectors", on page 23.

1.1.1 Korea Certification Class A

이 기기는 업무용(A급) 전자파 적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.

1.2 Key Features

The R&S RTE oscilloscope provides fast signal acquisition and analysis:

- 1 million waveforms per second waveform acquisition rate
- Bandwidth up to 2 GHz, memory depth of 50 MSa per channel
- Highly accurate digital trigger system
- Very low noise floor
- Precise measurements due to single-core A/D converter
- High measurement speed, even for complex analysis functions
- High-quality line of probes

For a detailed specification refer to the data sheet.

The R&S RTE oscilloscope brings various benefits in your daily work:

- Find rare signal faults quickly with no trade-offs for measurement and analysis due to highest acquisition rate and shortest blind time.
- Access and analyze prior waveforms in the memory using the history function
- Get fastest results even with maximum data with hardware-accelerated processing: mathematical operations, mask tests, histograms, automatic and cursor measurements, and spectrum display.
- Capture closest successive events with the real-time digital trigger system. It works with high trigger sensitivity at full bandwidth and very low trigger jitter.
- See signal details at your fingertip with fingertip zoom
- Get key measurement results at the push of a button with Quick Measurement
- Easy to use:
  – Smart and straightforward user guidance
  – Color-coded control elements for clear identification
  – Signal icons with drag & drop functionality
  – Toolbar with frequently used functionality
- Use various options for triggering and decoding of serial protocols, MSO option, high resolution, and more.

1.3 Documentation Overview

This section provides an overview of the R&S RTE user documentation.

1.3.1 Manuals and Instrument Help

You find the manuals on the product page at:

www.rohde-schwarz.com/manual/rte

Getting started manual

Introduces the R&S RTE and describes how to set up and start working with the instrument, and describes basic operations. A printed English version is included in the delivery. Editions in other languages are available on the product website.

Instrument help

The help offers quick, context-sensitive access to the complete information for the firmware basic functionality and applications.

User manual

Describes all instrument functions in detail. It also provides an introduction to remote control, a complete description of the remote control commands with programming examples, and information on maintenance and instrument interfaces. Includes the contents of the getting started manual.

The *online version* of the user manual provides the complete contents for immediate display on the internet.

Basic safety instructions

Contains safety instructions, operating conditions and further important information. The printed document is delivered with the instrument.

Instrument security procedures manual

Deals with security issues when working with the R&S RTE in secure areas.

Service Manual

Describes the performance test for checking the rated specifications, module replacement, firmware update, troubleshooting and fault elimination, and contains mechanical drawings and spare part lists. The service manual is available for registered users on the global Rohde & Schwarz information system (GLORIS, https://gloris.rohde-schwarz.com).

1.3.2 Data Sheet and Brochure

The data sheet contains the technical specifications of the R&S RTE. It also lists the options with their order numbers and optional accessories. The brochure provides an overview of the instrument and deals with the specific characteristics.

See www.rohde-schwarz.com/brochure-datasheet/rte

1.3.3 Release Notes, Open Source Acknowledgment

The release notes list new features, improvements and known issues of the current firmware version, and describe the firmware installation. The open source acknowledgment document provides verbatim license texts of the used open source software. It can also be read directly on the instrument.

See www.rohde-schwarz.com/firmware/rte.

1.3.4 Application notes, Application cards, Videos

These documents deal with special applications or background information on particular topics.

See www.rohde-schwarz.com/application/rte and Oscilloscopes Application Videos - Media Center

1.4 Options Described in this Document

In addition to the base unit, the following options are described in this documentation:

Type	Designation	Order No. for 1326.2000.xx / 1317.2500.xx instruments
R&S RTE-B1	MSO	1317.4961.02
R&S RTE-B6	Waveform and pattern generator	1326.3012.02 / n.a.
R&S RTE-K1	I²C and SPI serial triggering and decoding	1326.1178.02 / 1317.7125.02
R&S RTE-K2	UART/RS-232/RS-422/RS-485 serial triggering and decoding	1326.1184.02 / 1317.7131.02
R&S RTE-K3	CAN and LIN serial triggering and decoding	1326.1190.02 / 1317.7148.02
R&S RTE-K4	FlexRay™ serial triggering and decoding	1326.1203.02 / 1317.7154.02
R&S RTE-K5	I²S (audio) serial triggering and decoding	1326.1210.02 / 1317.7160.02

Type	Designation	Order No. for 1326.2000.xx / 1317.2500.xx instruments
R&S RTE-K6	MIL-STD-1553 serial triggering and decoding	1326.1226.02 / 1325.9781.02
R&S RTE-K7	ARINC 429 serial triggering and decoding	1326.1232.02 / 1325.9798.02
R&S RTE-K8	Ethernet serial triggering and decoding	1326.1332.02 / 1317.7402.02
R&S RTE-K9	CAN-FD serial triggering and decoding	1326.1249.02 / 1325.9898.02
R&S RTE-K10	SENT serial triggering and decoding	1326.1603.02 / 1326.1532.02
R&S RTE-K18	Spectrum Analysis	1326.3006.02 / 1326.3035.02
R&S RTE-K31	Power analysis	1326.1278.02 / 1317.7177.02
R&S RTE-K35	Bus analysis for I2C, SPI, RS232/UART, CAN, CAN-FD, LIN, SENT, 100BASE-Tx and 100BASE-T1. Requires corresponding protocol decoding option.	1801.2852.02
R&S RTE-K50	Custom Manchester and NRZ serial triggering and decoding	1326.1326.02 / 1326.1310.02
R&S RTE-K55	MDIO serial triggering and decoding	1326.1255.02 / 1326.0720.02
R&S RTE-K57	100BASE-T1 serial triggering and decoding	1333.0615.02
R&S RTE-K60	USB 1.0/1.1/2.0/HSIC serial triggering and decoding	1326.1626.02 / 1326.1610.02
R&S RTE-K63	USB-PD serial triggering and decoding	1326.3158.02 / 1326.3141.02
R&S RTE-K65	SpaceWire serial triggering and decoding	1326.2845.02 / 1326.2839.02
R&S RTE-K76	CXPI serial triggering and decoding	1326.3193.02 / 1326.3187.02

1.5 Conventions Used in the Documentation

1.5.1 Typographical Conventions

The following text markers are used throughout this documentation:

Convention	Description
"Graphical user interface elements"	All names of graphical user interface elements on the screen, such as dialog boxes, menus, options, buttons, and softkeys are enclosed by quotation marks.
[Keys]	Key and knob names are enclosed by square brackets.
`Filenames, commands, program code`	Filenames, commands, coding samples and screen output are distinguished by their font.
Input	Input to be entered by the user is displayed in italics.
Links	Links that you can click are displayed in blue font.
"References"	References to other parts of the documentation are enclosed by quotation marks.

1.5.2 Conventions for Procedure Descriptions

When operating the instrument, several alternative methods may be available to perform the same task. In this case, the procedure using the touchscreen is described. Any elements that can be activated by touching can also be clicked using an additionally connected mouse. The alternative procedure using the keys on the instrument or the on-screen keyboard is only described if it deviates from the standard operating procedures.

The term "select" may refer to any of the described methods, i.e. using a finger on the touchscreen, a mouse pointer in the display, or a key on the instrument or on a keyboard.

2 Getting Started

Note: the following chapters are identical to those in the R&S RTE Getting Started manual.

- Preparing for Use..15
- Instrument Tour..21
- Trying Out the Instrument...35
- Operating the Instrument..67

2.1 Preparing for Use

This section describes the basic steps to be taken when setting up the R&S RTE for the first time.

NOTICE

Risk of instrument damage due to inappropriate operating conditions

An unsuitable operating site or test setup can damage the instrument and connected devices. Before switching on the instrument, observe the information on appropriate operating conditions provided in the data sheet. In particular, ensure the following:

- All fan openings are unobstructed and the airflow perforations are unimpeded. A minimum distance of 10 cm to other objects is recommended.
- The instrument is dry and shows no sign of condensation.
- The instrument is positioned as described in the following sections.
- The ambient temperature does not exceed the range specified in the data sheet.
- Signal levels at the input connectors are all within the specified ranges.
- Signal outputs are connected correctly and are not overloaded.

2.1.1 Unpacking and Checking the Instrument

To remove the instrument from its packaging and check the equipment for completeness, proceed as follows:

1. Pull off the polyethylene protection pads from the instrument's rear feet.
2. Carefully remove the pads from the instrument handles at the front.
3. Pull off the corrugated cardboard cover that protects the rear of the instrument.
4. Carefully unthread the corrugated cardboard cover at the front that protects the instrument handles and remove it.

5. Check the equipment for completeness using the delivery note and the accessory lists for the various items.
6. Check the instrument for any damage. If there is damage, immediately contact the carrier who delivered the instrument. Make sure not to discard the box and packing material.

Packing material

Retain the original packing material. If the instrument needs to be transported or shipped later, you can use the material to protect the control elements and connectors.

2.1.2 Positioning the Instrument

The instrument is designed for use under laboratory conditions. It can be used in standalone operation on a bench top or can be installed in a rack.

2.1.2.1 Bench Top Operation

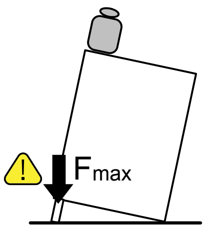
For standalone operation, place the instrument on a horizontal bench with even, flat surface. The instrument can be used in horizontal position, standing on its feet, or with the support feet on the bottom extended.

CAUTION

Risk of injury if feet are folded out

The feet can fold in if they are not folded out completely or if the instrument is shifted. This can cause damage or injury.

- Fold the feet completely in or out to ensure stability of the instrument. Never shift the instrument when the feet are folded out.
- When the feet are folded out, do not work under the instrument or place anything underneath.
- The feet can break if they are overloaded. The overall load on the folded-out feet must not exceed 200 N.

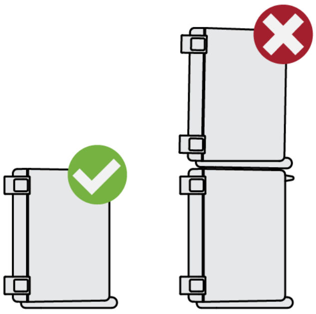


CAUTION

Risk of injury when stacking instruments insecurely

Never stack instruments on top of each other. The instrument's top surface area is too small to stack multiple instruments. Stacked instruments can tilt over and cause injury or damage the instrument.

If you need to stack instruments, install them in a rack.



2.1.2.2 Rackmounting

The instrument can be installed in a rack using a rack adapter kit. The order number is given in the data sheet. The installation instructions are part of the adapter kit.

NOTICE

Risk of instrument damage due to insufficient airflow in a rack

If you mount several instruments in a rack, you need an efficient ventilation concept to ensure that the instruments do not overheat. Insufficient airflow for a longer period can disturb the operation and even cause damage.

2.1.3 Starting the Instrument

CAUTION

Risk of injury

Connect the oscilloscope to an outlet that has a ground contact.

If grounding is *not* ensured by the mains system, ground the oscilloscope using the protective earth conductor on the front panel and an appropriate cable.

Do not use an isolating transformer to connect the oscilloscope to the AC power supply.

2.1.3.1 Powering On and Off

The R&S RTE can be used with different AC power voltages and adapts itself automatically to it.

To power on

1. Connect the power cable to the AC power connector on the rear panel of the R&S RTE.
2. Connect the power cable to the socket outlet.
3. Switch the main power switch at the rear of the instrument to position I.

   The power key on the front panel lights up.

Before you start measurements, be sure to comply with the warm-up phase specified in the data sheet.

If you leave the main power switch on, the instrument is in standby mode. To disconnect from power supply, power off the instrument. Powering off is only required if the instrument must be disconnected from all power supplies.

To power off

1. If the instrument is running and the [POWER] key is green, press the [POWER] key on the front panel to shut down the instrument.
2. Switch the main power switch at the rear of the instrument to position 0.
3. Disconnect the AC power cable from the AC power supply.

NOTICE

Risk of losing data

If you switch off the running instrument using the rear panel switch or by disconnecting the power cord, the instrument loses its current settings. Furthermore, program data can be lost.

Press the POWER key first to shut down the application properly.

2.1.3.2 Starting Up and Shutting Down

The [POWER] key is located in the bottom left corner of the front panel.

To start up the instrument

1. Make sure that the R&S RTE is connected to the AC power supply and the main power switch on the rear panel is in position I.
2. Press the [POWER] key on the front panel.

   The instrument performs a system check, boots the Windows operating system, and then starts the R&S RTE firmware.

The [POWER] key turns green and the illuminated keys on the front panel light up. If the previous session was terminated regularly, the oscilloscope uses the last settings.

To shut down the instrument

► Press the [POWER] key on the front panel.

All current settings are saved, and the software shuts down. The [POWER] key turns orange. The standby power supplies only the power switch circuits.
Now it is safe to power off the instrument.

The "Exit" function in the "File" menu shuts down only the firmware application. To shut down the instrument completely, also shut down the operating system in the "Start" menu, or use the [POWER] key.

2.1.3.3 EMI Suppression

Electromagnetic Interference (EMI) may affect the measurement results.

To suppress generated Electromagnetic Interference:
- Use suitable shielded cables of high quality. For example use double-shielded RF and LAN cables.
- Always terminate open cable ends.
- Note the EMC classification in the data sheet.

2.1.4 Connecting External Devices

The following interfaces for external devices are provided:

- USB connectors at the front and rear panel of the instrument
- Monitor connector DVI-D at the rear panel of the instrument

- Connecting USB Devices....................19
- Connecting an External Monitor....................21

2.1.4.1 Connecting USB Devices

The USB interfaces on the front and rear panels allow you to connect USB devices directly to the instrument. The number of USB connectors can be increased by using USB hubs. Due to the large number of available USB devices, there is almost no limit to the expansions that are possible with the R&S RTE.

The following USB devices can be useful, for example:
- USB flash drives to save screenshots and measurement results, and for easy installation of firmware applications
- Keyboard and/or mouse to simplify the operation and the entry of data, comments, filenames, etc.
- Printer to print measurement results and screenshots

You can connect or disconnect all USB devices during operation of the instrument.

Installing USB devices on R&S RTE is easy under the Windows operating system, because all USB devices are plug&play. After a device is connected to the USB interface, Windows automatically searches for a suitable device driver.

If the operating system does not find a suitable driver, it prompts you to specify a directory that contains the driver software. If the driver software is on a storage media, connect the appropriate drive to the instrument before proceeding. If the instrument is integrated in a network, you can also install driver data stored in a network directory.

When a USB device is disconnected from the R&S RTE, Windows immediately detects the change in hardware configuration and deactivates the corresponding driver.

The properties of external USB devices are configured in the operating system, not in the R&S RTE software. It is recommended that you use mouse and keyboard to access and modify the settings of the Windows operating system. To access Windows, press the Windows key on the external keyboard, or select "File" > "Minimize Application" on the R&S RTE menu.

Connecting a USB flash drive

If the installation of a USB flash drive is successful, Windows informs you that the device is ready to use. The device is made available as a new drive ("D:") and is displayed in Windows Explorer. The name of the drive depends on the manufacturer.

Connecting a keyboard

The keyboard is detected automatically when it is connected. The default input language is English – US.

To configure the keyboard properties:

1. Tap the "Find" icon (magnifier) on the Windows taskbar.
2. Type *keybord*.
3. Select "Edit language and keyboard options".

Connecting a mouse

The mouse is detected automatically when it is connected. To configure the mouse properties:

1. Tap the "Find" icon (magnifier) on the Windows taskbar.
2. Type *mouse*.
3. Select "Mouse settings".

Connecting a printer

When printing a file, the instrument checks whether a printer is connected and turned on, and whether the appropriate printer driver is installed. If necessary, printer driver installation is initiated by the Windows system. To install a printer driver:

1. Tap the "Find" icon (magnifier) on the Windows taskbar.

2. Type *printer*.
3. Select "Printers & scanners".
4. Select "Add a printer or scanner".

2.1.4.2 Connecting an External Monitor

You can connect an external monitor or projector to the R&S RTE. The following connectors are available:

- "MONITOR (DVI-D)" on page 26

Before connecting an external monitor, ensure that the monitor and the R&S RTE are connected to a ground contact. Otherwise the instrument can be damaged.

After connecting an additional monitor or projector to the instrument, configure it for usage. The relevant settings are Windows settings but you can configure the displays directly in the instrument setup.

1. Check the input type of the monitor or projector. Make sure to select the correct cable. To use a VGA monitor, you need an active DVI-D to VGA adapter.
2. Press the [SETUP] key.
3. Select the "System" tab.
4. Tap "Display / Monitors".
5. To show the instrument's display content only on the external monitor, select "Projector only".
   To show the instrument's display content on both the oscilloscope and the external monitor, select "Duplicate".

The touchscreen of the R&S RTE has a screen resolution of 1024 x 768 pixel. Most external monitors have a higher screen resolution. If the screen resolution of the monitor is set higher than the instrument's resolution, the application window uses a 1024 x 768 area of the monitor display. For full screen display, adjust the monitor's screen resolution using "Additional display settings".

2.2 Instrument Tour

This chapter describes the front and rear panels of the instrument including all function keys and connectors, and also the touchscreen with its control elements.

2.2.1 Front Panel

The front panel of the R&S RTE is shown in Figure 2-1. The function keys are grouped in functional blocks to the left and the right of the touchscreen. Below the screen, various connectors are located.

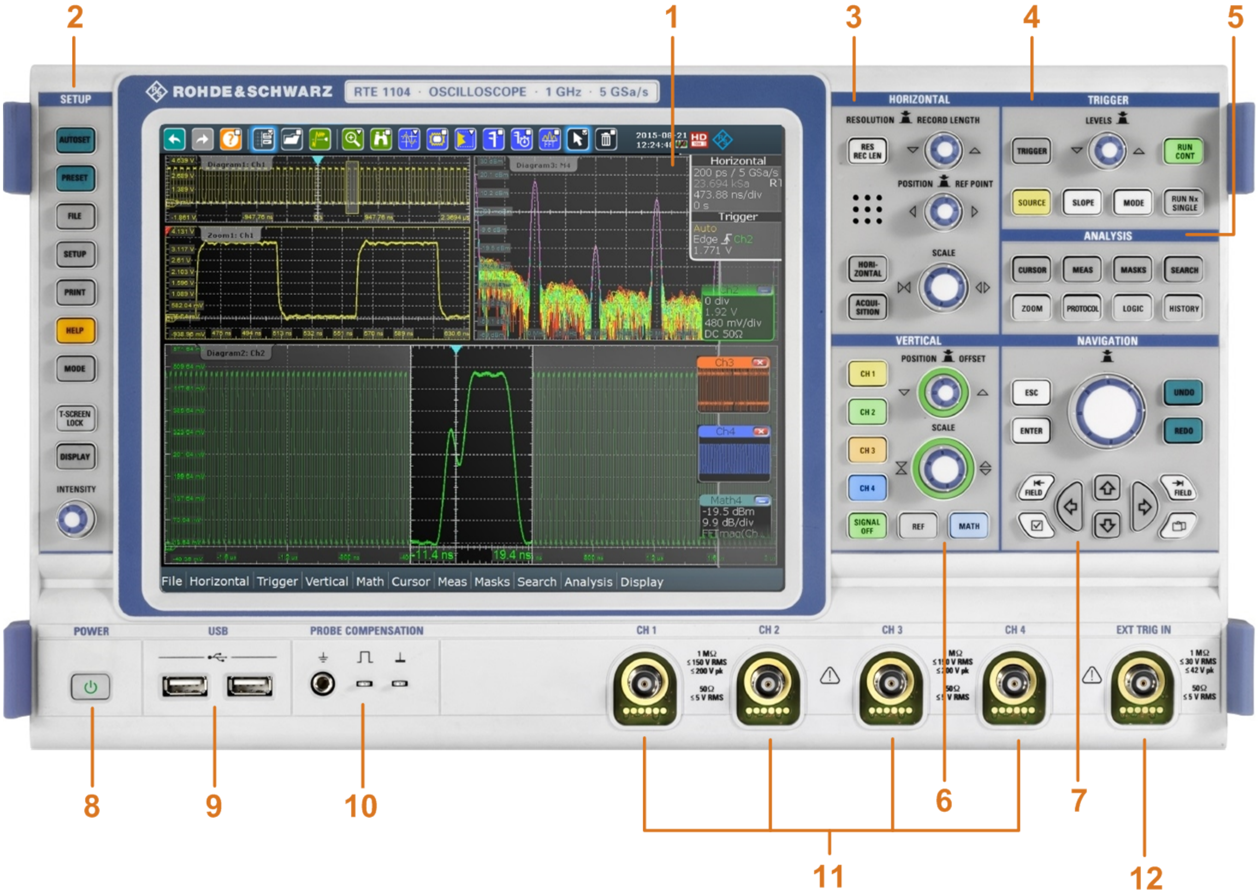


Figure 2-1: Front panel of R&S RTE1104 with 4 input channels

1 = Touchscreen
2 = [SETUP] controls
3 = [HORIZONTAL] controls
4 = [TRIGGER] controls
5 = [ANALYSIS] controls
6 = [VERTICAL] controls
7 = [NAVIGATION] controls
8 = [POWER] key
9 = USB 2.0 connectors
10 = Connectors for probe compensation
11 = Input channels
12 = External trigger input

NOTICE

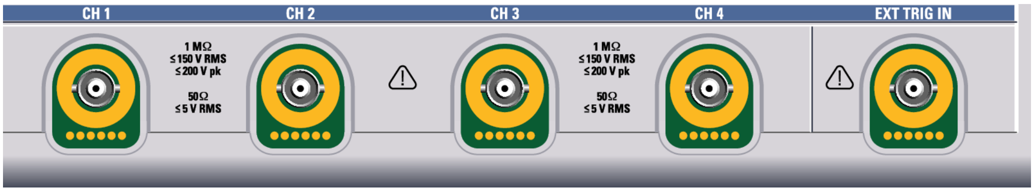
Instrument damage caused by cleaning agents

Cleaning agents contain substances such as solvents (thinners, acetone, etc.), acids, bases, or other substances. Solvents can damage the front panel labeling, plastic parts, or screens, for example.

Never use cleaning agents to clean the outside of the instrument. Use a soft, dry, lint-free dust cloth instead.

2.2.1.1 Input Connectors

The R&S RTE has two or four channel inputs to connect the input signals, and an external trigger input to control the measurement by an external signal.



The input connectors are provided with a special Rohde & Schwarz active probe interface, and they are BNC compatible. Thus, the instrument can automatically detect passive probes with standard BNC connector and active Rohde & Schwarz probes having the Rohde & Schwarz probe interface.

The input impedance is selectable, the values are 50 Ω and 1 MΩ.

WARNING

Risk of electrical shock or fire

Voltages higher than 30 V RMS or 42 V peak or 60 V DC are regarded as hazardous contact voltages. When working with hazardous contact voltages, use appropriate protective avoid electrical shock and injuries:

- Ground the instrument.
- Use only insulated probes, cables, test leads and adapters.
- Do not touch voltages higher than 30 V RMS or 42 V peak or 60 V DC.

⚠ CAUTION

Risk of injury and instrument damage

The instrument is not rated for any measurement category.

Make sure that the input voltage on *channel inputs* does not exceed 200 V peak, 150 V RMS at 1 MΩ input impedance and 5 V RMS at 50 Ω input impedance.

Transient overvoltages must not exceed 200 V peak.

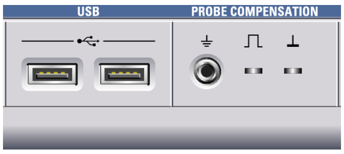
When performing measurements in circuits with transient overvoltages of category II, III or IV circuits, make sure that no such overvoltages reach the R&S RTE input. Therefore, use only probes that comply with DIN EN 61010-031. When performing measurements in category II, III or IV circuits, it is mandatory to insert a probe that appropriately reduces the voltage so that no transient overvoltages higher than 200 V peak are applied to the instrument. For detailed information, refer to the documentation and safety information of the probe manufacturer.

For the *external trigger input*, the maximum input voltage is 30 V RMS at 1 MΩ input impedance and 5 V RMS at 50 Ω input impedance. For further specifications, refer to the data sheet.

Explanation: According to section AA.2.4 of EN 61010-2-030, measuring circuits without any measurement category are intended for measurements on circuits which are not connected to the mains system.

2.2.1.2 Other Front Panel Connectors

Besides the input connectors, the instrument has USB connectors and probe compensation connectors at the front panel.



[USB]

Two USB type A connectors that comply with standard USB 2.0. They are used to connect devices like keyboard, mouse, printer and USB flash drive.

Note: Electromagnetic interference (EMI) can affect the measurement results. To avoid any impact, do not use USB connecting cables exceeding 1 m.

PROBE COMPENSATION

Probe compensation terminal to support adjustment of passive probes to the oscilloscope channel.

⏚	Protective earth conductor for grounding the instrument.
⎍	Square wave signal for probe compensation, 1 kHz and 1 V_{pp}.
⊥	Ground connector for probes.

2.2.2 Rear Panel

Figure 2-2 shows the rear panel of the R&S RTE with its connectors.

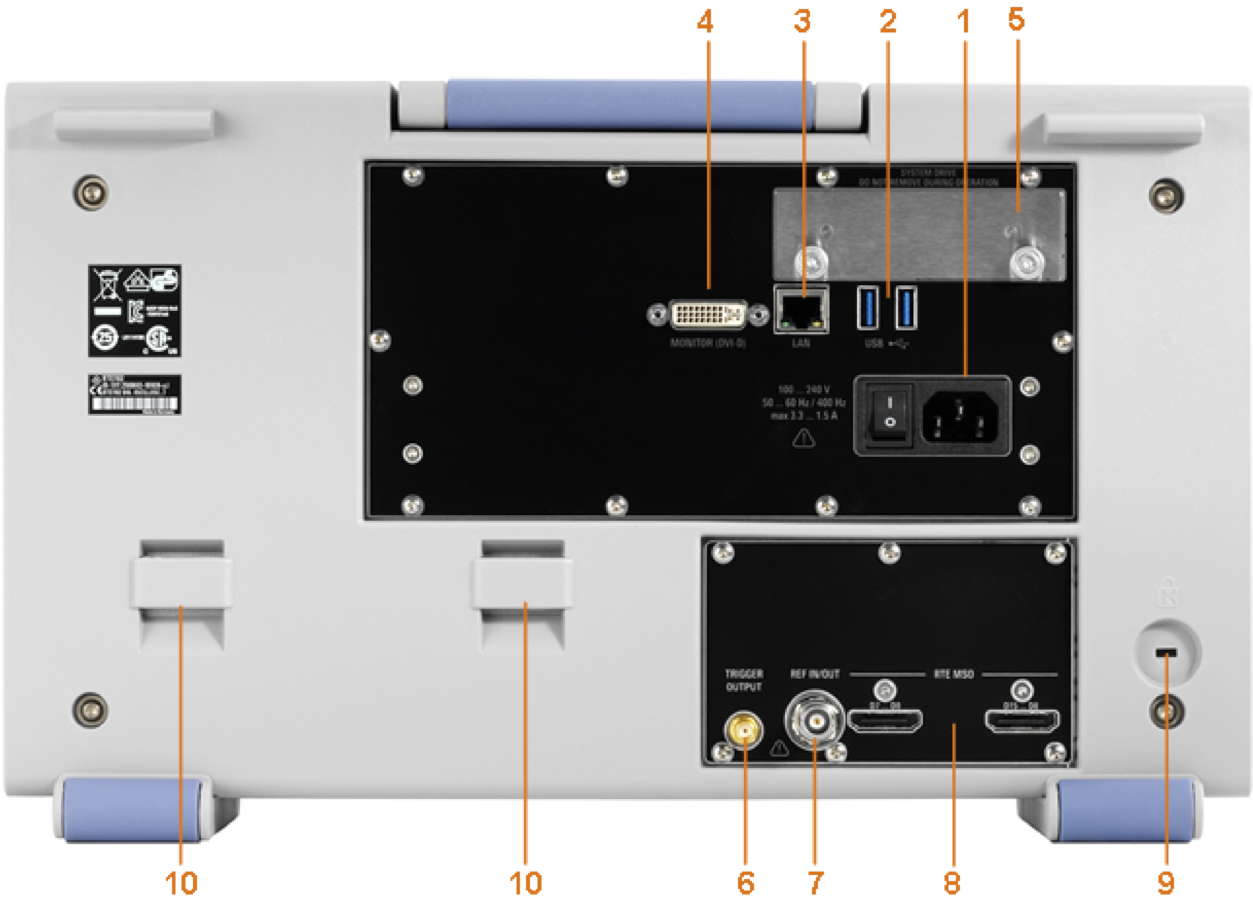


Figure 2-2: Rear panel view of R&S RTE

1 = AC power supply connector and main power switch
2 = USB connectors
3 = LAN connector
4 = DVI-D connector for external monitor
5 = Optional exchangeable solid state disk (SSD), option R&S RTE-B18
6 = External trigger output
7 = Reference input/output
8 = Slot for hardware option: mixed signal option R&S RTE-B1 (shown in picture), or waveform generator R&S RTE-B6, or GPIB connector R&S RTE-B10)
9 = Kensington lock slot to secure the instrument against theft
10 = Lugs to attach the accessory bag

AC power supply connector and main power switch

Connection to the AC power line. The R&S RTE can be used with different AC power voltages and adapts itself automatically to it. The nominal voltage and frequency ranges are displayed on the rear panel and quoted in the data sheet.

If grounding is *not* ensured by the mains system, ground the oscilloscope using the protective earth conductor on the front panel and an appropriate cable.

When you power up the instrument, be sure to comply with the warm-up phase specified in the data sheet before you start measurements.

See also: Chapter 2.1.3, "Starting the Instrument", on page 17

USB
Two USB type A connectors that comply with standard USB 3.0. They are used to connect devices like keyboard, mouse, printer, and flash drive to store and reload instrument settings and measurement data.

Note: Electromagnetic interference (EMI) can affect the measurement results. To avoid any impact, do not use USB connecting cables exceeding 1 m.

LAN
8-pin RJ-45 connector used to connect the instrument to a Local Area Network (LAN). It supports up to 1000 Mbit/s (10/100/1000BASE-T Ethernet).

MONITOR (DVI-D)
Digital connector for an external monitor or projector. The monitor shows the complete content of the instrument's screen.

See also: Chapter 2.1.4.2, "Connecting an External Monitor", on page 21.

TRIGGER OUTPUT
The SMA connector for external trigger output is used to provide the internal trigger signal of the oscilloscope to trigger other instruments for synchronized measurements.

When a trigger occurs, the R&S RTE creates a pulse of 5 V with a source impedance of 50 Ω and delivers it to the external trigger output. The instrument can also send the pulse on mask test violation or violation of measurement limits and margins.

If the connector is terminated with 50 Ω, the signal level is 2.5 V (50 mA). With 1 MΩ termination, the level is 5 V. A short-circuit of the connector to ground creates current of 100 mA.

To enable the trigger out signal, select "Trigger" menu > "Ctrl/Action". Here you also adjust polarity, delay, and length of the pulse. The default is a positive pulse of 100 ns. The minimum delay is 800 ns.

REF IN/OUT
BNC female connector for input or output of reference signals. The input or output direction is set in "Horizontal" menu > "Reference clock".

The input frequency range is 10 MHz. The input impedance is 50 Ω.

The nominal output frequency is 10 MHz, the impedance is 50 Ω. For detailed specifications, refer to the data sheet.

Mixed signal option R&S RTE-B1 / digital extension port R&S RTE-B1E
The mixed signal option R&S RTE-B1 provides logic analizer functionality and 16 digital channels. It also can connect the R&S RT-ZVC multi-channel power probe. The connectors can be used to connect two logical probes with 8 digital channels each (D0 to D7 and D8 to D15), or two flat interface cables of R&S RT-ZVC.

The digital extension port R&S RTE-B1E provides the connectors for the R&S RT-ZVC multi-channel power probe without MSO. The connectors can be used to connect two flat interface cables of R&S RT-ZVC.

One of the options can be installed in the option slot at the rear panel.

The maximum input voltage is 40 V peak at 100 kΩ input impedance. The maximum input frequency for a signal with the minimum input voltage swing of 500 mV (V_{pp}) is 400 MHz. For detailed specifications, refer to the data sheet.

Waveform generator option R&S RTE-B6
The waveform generator can generate various function and arbitrary waveforms, sweeps, and parallel patterns. For detailed specifications, refer to the data sheet.

The option can be installed in the option slot at the rear panel.

The module provides the following connectors:

[Gen1, Gen2]	BNC connectors
[PattGen]	Connector for the pattern generator

GBIP option R&S RTE-B10
The GBIP option provides a GBIP interface and connector for remote control. For detailed specifications, refer to the data sheet.

2.2.3 Keys and Controls

2.2.3.1 POWER Key

The [POWER] key is located on the lower left corner of the front panel. It starts up and shuts down the instrument's software.

The light of the key shows the instrument state:

- Orange: standby, the main power switch is on, the software is shut down.
- Green: The instrument is ready for operation.

See also: Chapter 2.1.3, "Starting the Instrument", on page 17.

2.2.3.2 SETUP Controls

SETUP keys set the instrument to a defined state, change basic settings, and provide print and help functions. The intensity rotary knob adjusts the display contrast for several display elements.

[AUTOSET]
The instrument analyzes the enabled channel signals, and adjusts appropriate horizontal, vertical, and trigger settings to display stable waveforms.

[PRESET]
Resets the instrument to a default state. All measurements, mask tests, zoom, and most individual settings are deleted, and all channels except for channel 1 are disabled. You can define preset configurations and save them to a file. The [PRESET] key can be configured to set either factory defaults or a user-defined preset configuration.

[FILE]
Opens and closes the "File" dialog box, where you can:
- Save instrument settings (user settings)
- Load instrument settings which were saved before
- Save waveform data and measurement results
- Define a naming pattern for autonaming of files

[SETUP]
Opens and closes the "Setup" dialog box, where you can:
- Access Windows configuration and install firmware updates
- Configure the touchscreen
- Check and install option keys for software options
- Check availability of hardware options
- Configure remote settings, LAN settings, and GPIB

[PRINT]
Starts a printing or saving action, or opens a report. The function is assigned to the key in "File" menu > "Frontpanel Setup" > "Hardkeys". By default, the key saves a screenshot of the waveform display.

See also: Chapter 3.3.2, "Hardkeys: Function Assignment", on page 101

[HELP]
Opens the appropriate help topic for the active tab. If no dialog box is open, the contents page of the help appears.

[MODE]
Opens and closes a dialog box where you can change the instrument mode:
- Enable high definition mode
- Enable functionality in beta state

[DISPLAY]
Opens and closes the "Display" dialog box to configure the appearance of the waveforms, the diagram layout, color tables, and the XY-diagram. You can also assign name labels to the waveforms.

[T-SCREEN LOCK]
Locks the touchscreen to prevent unintended use. When the touchscreen is off, the key is illuminated. Press again to unlock the touchscreen.

[INTENSITY]
Adjusts the intensity of the waveforms on the screen, or the background transparency of dialog boxes, or the transparency of result boxes. If a dialog box is open, turning the knob changes the transparency of dialog boxes. If a result box is open, the transparency of result boxes is changed. Otherwise the waveform intensity is adjusted. Press the knob to toggle between the three settings. The controlled parameter and its value are shown in the input box in the upper left corner of the screen.

2.2.3.3 HORIZONTAL Controls

The keys and rotary knobs in the HORIZONTAL functional block adjust the acquisition settings and horizontal parameters. These settings are effective for all channel waveforms.

[RES REC LEN], [HORIZONTAL]
Open and close the "Setup" tab in the "Horizontal" dialog box, where you can:
- Adjust the time scale, and acquisition time
- Adjust the horizontal position, and reference point
- Adjust the resolution and the record length

[ACQUISITION]
Opens and closes the "Acquisition" tab in the "Horizontal" dialog box, where you can define the acquisition processing (acquisition mode and waveform arithmetic).

[RESOLUTION / RECORD LENGTH]
The rotary knob changes the resolution or the record length. Press the knob to toggle the setting. The controlled parameter and its value are shown in the input box in the upper left corner of the screen.

For resolution, turn clockwise to increase the resolution: the time between two acquisition points gets shorter. Record length and sample rate increase while the acquisition time remains constant.

For record length, turn clockwise to increase the record length, and the resolution increases too - the time between to acquisition points gets shorter.

[POSITION / REF POINT]
The rotary knob changes the horizontal position of the waveform or the position of the reference point on the screen.

Press the knob to toggle the setting. The controlled parameter and its value are shown in the input box in the upper left corner of the screen.

"Horizontal position" defines the time distance of the reference point from the zero point of the diagram. Turn clockwise to move the waveform to the right.

"Reference point" defines the position of the reference point on the screen. Turn clockwise to move it to the right. The reference point marks the rescaling center of the time scale. It is indicated by a gray triangle outline at the top of the diagram. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.

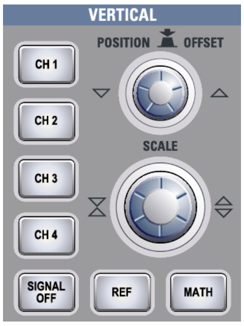
[SCALE]
The rotary knob adjusts the time scale for all signals. The time scale is also known as timebase.

Turn clockwise to stretch the waveforms. Doing so, the scale value *time/div* decreases.

Press the knob to toggle between coarse and fine scale adjustment.

2.2.3.4 VERTICAL Controls

The keys and knobs in the VERTICAL functional block select a signal and adjust the vertical scale and position of the selected signal.



[CH ×]
Turns on, selects, and configures a channel. If the channel is active, the key lights up in the corresponding channel color .

The effect of the keypress depends on state of the channel:
- If channel is off: Pressing the key turns on the channel and selects it.
- If the channel is on, but not selected: Pressing the key selects the channel waveform.
- If the waveform is selected: Pressing the key opens the "Vertical" dialog box for the appropriate channel.

The vertical rotary knobs are focused on the selected waveform. They are illuminated in the color of the selected waveform.

[REF]
Opens the "Reference" dialog box, where you can configure and display reference waveforms. Press the key repeatedly to switch the reference waveform.

If a reference waveform is selected, the vertical rotary knobs are illuminated in white or light gray (default colors), depending on the selected waveform.

[MATH]
Opens the "Math" dialog box, where you can configure the calculation of mathematical waveforms using various mathematic operations on other waveforms. Press the key repeatedly to switch the math waveform.

If a math waveform is selected, the vertical rotary knobs are illuminated in brown (default color), the brightness of the color depends on the selected waveform.

[POSITION / OFFSET] (upper knob)
The upper rotary knob changes the vertical position or the offset of the selected waveform. The horizontal axis and the selected waveform are moved vertically. The knob lights up in the color of the selected waveform.

Press the knob to toggle the setting. The controlled parameter and its value are shown in the input box in the upper left corner of the screen. Turn clockwise to move up the waveform.

- Position indicates the vertical location in divisions.
- Offset moves the vertical center of the selected channel to the offset value.

[SCALE]
This rotary knob adjusts the vertical scale for the selected waveform. The knob lights up in the color of the selected waveform.

Turn clockwise to stretch the waveform. Doing so, the scale value V/div decreases.

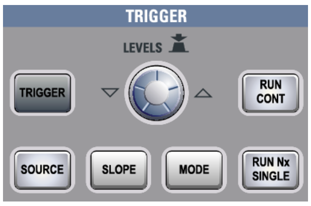
Press the knob to toggle between coarse and fine scale adjustment.

[SIGNAL OFF]
Turns off the selected signal and selects the next channel, math, or reference waveform.

The key is illuminated in the color of the selected signal and changes the color according to the new selection.

2.2.3.5 TRIGGER Controls

The keys and knob in the TRIGGER functional block adjust the trigger and start or stop acquisition.



[TRIGGER]
Opens and closes the "Trigger" dialog box, where you can:
- Select a trigger type and configure it.
- Set general trigger parameters and control the acquisition run.

- Configure a sequence of subsequent trigger events.

[LEVELS]
The rotary knob sets the trigger level for all trigger types. Turn clockwise to move up the trigger level. If the selected trigger type requires two trigger levels - upper and lower level - press the knob to toggle between the two levels.

[SOURCE]
Opens a dialog box where you can select the trigger source. Press the key again to switch the source. The key lights up in the color of the selected trigger source.

[SLOPE]
Toggles the trigger slope or trigger polarity, dependent on the trigger type. The current setting is shown on the trigger label, which is in the upper part of the signal bar.

[MODE]
Toggles the trigger mode between Auto and Normal. The current setting is shown on the trigger label.

[RUN CONT]
Starts and stops the continuous acquisition. A green light indicates a running acquisition. A red light shows that acquisition is stopped.

[RUN N× SINGLE]
Starts a defined number of acquisitions. A green light indicates a running acquisition. A red light shows that acquisition is stopped. Press the key again to stop running acquisitions.

To set the number of acquisitions, press the [TRIGGER] key, select the "Ctrl/Action" tab, and set "Average count (N-single count)".

2.2.3.6 ANALYSIS Keys

The keys in the ANALYSIS functional block provide direct access to measurement and analyzing functions. If you press [CURSOR], [ZOOM] or [MEAS], the action starts on first keypress, and a second keypress opens the corresponding dialog box. If you press another function key, the dialog box opens.

[CURSOR]
Displays vertical and horizontal cursors in the active diagram and opens the "Cursor Results" box.

Cursors are markers which are placed at points of interest on a waveform. The instrument measures the cursor positions and delta values between parallel cursors.

If you press the key while a cursor measurement is enabled, the "Cursors" dialog box opens.

In the "Cursors" dialog box, you can:
- Configure up to 2 cursor sets
- Define style and labels of the cursors
- Connect the cursor to the waveform and couple the cursors

[MEAS]
Starts the default automatic measurement for the active waveform and opens the "Measurement" result box.

If you press the [MEAS] key while a measurement is enabled, the "Measurements" dialog box is displayed, where you can:
- Configure amplitude and time measurements, eye, spectrum, and histogram measurements
- Configure gated measurement
- Configure long term and statistic measurements

[ZOOM]
Displays a zoom diagram for the active diagram. The key is illuminated if at least one zoom is active. If you press the key while the zoom function is on, the "Zoom" dialog box opens, where you can configure several zoom areas for detailed signal observation.

[PROTOCOL]
Opens the "Protocol" dialog box which contains the configuration of serial buses and the settings for decoding the signals.

The key lights up if the decoding of a serial bus is active. You can switch off the decoded bus using the [SIGNAL OFF] key.

[SEARCH]
Opens and closes the "Search" dialog box, where you can:
- Configure trigger events to be searched for
- Limit the search by gating
- Configure the presentation of search results

[MASKS]
Opens and closes the "Masks" dialog box. Masks are used for error detection and compliance tests of digital signals.

You can:
- Configure masks and masks segments
- Define mask test parameters
- Configure actions triggered by mask violation
- Configure the mask display

[LOGIC]
Opens the dialog box for configuration of parallel buses and digital channels. The key lights up if you enable at least one parallel bus. You can switch off the selected bus using the [SIGNAL OFF] key.

[HISTORY]
The sample memory contains several stored acquisitions before the current one, which is shown in the display. Press the key to open the quick access "History" dialog box, where you can view the stored acquisitions and use them for further analysis. Press the key again to open the main "History" dialog box with more settings and information.

The key is illuminated as long as a history acquisition or replay is displayed.

2.2.3.7 NAVIGATION Controls

The rotary knob and the navigation keys provide an alternative way to navigate in dialog boxes and to enter numeric data.

See also: Chapter 2.4.9, "Using Dialog Boxes", on page 87

Navigation rotary knob
The navigation knob has various functions:
- In numeric entry fields: turn to increase or decrease the value.
- In tables: press to activate the edit mode, turn clockwise to increase the value or turn counterclockwise to decrease it, and press to enter the value and move to the next cell.
- To set cursor positions, histogram areas, and mask points in input boxes: press to toggle the parameter, turn clockwise to increase the value or turn counterclockwise to decrease it.
- To move zoom area, cursor line, or gate in diagrams: Turn to move the element that has the focus, and press to toggle the focus.

[ESC]
Closes a dialog box or input box.

[UNDO]
Reverses the last setting actions step by step. Undo is not possible after load and recall actions, and after creating a reference waveform.

[REDO]
Recovers the undo steps in reverse order.

[ENTER]
The [ENTER] key has various functions:
- In dialog boxes and opened selection lists: the key applies the selected value.

- In tables: the key activates the edit mode. If the table cell is in edit mode, the key confirms the value, quits the edit mode and moves to the next cell.

Field left, Field right
In dialog boxes and tables, the keys move the focus.

In diagrams, they switch the focus between zoom areas, cursor lines, and gates.

Checkmark [☑]
The checkmark key [☑] has different functions depending on the focus:
- In usual dialog box: if the focus is on a selection list, the key opens the list and applies the selected value.
- In tables: activates the edit mode.

Tab
The tab key has various functions:
- In dialog boxes with only horizontal tabs, the key switches the horizontal tabs.
- In dialog boxes with horizontal and vertical tabs, the key switches the tabthat has the focus.
- In a table or diagram, the key moves the focus in the same way as the [⇨] key.

Up arrow [⇧], Down arrow [⇩]
The up and down arrow keys have the following effects:
- In numeric edit fields: increase or decrease the parameter value.
- In tables: scroll vertically through the rows.
- In dialog boxes, for option buttons in a column: select an option. In an open selection list, the keys scroll the list.

Left arrow [⇦], Right arrow [⇨]
The left and right arrow keys have the following effects:
- In edit fields: move the cursor.
- In tables: scroll horizontally through the columns.
- In dialog boxes, for option buttons in a row: select an option.

2.3 Trying Out the Instrument

This chapter introduces the most important functions and settings of the R&S RTE step by step. The complete description of the functionality and its usage is given in the "User Manual". Basic instrument usage is described in Chapter 2.4, "Operating the Instrument", on page 67.

Prerequisites

- The instrument is connected to the mains system, and started up as described in Chapter 2.1, "Preparing for Use", on page 15.
- A probe is available.

For these first measurements, you use the internal calibration signal, so you do not need any additional signal source or instruments. Try out the following:

- Displaying a Basic Signal....................36
- Acquiring Data....................38
- Changing the Waveform Scaling and Position....................39
- Zooming into the Display....................43
- Displaying the Waveform History....................46
- Showing Basic Measurement Results....................48
- Performing a Basic FFT Analysis....................55
- Performing Mathematical Calculations....................57
- Performing a Search....................58
- Performing a Mask Test....................60
- Printing and Saving Screenshots....................63
- Saving Data....................65

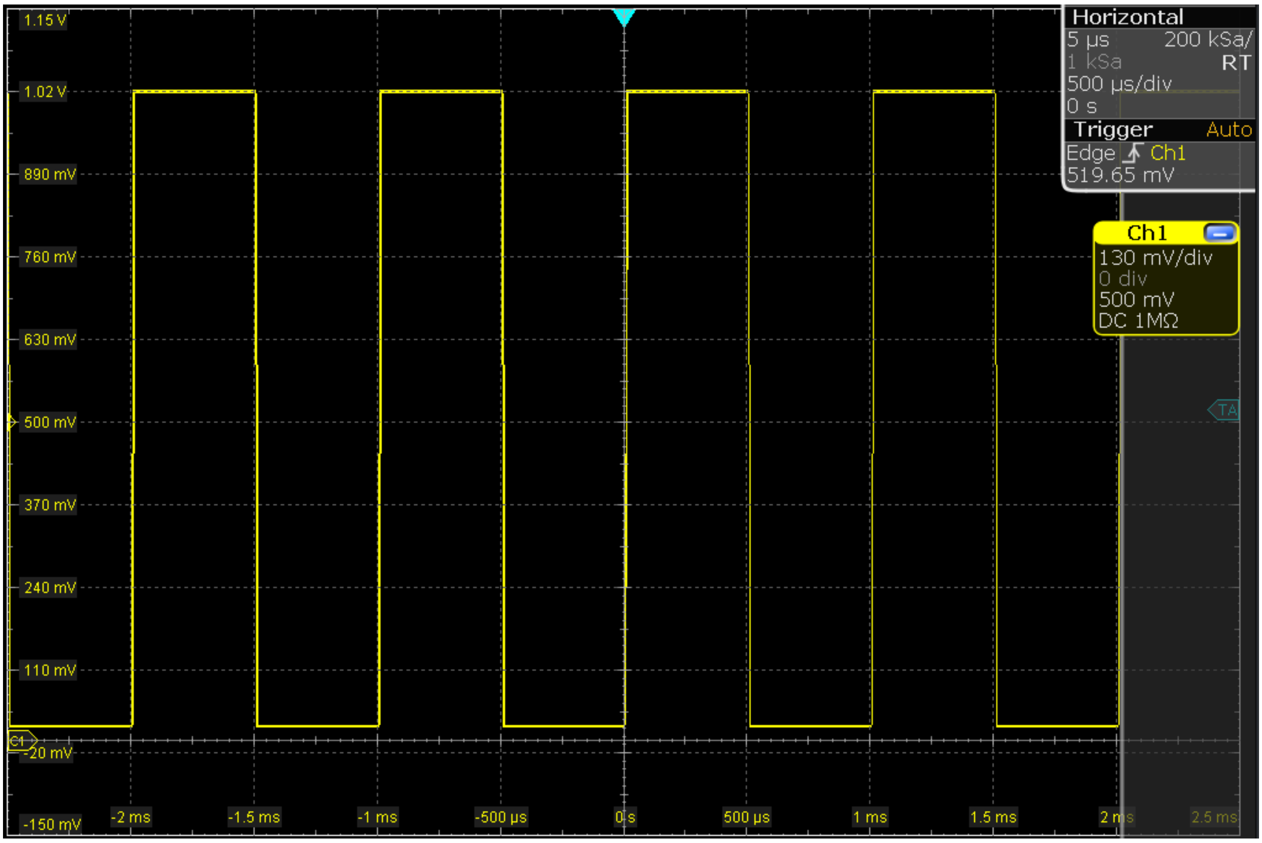
2.3.1 Displaying a Basic Signal

Displaying the input from a signal channel is simple and straightforward. Furthermore, you get to know some basic trigger functions. The R&S RTE provides wide-ranging trigger functions to find various signal anomalies, which are described in the instrument help and in the "User Manual".

1. Press the [PRESET] key on the front panel (in the [SETUP] area on the left).
2. Connect the probe to the input connector [CH 1].
   Connect the probe's ground connector to the right compensation pin ⊥, and the tip to the left pin ⎍.

   The instrument recognizes the probe, and a signal is displayed in the diagram.
3. Press the [AUTOSET] key on the front panel (in the [SETUP] area on the left).

   Autoset finds appropriate horizontal and vertical scales and trigger conditions to present a stable square waveform. The trigger is set to edge trigger on rising edge with auto trigger mode.



4. If necessary, compensate the passive probe as described in Chapter 4.10.1, "Adjusting Passive Probes", on page 196.
5. In the [TRIGGER] area of the front panel, press the [SOURCE] key. Press the key again to switch the trigger source to "C2".

   An unstable waveform is displayed. In auto mode, the instrument triggers repeatedly after a time interval if no real trigger occurs.
6. In the [TRIGGER] area, press the [MODE] key.
7. Check the "Trigger" settings in the upper right corner of the screen.

   The trigger mode has changed to "Normal". The waveform is no longer refreshed, and "Wait" is displayed in the trigger settings. The instrument cannot find a real trigger event because there is no signal on channel 2.
8. Tap the "Undo" icon on the toolbar repeatedly until the trigger mode is "Auto" and the trigger source is "CH1".

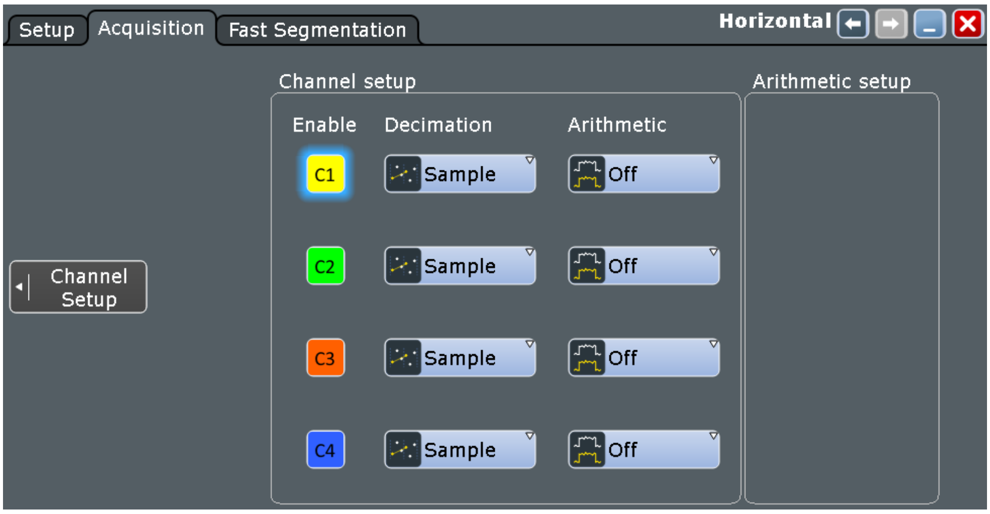
   
9. Press the [SLOPE] key to toggle the trigger slope.
   Watch the waveform and the "Trigger" settings.

2.3.2 Acquiring Data

You can acquire data using different arithmetic methods to get envelope or average waveforms.

If you need to change only one setting during analysis, and you need to change it often, you can reduce the dialog box to a small box that only contains the required setting. Thus you can change the setting and see the result immediately.

1. Press the [ACQUISITION] key on the front panel, in the [HORIZONTAL] area.
2. In the "Acquisition" tab of the "Horizontal" dialog box, select the "Decimation" type *Sample*.



3. Tap the "Arithmetic" button for C1 but do not change the setting.
4. Tap the "Minimize" icon in the upper right corner of the dialog box.

   The dialog box turns into a mini box that contains only the "Wfm Arithmetic" setting.

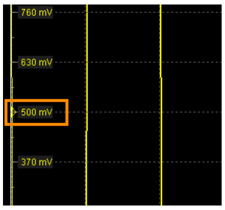
5. Select *Envelope* and check the waveform.

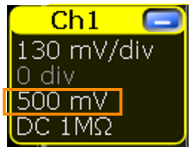
   The instrument sets the "Decimation" type automatically to *Peak detect* to display the correct envelope waveform.
6. Select *Average* and check the waveform.
7. Tap the "Maximize" icon in the mini dialog box.

   The complete "Acquisition" dialog box is restored.
8. Close the dialog box by tapping ☒.

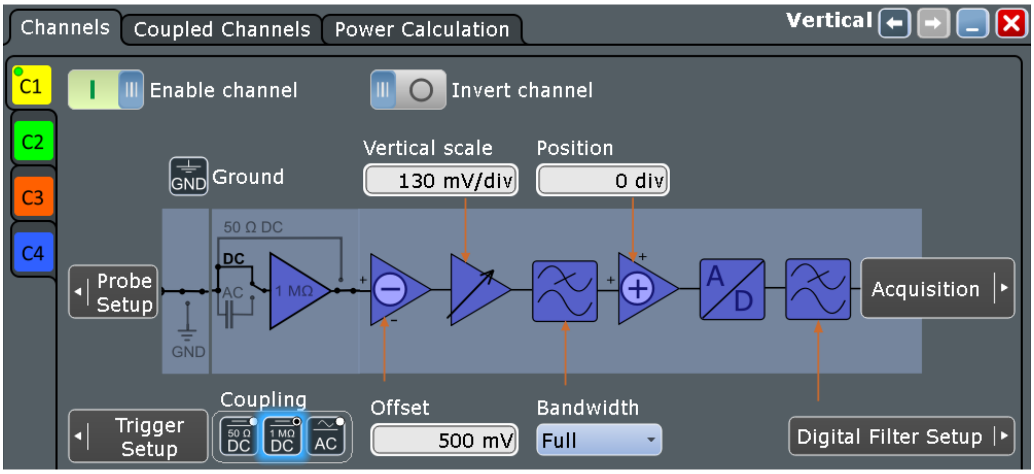
2.3.3 Changing the Waveform Scaling and Position

As you can see on the y-axis of the display, the calibration signal has a vertical offset of about 500 mV. The value can differ.





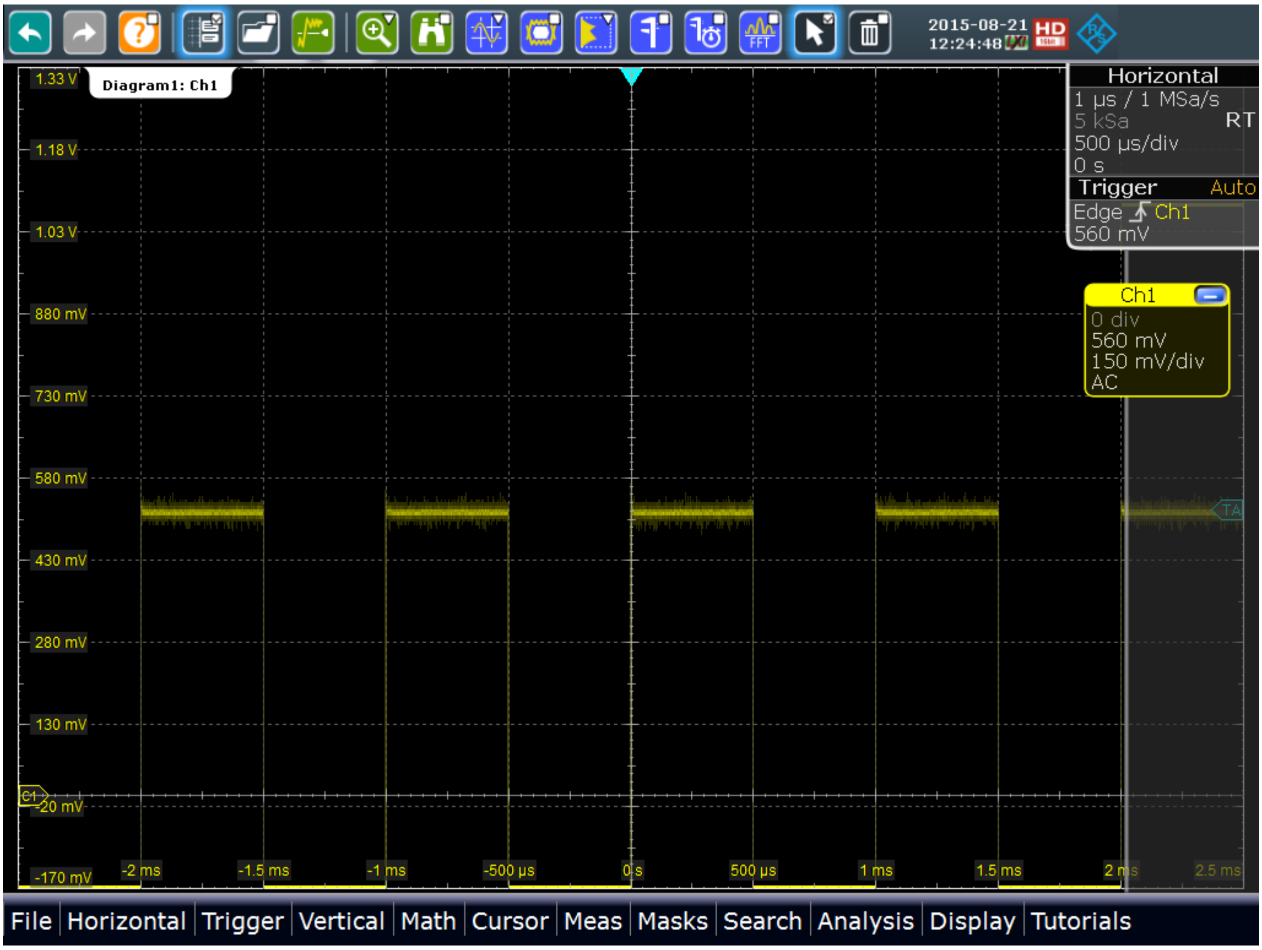
This value is also indicated in the signal icon for channel1 (3rd row). If you press the [Ch1] key, the "Vertical" settings dialog box also displays the "Offset" value. The offset is the DC component of the signal.



If you use a passive probe, you can filter the DC component by using the AC coupling function. Then you quickly find the new trigger level, and try out the scaling functions:

1. Press the [Ch1] key on the front panel (in the [VERTICAL] area) to display the "Vertical" dialog box.
2. Change the "Coupling" to "AC". Close the dialog box.

   The DC component of the signal is eliminated; the waveform position moves down vertically and is now centered on 0 V.

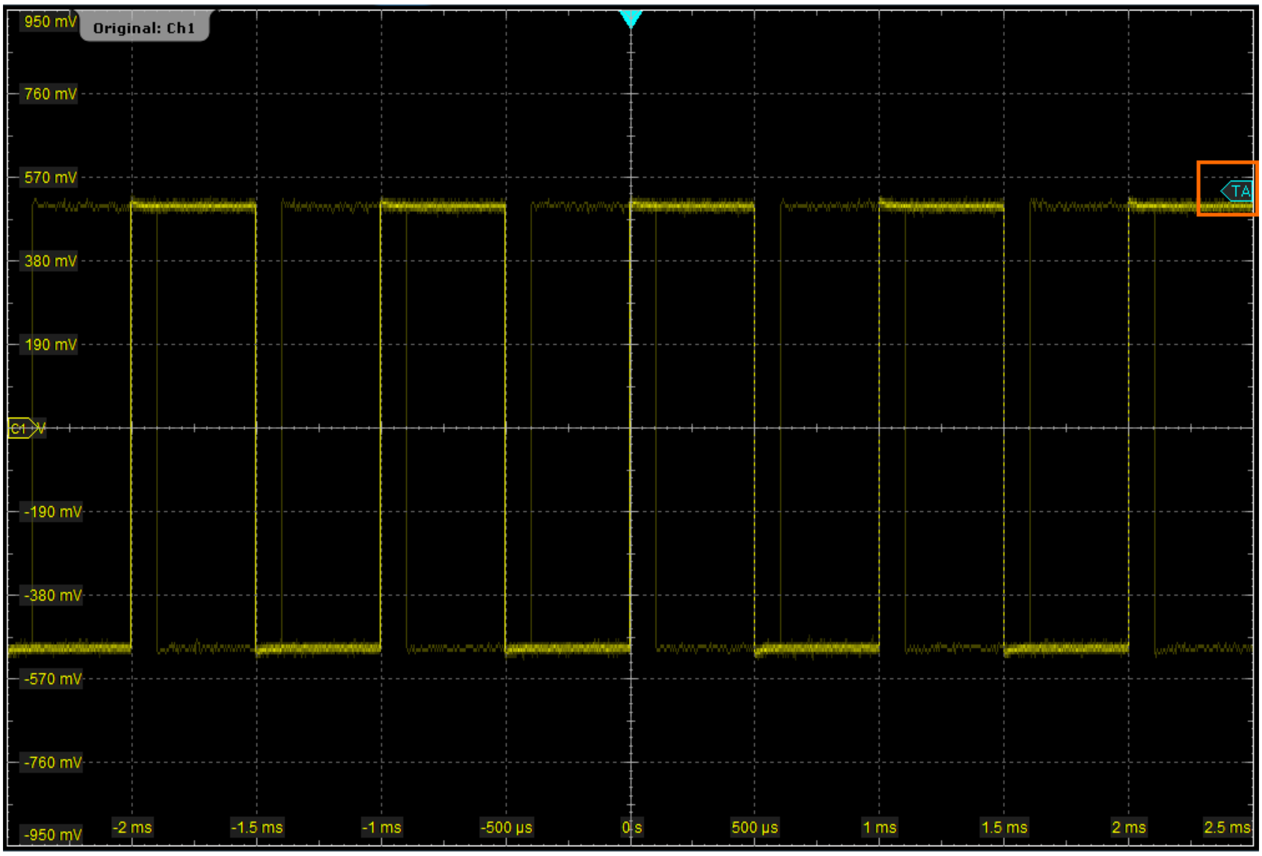


3. To move the waveform back to the center of the screen, eliminate the offset in the vertical settings:
   a) Press the [Ch1] key again.
   b) Enter *0 V* in the "Offset" field.

   The waveform is now displayed in the center of the display, with the x-axis crossing at 0 V. The waveform can be unstable if the trigger level is above.
4. Tap the "Show signal bar" icon on the toolbar.

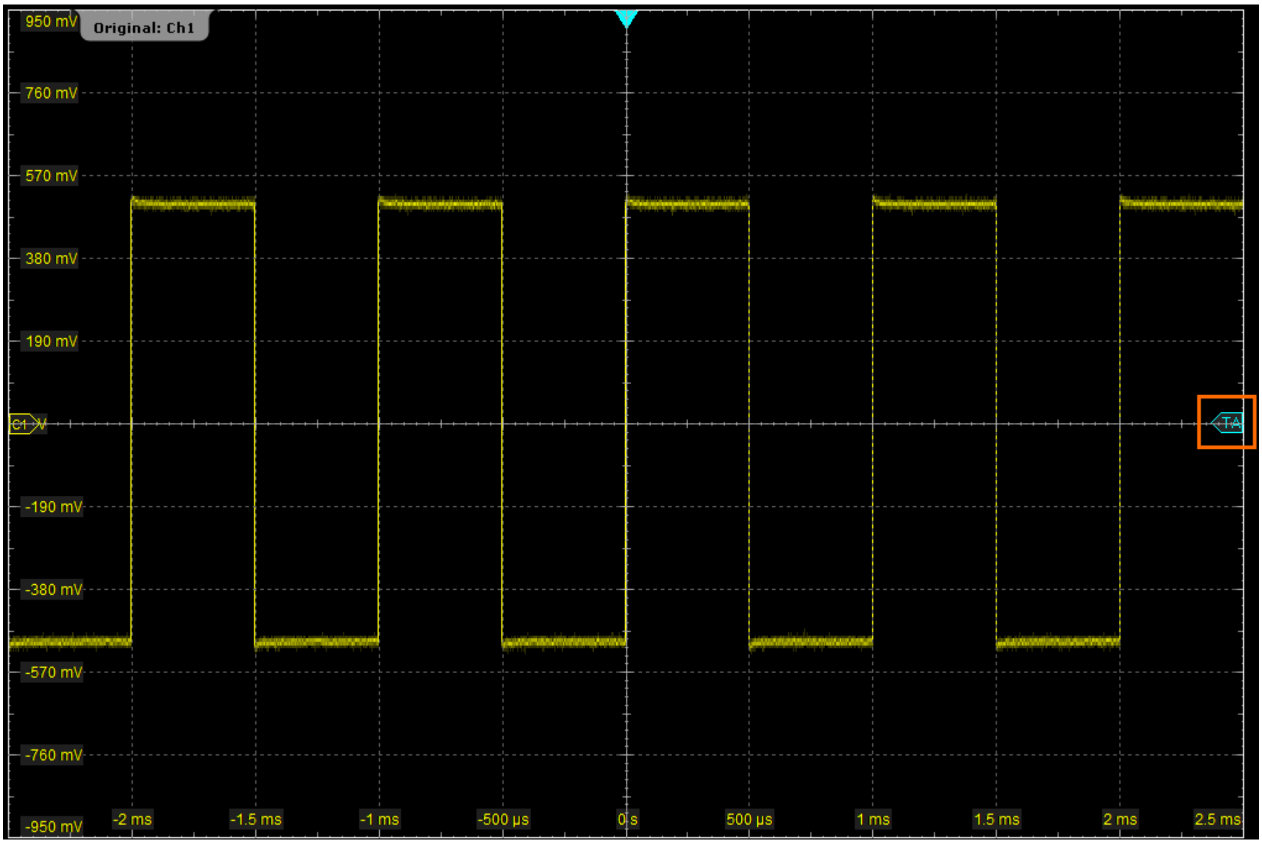
   

   The signal bar disappears, and you can see the trigger level marker on the right.

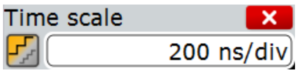


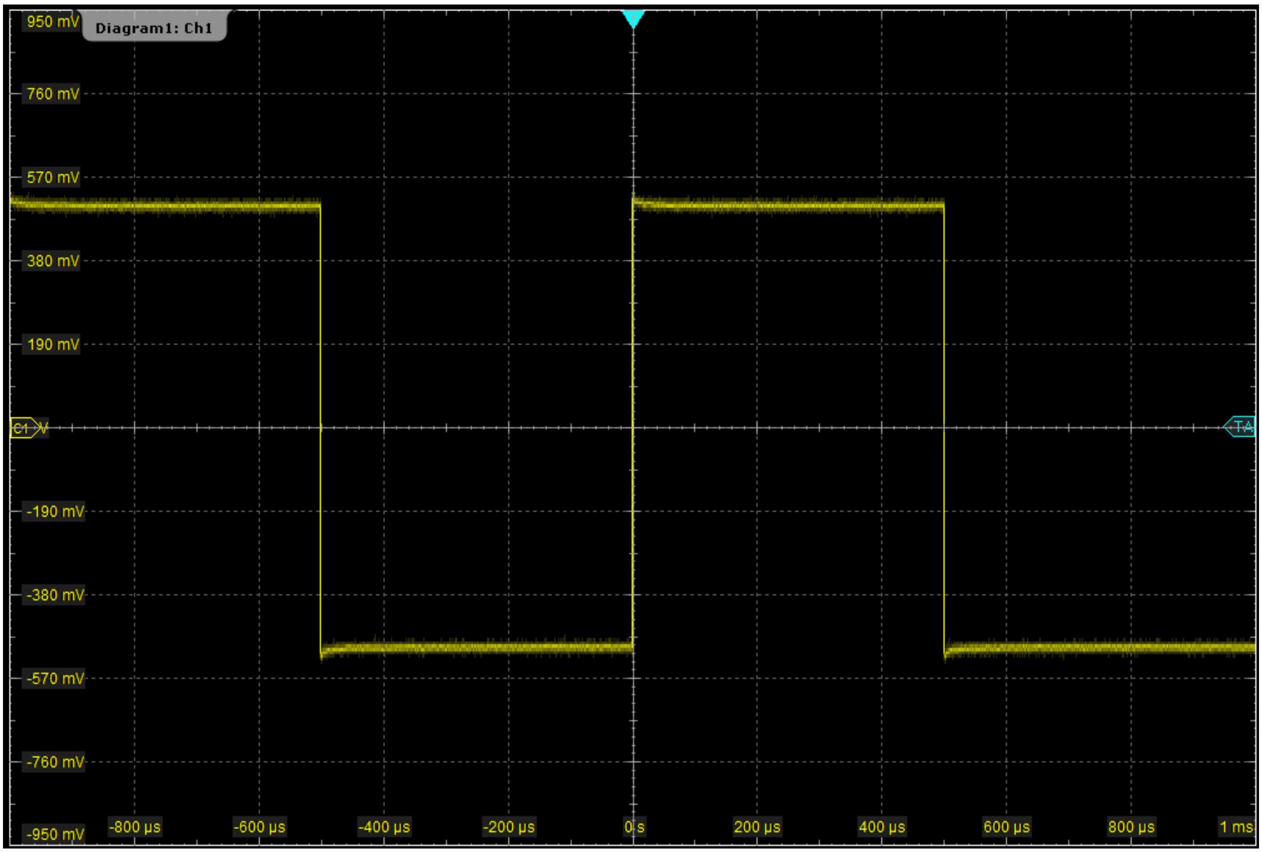
5. Tap the "Find level" icon on the toolbar.

The instrument adjusts the trigger level, and a stable waveform is displayed.



6. To examine one pulse in the signal in more detail, turn the horizontal "Scale" rotary knob.
   The current scale factor per division is displayed in the upper left-hand corner of the display while you turn the knob. You can switch between a small and large step size in the scaling factor by tapping the step icon.



7. To return to the original scaling, try the [UNDO] key in the [NAVIGATION] area:
   a) Press the [UNDO] key repeatedly until the original scaling is displayed.
   b) Press the [REDO] key to retrace the undone steps. Thus, you can toggle between the two displays using the undo and redo keys until you perform a different action.

   Tip: Instead of using the [UNDO] and [REDO] keys, you can tap the corresponding icons on the toolbar.

If you use a Rohde & Schwarz active single-ended probe, you can measure the DC component of the signal directly at the probe tip with the integrated R&S ProbeMeter:

1. On the "Vertical" menu, tap "Probe Setup".
2. Make sure that the correct channel is selected on the left tab.
3. In the "Additional" section, tap "ProbeMeter".

   A result box shows the DC voltage measured by the R&S ProbeMeter.

2.3.4 Zooming into the Display

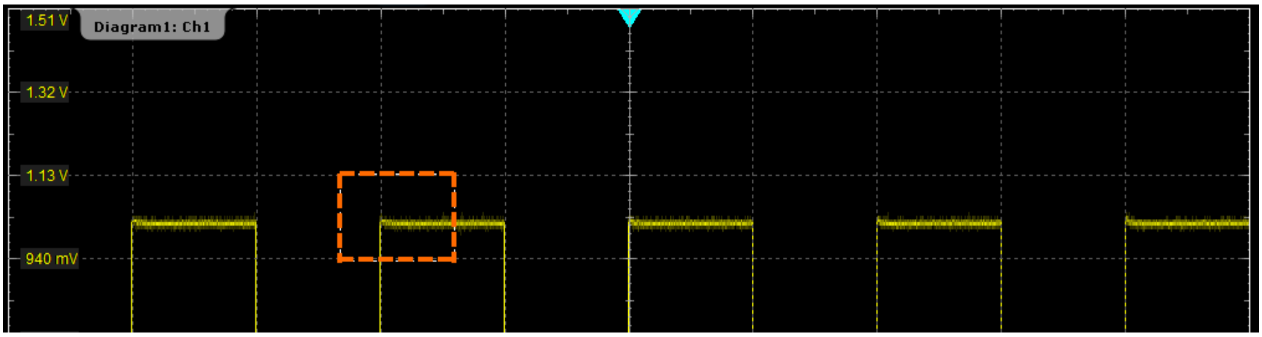
Using the [SCALE] rotary knobs, you can change the scaling of the time base and signal amplitudes in order to enlarge the waveform. If you need to see more details, use one of the zoom functions. The instrument has 4 zoom types, 2 of them you try out in this chapter.

2.3.4.1 Using the Standard Zoom

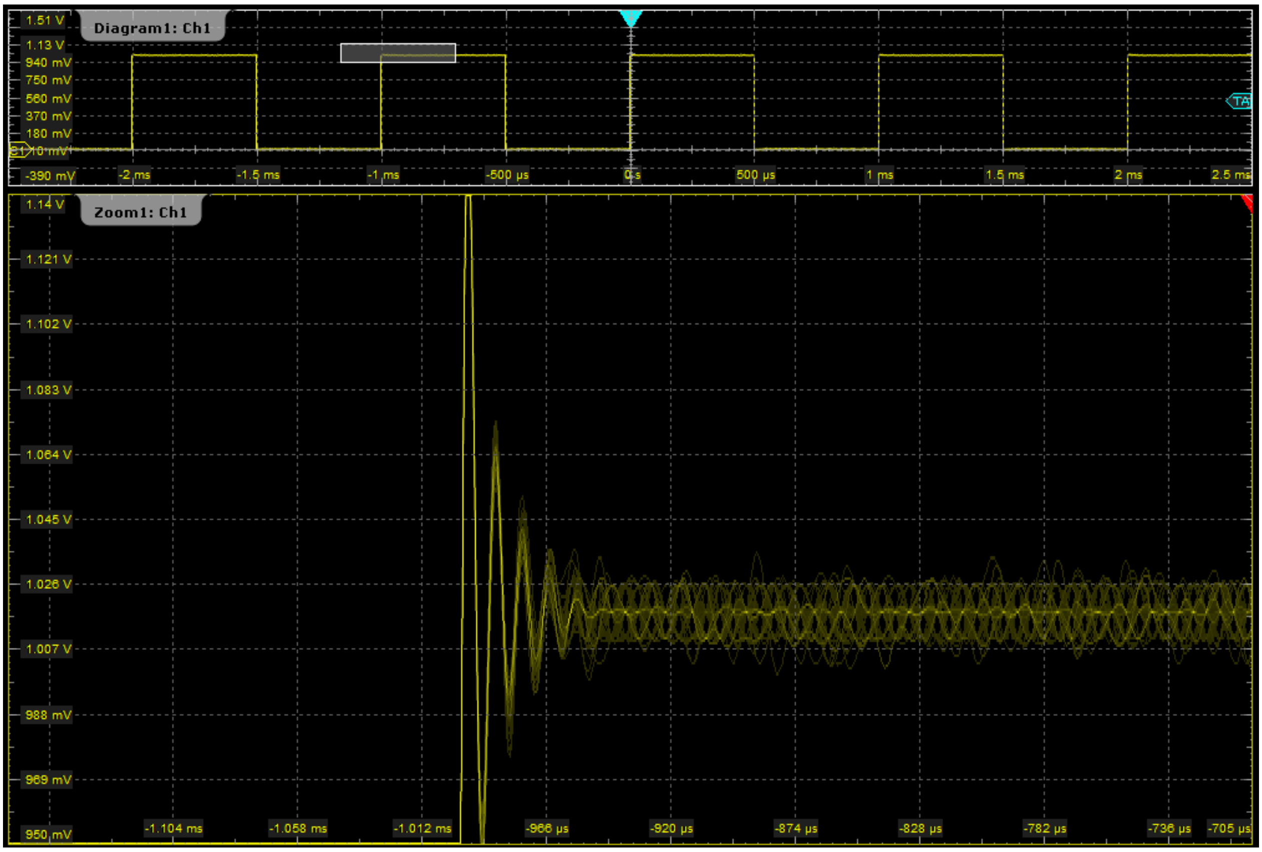
1. Restore the default signal channel settings by pressing the [PRESET] and [AUTOSET] keys.
2. On the toolbar, tap the "Zoom" icon.

3. Tap the position in the diagram that you want to define as one corner of the zoom area. Then drag your finger to the opposite corner of the zoom area.
   While you drag your finger on the touchscreen, a dotted rectangle is displayed to indicate the current zoom area. When the rectangle covers the required zoom area, remove your finger.



The indicated area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle.



4. To remove the zoom window and make room on the display for other results, tap the "Delete" icon and then the zoom window.

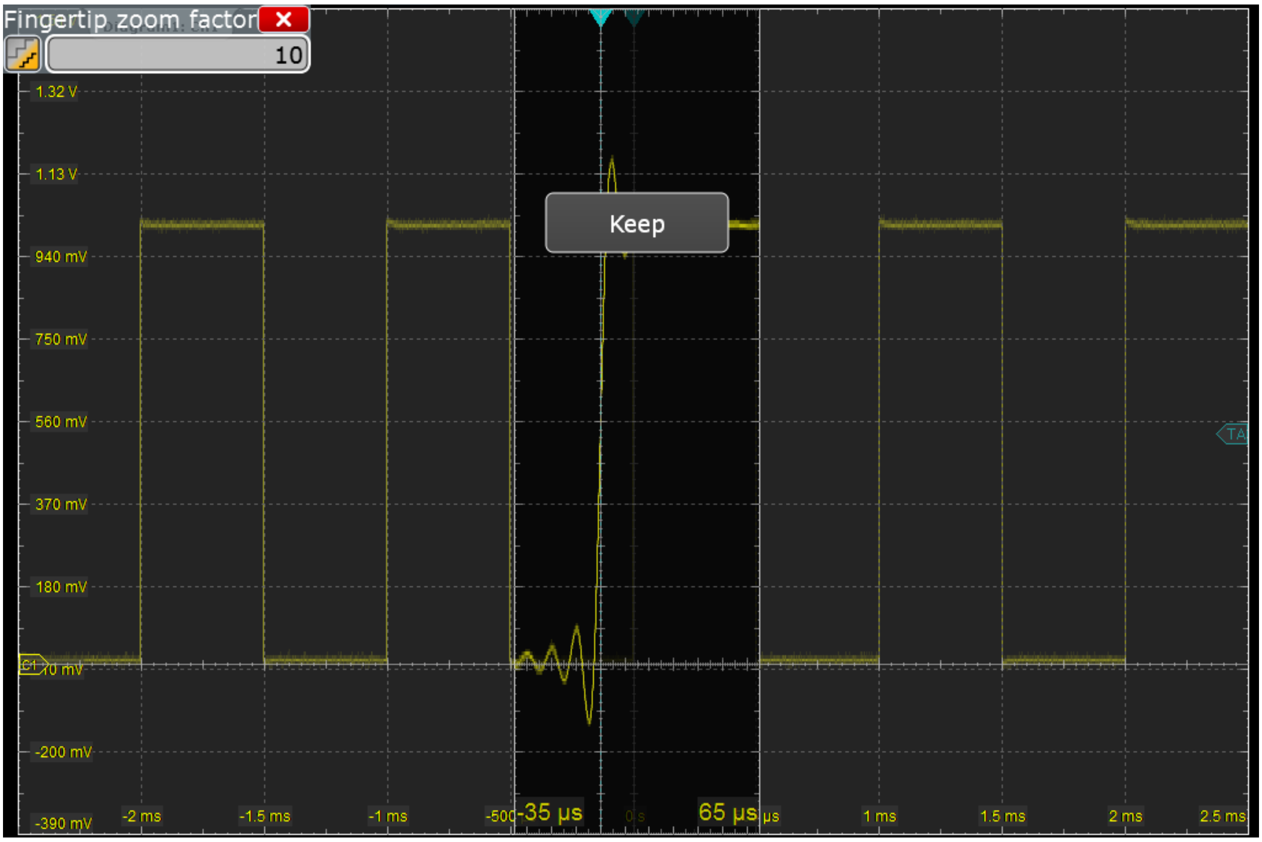
2.3.4.2 Using the Fingertip Zoom

1. Touch the "Zoom" icon on the toolbar and drag it down until the "Zoom" toolbar menu opens. Release the finger.
2. Tap "Fingertip zoom".

3. Touch and hold the waveform and move your finger slowly in horizontal direction.

Tip: You can turn the NAVIGATION knob to change the zoom factor while holding the waveform.

4. Release the finger when the waveform segment of interest is visible in the zoom.
5. Tap "Keep" to convert the fingertip zoom into a standard zoom diagram.



2.3.5 Displaying the Waveform History

During a continuous acquisition, the instrument stores the acquired data in the memory and shows the current acquisition on the display. The history accesses and displays the samples that were saved before the current acquisition. When the acquisition was stopped and a new acquisition is started with [RUN CONT] or [RUN N× SINGLE], the memory is cleared and written anew.

In the following example, you acquire 10 waveforms, and display the 3 most recent waveforms.

1. In the [HORIZONTAL] area, press the [ACQUISITION] key.

   The "Horizontal" settings dialog box opens.
2. Set the "Average count" to *10* to perform 10 waveform acquisitions.
3. Close the "Horizontal" dialog box.
4. In the [TRIGGER] area, press the [RUN N× SINGLE] key.

10 waveform acquisitions are performed. The most recent acquisition is displayed in the diagram.

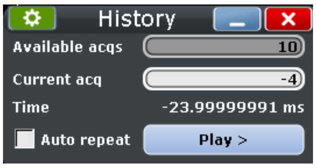
5. In the [ANALYSIS] area, press the [HISTORY] key.

   The quick access "History" dialog box appears, and the history mode is on.

6. Tap "Play".

   The 10 stored waveforms are displayed one after the other, but very fast.

7. In the "Current acq." field, enter *-4* to display the sixth waveform, counted from acquisition start. The latest acquisition has the number 0, the oldest has -9.



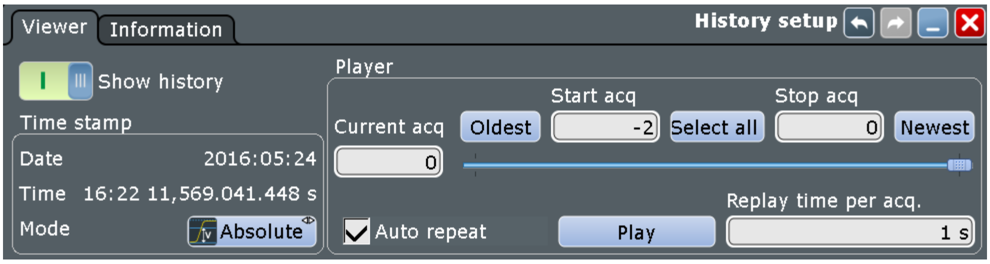
8. Tap ⚙ to open the "History" setup dialog box.

9. Enter *-2* in the "Start acq" field. Tap "Newest" to enter *0* in the "Stop acq" field.

   Thus the three latest acquisitions are displayed.

10. In the "Replay time per acq." field, enter *1 s* to display each waveform for one second.

11. Enable the "Auto repeat" option to see the three waveforms repeatedly.



12. Tap "Play".
    The currently displayed waveform is indicated in the "Current acq." field.

13. Close the "History" dialog box so you can see the waveform better.

14. Tap "Running" to stop the display.
    During running display, the "Play" is labeled "Running".

15. Close the quick access "History" dialog box.

    The history mode is disabled. The [HISTORY] key is no longer illuminated.

2.3.6 Showing Basic Measurement Results

Using the R&S RTE you can perform and display different measurements simultaneously. The color of the results in the result table corresponds with the source waveform color.

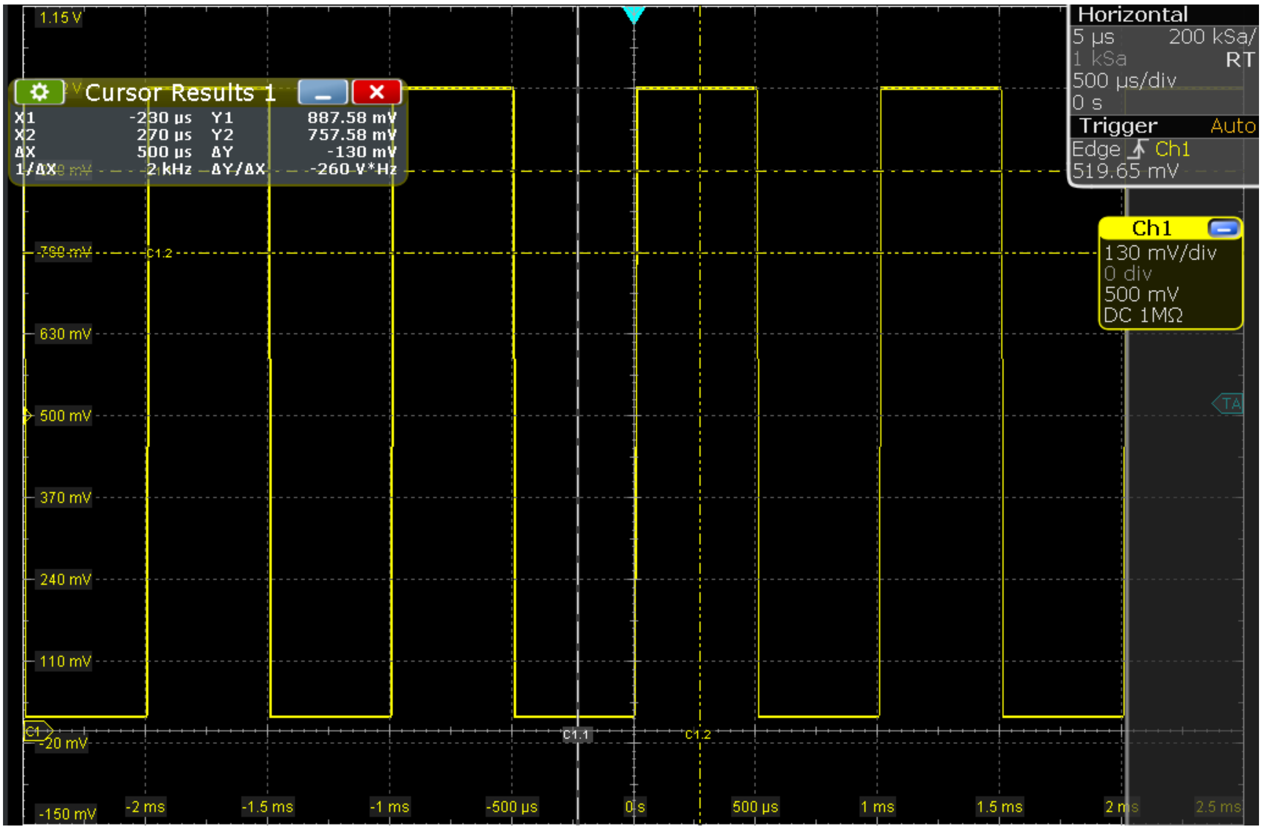
2.3.6.1 Performing a Cursor Measurement

1. Restore the default signal channel settings: Press the [PRESET] and [AUTOSET] keys.
2. Tap the "Cursor" icon on the toolbar.

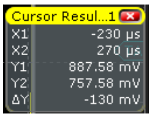
   

3. Tap the diagram in which you want to set the cursors.
   Alternatively, draw a rectangle on the screen to position the cursor lines.

   The cursor lines appear in the diagram and the "Cursor Results" box opens. The measured values of the waveform at the cursor positions are displayed.
4. You can move the cursor lines in different ways:
   - Touch a cursor line and drag it on the screen.
   - Tap a cursor line to activate it. Turn the NAVIGATION knob to adjust the position.
   - Double-tap a cursor line. Enter the position values in the cursor setup.



5. To save space in the display, minimize the result box. The most important results are displayed and updated in the result icon, as well.



6. To remove the result icon and make room on the display for other results, tap the red cross on the icon label.

2.3.6.2 Performing Automatic Amplitude Measurements

You can start up to 8 automatic measurements to run in parallel. The "Automatic measurment" icon starts the measurements one after the other.

In the following example you start and configure 3 automatic measurements: amplitude, rise time, and fall time measurement, and you display the statistical evaluation.

1. Restore the default signal channel settings by pressing the [PRESET] and [AUTOSET] keys.
2. Tap the "Automatic measurment" icon. Then tap the waveform.

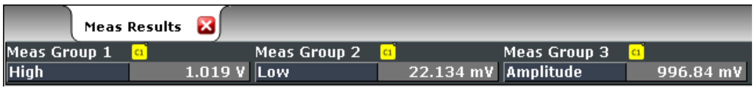
By default, the measurement of the high signal level is started.

3. Tap the "Automatic measurement" icon again. Then tap the waveform.

   By default, the measurement of the low signal level is started.

4. Tap the "Automatic measurement" icon again. Then tap the waveform.

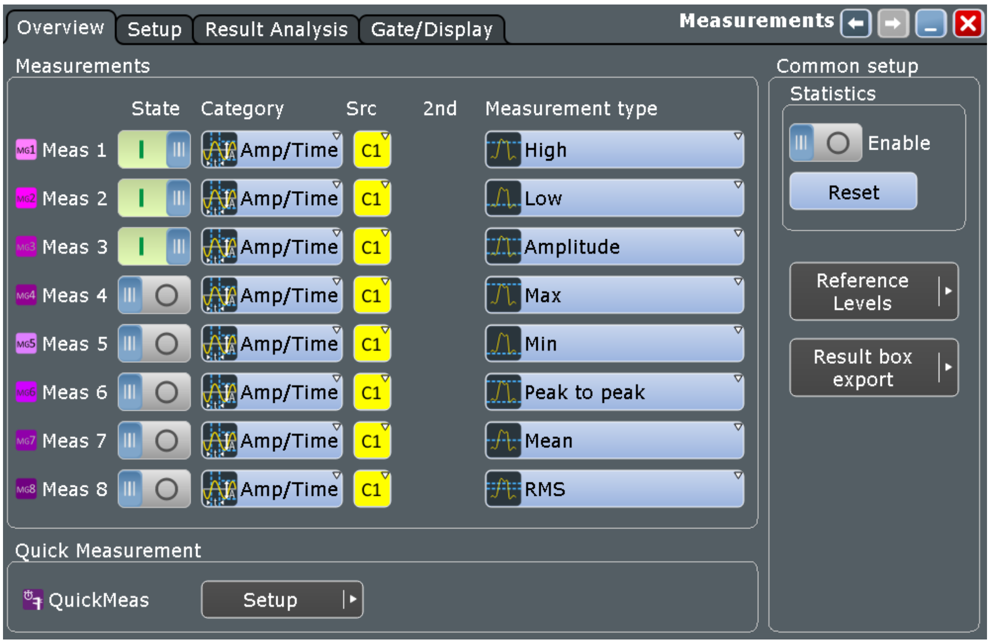
   By default, the amplitude measurement is started.



5. Double-tap the results to open the "Measurements" dialog box.

   In the "Setup" tab, you see the activated measurements and the assigned measurement types.

6. In the "Meas1" line, tap the measurement type button labeled "High".

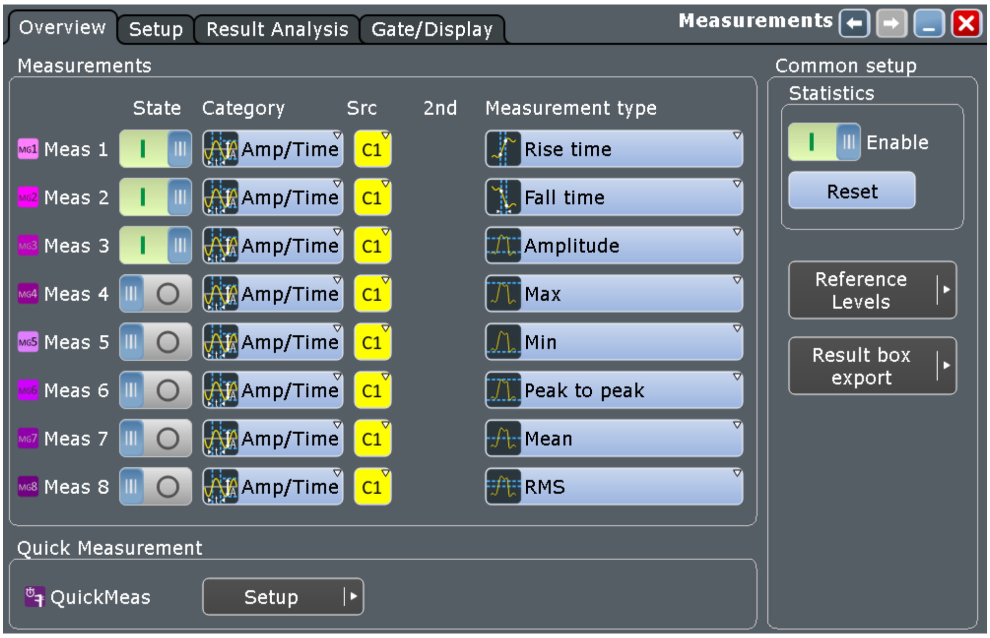


7. Select "Rise time".

8. In the "Meas2" line, tap the measurement type button labeled "Low".

9. Select "Fall time".

10. Under "Statistics", tap "Enable".



11. Close the "Measurements" dialog box and check the results.
12. To see all results, drag up the border of the results tab.
13. Close the "Measurement Results" box.

2.3.6.3 Performing and Configuring the Quick Measurement

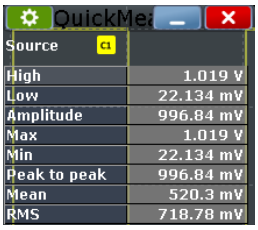
A set of up to eight different measurements on one source can be performed at once, simply by tapping the "Quick measurement" toolbar icon. The results are displayed in a result box. You can configure the measurement types to be included in quick measurement. This way, repeating measurements are performed very quickly.

In the following example, you start a quick measurement and change the QuickMeas configuration.

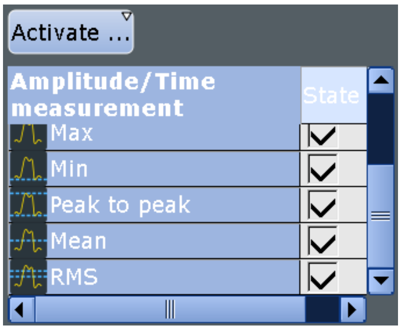
1. Press [AUTOSET].
2. Tap the "Quick measurement" icon on the toolbar.

3. Tap the diagram.

The result box shows the results of the default quick measurement.

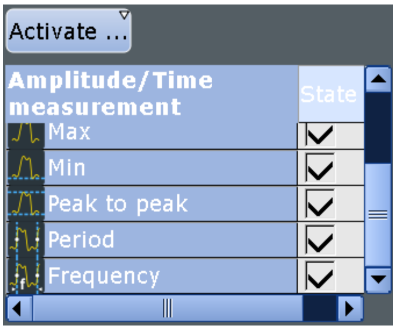


4. Tap the icon to open the "Measurements" dialog box.
5. Select the "Quick Meas" "Setup".
6. Scroll down in the table and disable the Mean and RMS measurements.



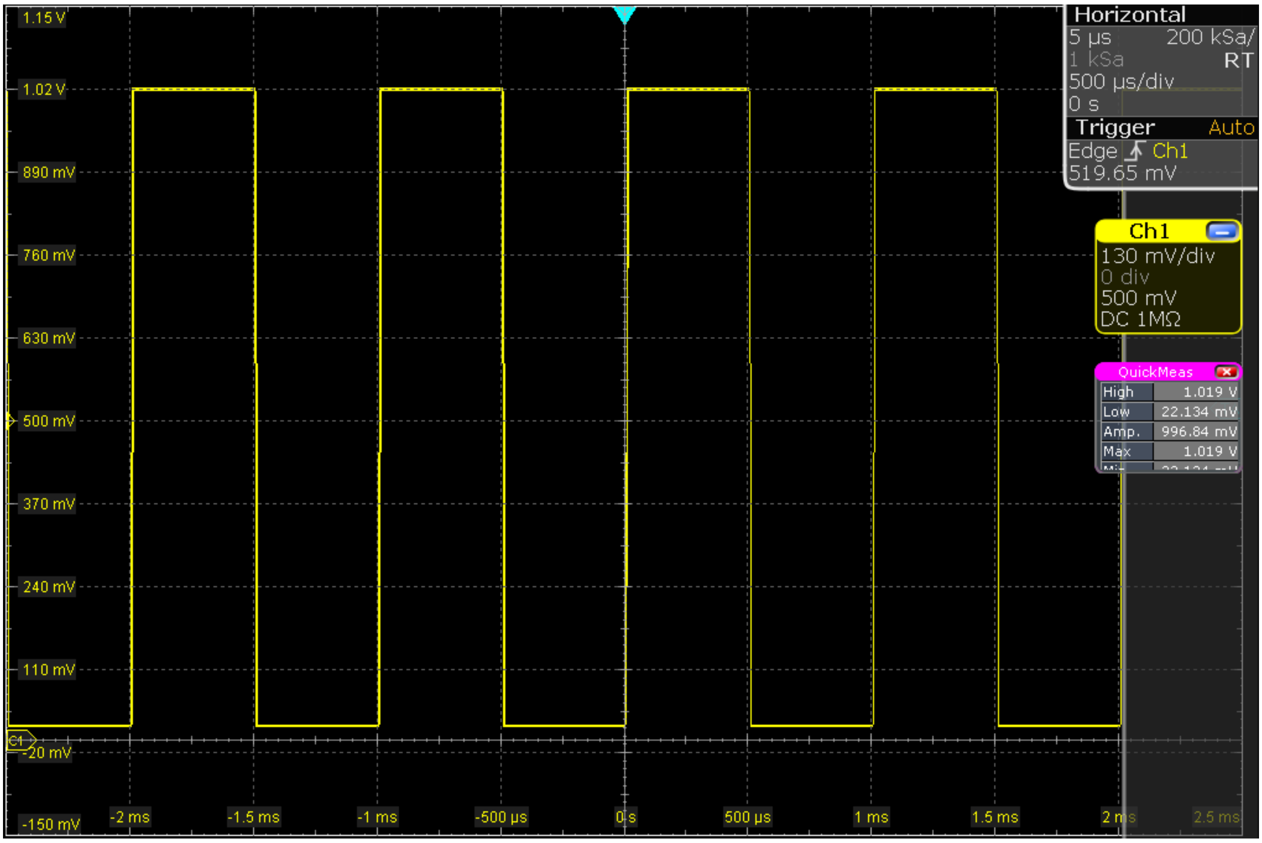
7. Tap "Activate" and select *Period*.

   Now the result box also shows the result of the period measurement.
8. Tap "Activate" and select *Frequency*.



   Now the result box also shows the result of the time measurements.
9. Tap "Set as QuickMeas".

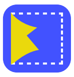
   The current configuration is set as default quick measurement and can be repeated until you save another configuration.
10. Close the dialog box.
11. To save space in the display, minimize the result box:
   The most important results are displayed and updated in the result icon.
   Do not close the result icon, as you need the results for the Search example (see Chapter 2.3.9, "Performing a Search", on page 58).



2.3.6.4 Displaying a Histogram

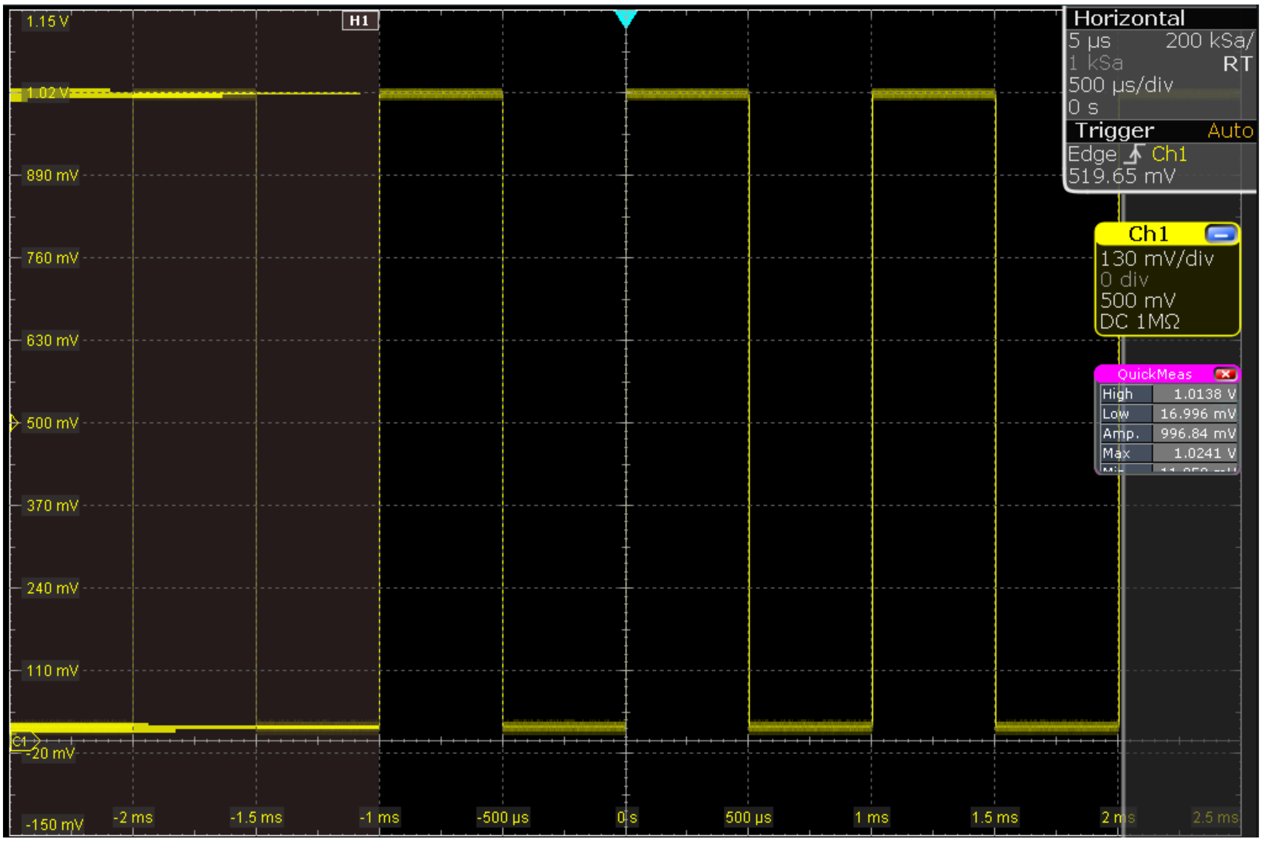
Histograms are useful to analyze the occurrence of measurement values statistically.

1. Tap the "Histogram" icon on the toolbar.

2. Tap the diagram in which you want to generate the histogram.

   You can also draw a rectangle on the screen to define the area on which the histogram is based. The histogram range is indicated in the diagram and a vertical histogram is defined and displayed.



3. To display the measurement results for the histogram, tap the "Measurement" icon on the toolbar.

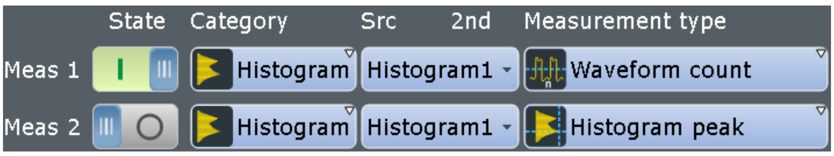
   

4. Tap the histogram.

   The waveform count for the histogram is displayed.

5. To get more measurement results for the histogram, double-tap the results. Alternatively, you can press the [MEAS] key on the front panel.

6. For "Meas 2", select "Category = Hist" and the required measurement type.

7. To finish the measurement, tap the "Close" icon in the result box.

8. To remove the histogram, tap the "Delete" icon on the toolbar and then the histogram.

   Both the histogram and any measurements based on that histogram are deactivated.

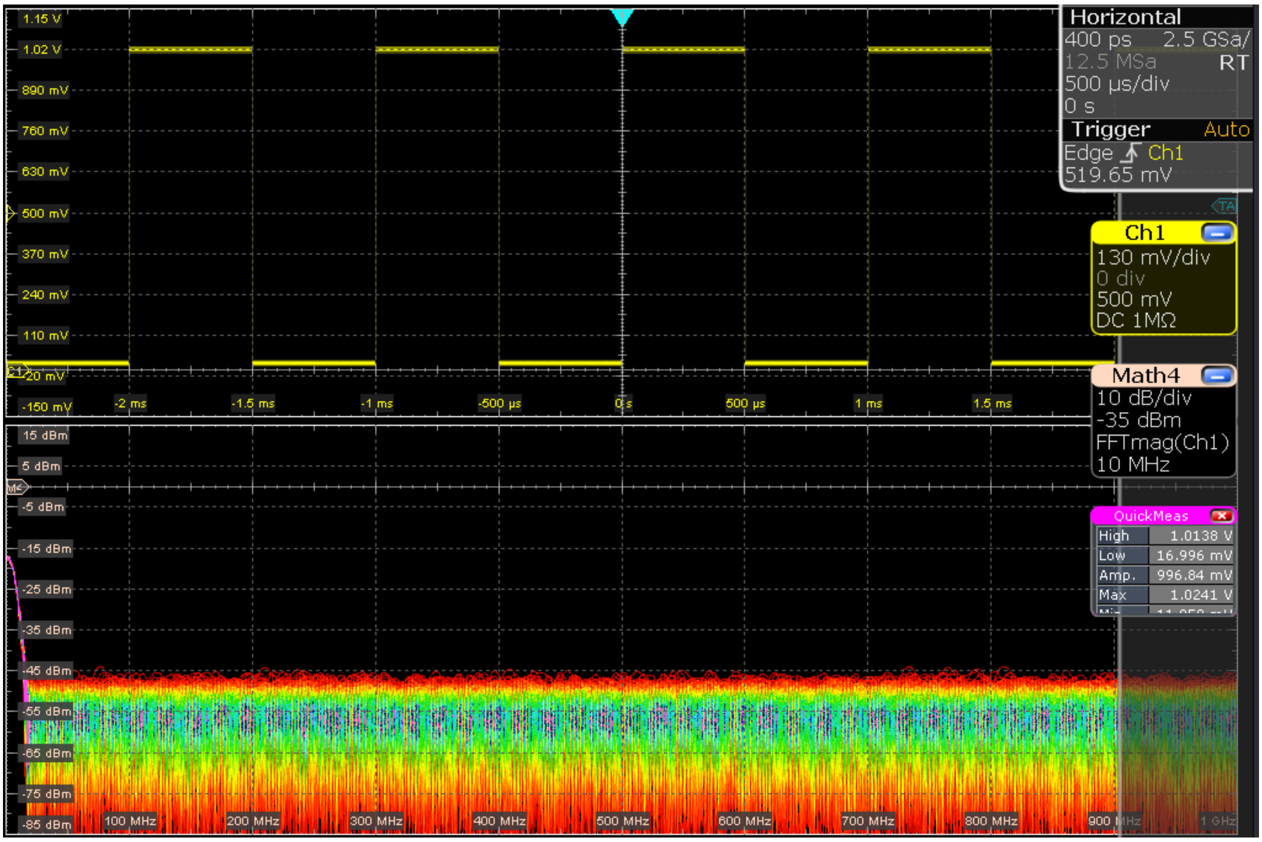
2.3.7 Performing a Basic FFT Analysis

During FFT analysis, a signal in the time domain is converted to a spectrum of frequencies. A basic spectrum waveform can be displayed quickly.

1. Restore the default signal channel settings by pressing the [AUTOSET] key.
2. Tap the "FFT" icon on the toolbar. Then tap the diagram.

   A math waveform is configured that uses the "Mag(FFT(x))" operator with Ch1 as source. The spectrum waveform is displayed in a new diagram.

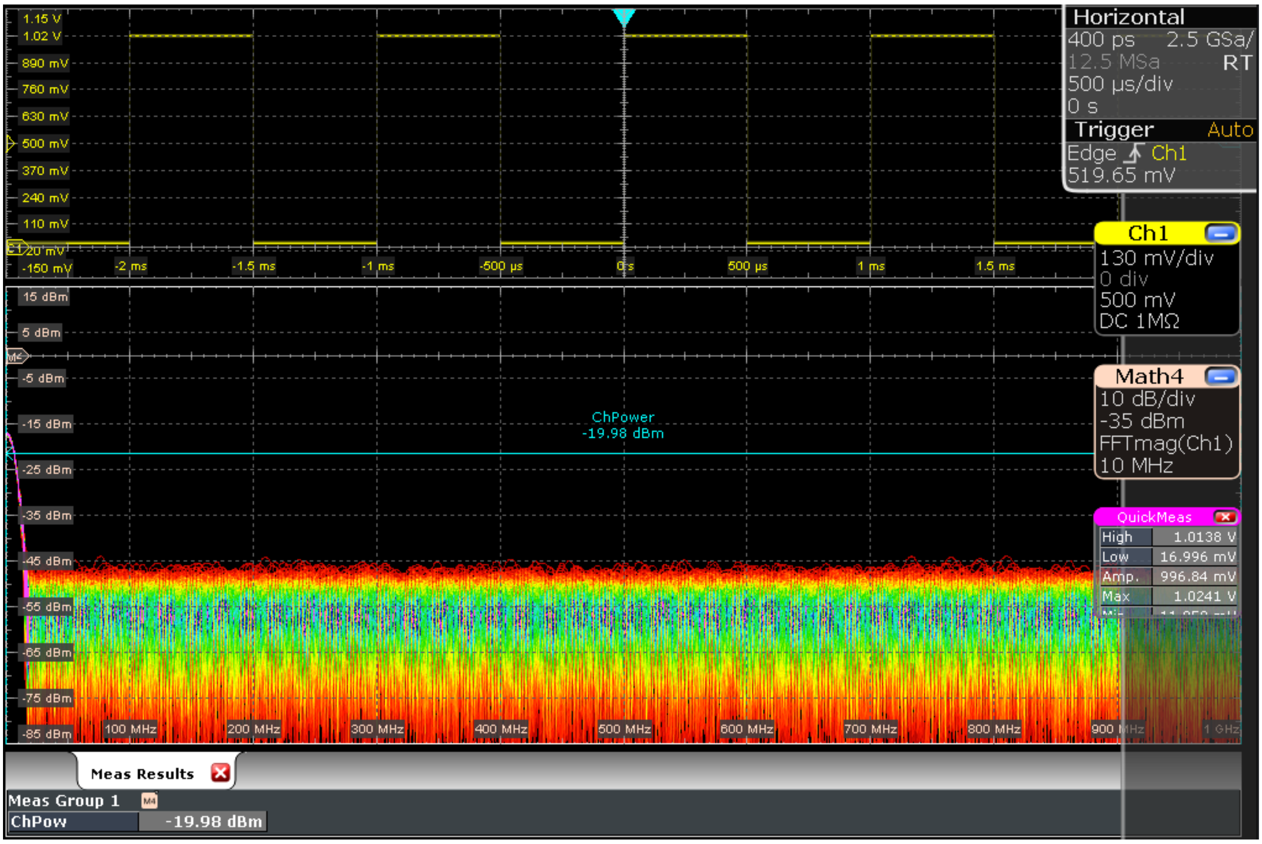


3. To measure the spectrum on the math channel, tap the "Measurement" icon on the toolbar.

4. Tap the spectrum waveform.

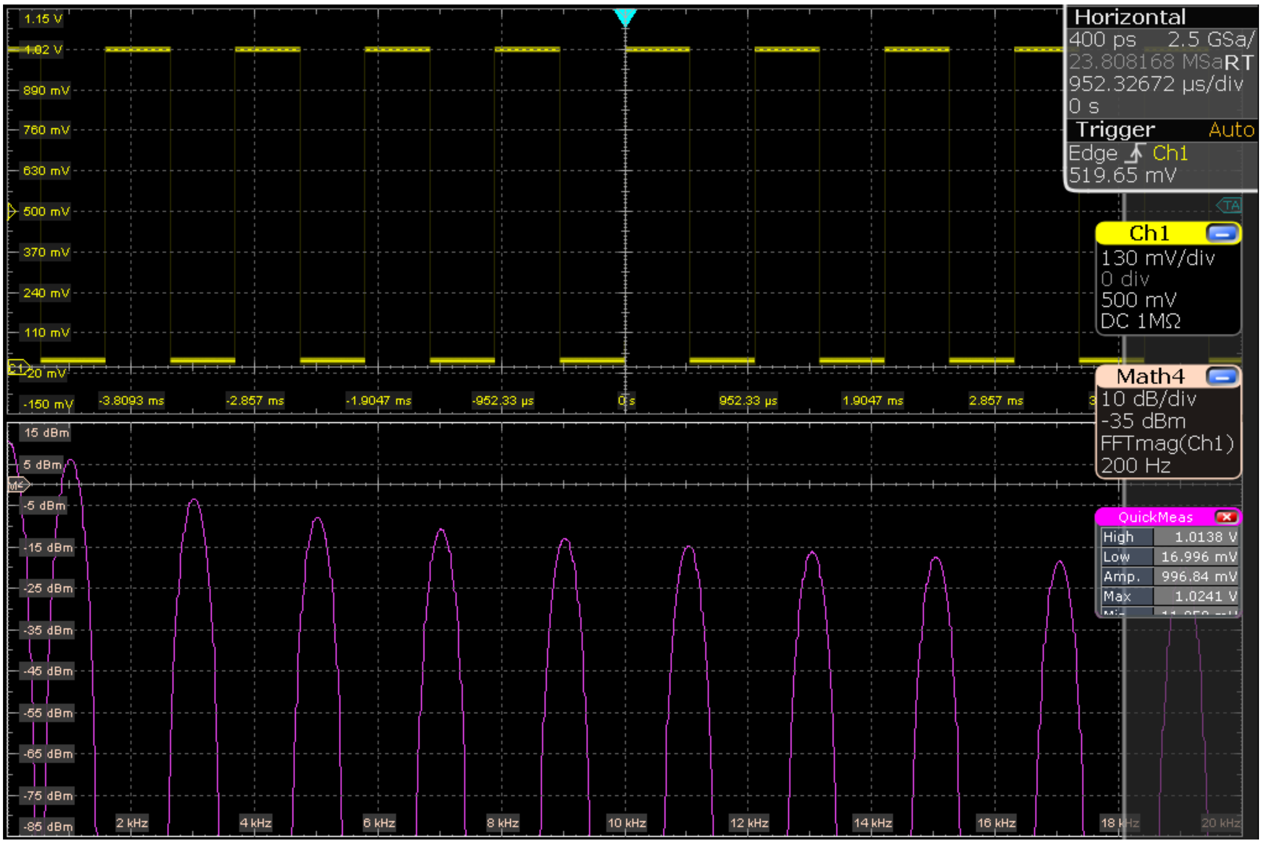
The channel power is displayed below the FFT diagram.



5. Double-tap the spectrum waveform.

   The "FFT Setup" dialog box opens.

6. Set the "Center frequency" to *10 kHz*.
   The instrument adjusts the frequency span automatically.
   Close the dialog box.



7. To remove the FFT results, tap the "Delete" icon and then the spectrum waveform.

8. Close the "Measurement" results.

2.3.8 Performing Mathematical Calculations

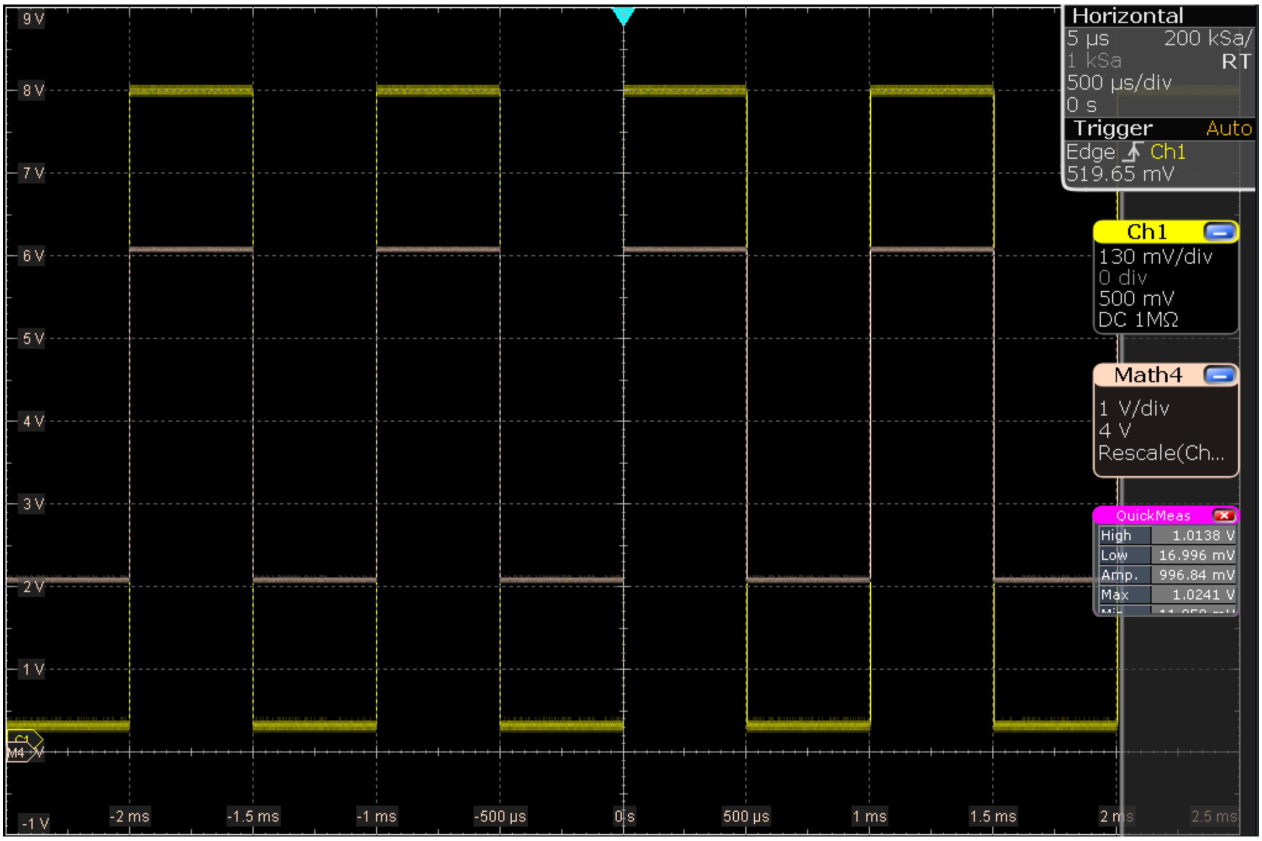
In addition to the measured waveforms, you can display calculated data to compare the current measurement result with.

For example, you can rescale the waveform and display it in the same diagram as the original waveform.

1. Press [AUTOSET].
2. Press the [MATH] key.
3. In the "Setup" tab, select the "Basic" subtab.
4. Tap the "Source1" icon and select *Ch1*.
5. Tap the "Operator" icon and select *Rescale*.
6. For "a", enter the vertical scaling factor, e.g. *4*.

7. Under "b", enter the vertical position offset, e.g. *2*.
   Look at the lower part of the dialog box and note that the instrument adjusts the "Vertical scale" and "Vertical offset" of the math waveform automatically.
8. Tap "Enable" to display the first math waveform.

   The original and the rescaled waveforms are displayed.
9. Close the "Math" dialog box.



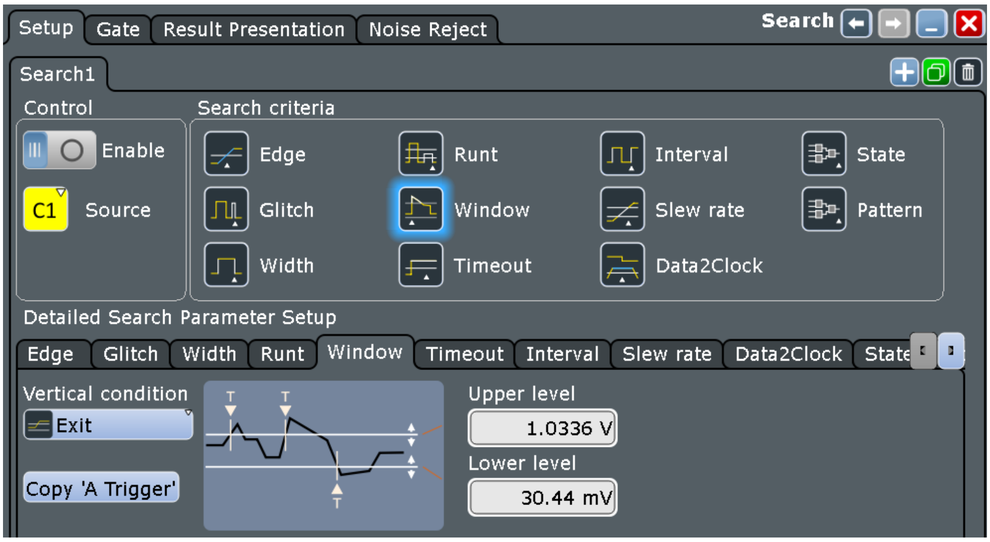
10. To remove the math waveform, do one of the following:
    - Tap the "Delete" icon and then the math waveform.
      
    - Minimize the math waveform by tapping the signal icon. Then close the signal icon

2.3.9 Performing a Search

In the following search, you detect positive and negative overshoots, i.e. values that exceed the high or low levels. To find these events, you can use the windows search.

To determine the search conditions, we use the results of the measurement example described in Chapter 2.3.6.3, "Performing and Configuring the Quick Measurement", on page 51.

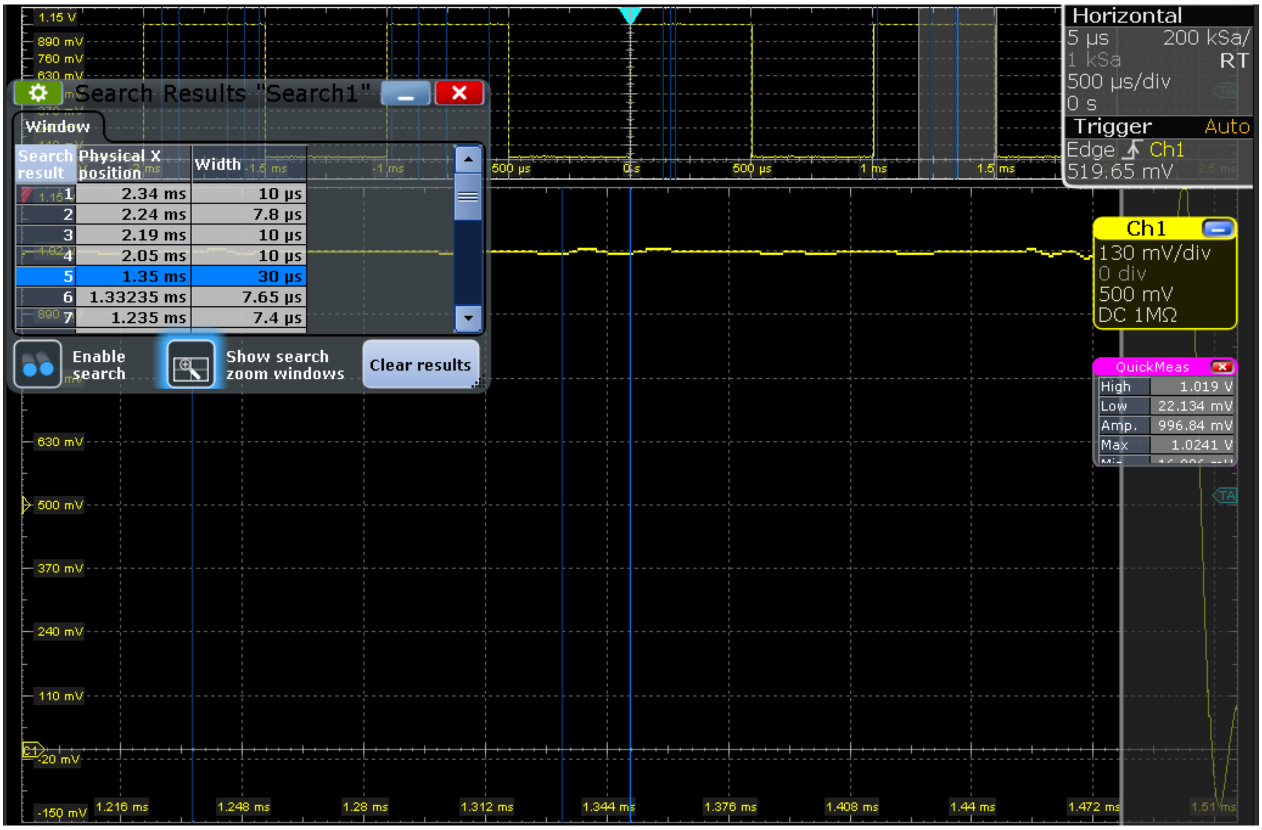
1. Press the [SEARCH] key on the front panel (ANALYSIS area).
2. Select "C1" as "Source".
3. Select the search criteria: Enable "Window".
4. In the "Window" tab below, define the search conditions:
   a) In the "Upper level" field, enter the result of the "High" level measurement.
   b) In the "Lower level" field, enter the result of the "Low" level measurement.
   c) As "Vertical condition", select "Exit" to find values that are outside the range defined by the high and low levels.



5. Select "Enable" to start the continuous search on the acquired data.
6. Close the "Search" dialog box.
7. In the "Search Results" box, select "Show search zoom windows".

   The acquisition stops, and the detected overshoots of the last acquisition are listed in the search result table. The search zoom window shows the last result that was found. Vertical lines indicate the time values for which a result was found.

8. In the results table, tap the row of the search result that you want to display in the search zoom diagram.

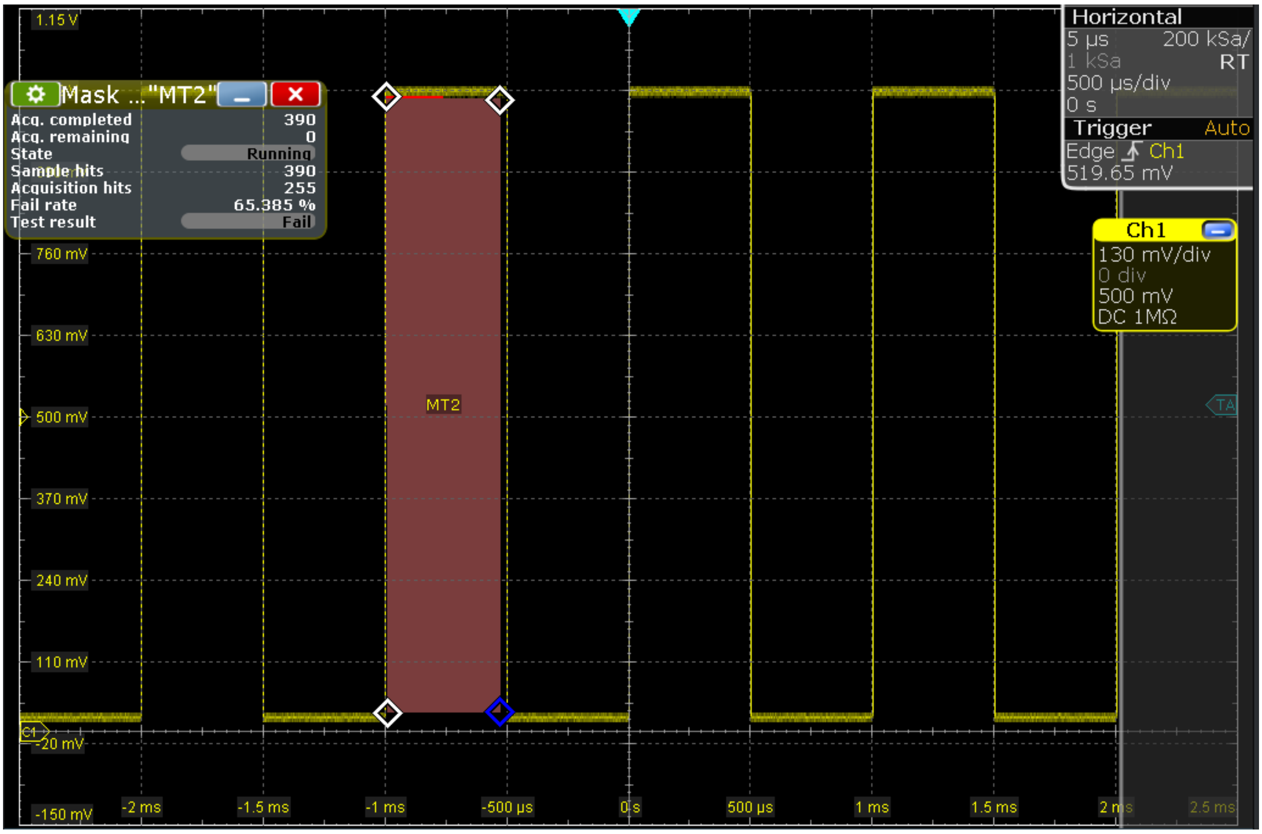


2.3.10 Performing a Mask Test

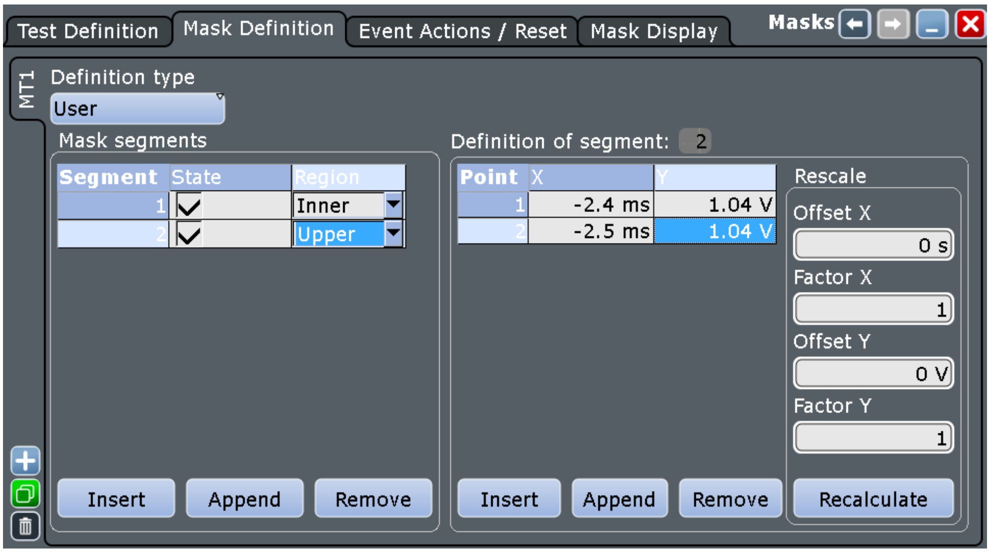
In the following example, you perform a mask test to determine whether the signal exceeds a rectangular area.

1. To restore the default signal channel settings, press [PRESET] and [AUTOSET].
2. Tap the "Masks" icon on the toolbar.

3. Tap the corner points of the first mask segment on the touchscreen. Select the corner points of one complete pulse, with a minor offset to the inside. To finish the mask definition, double-tap the last point.
To create an exact rectangle, draw the diagonal of the rectangle on the screen.



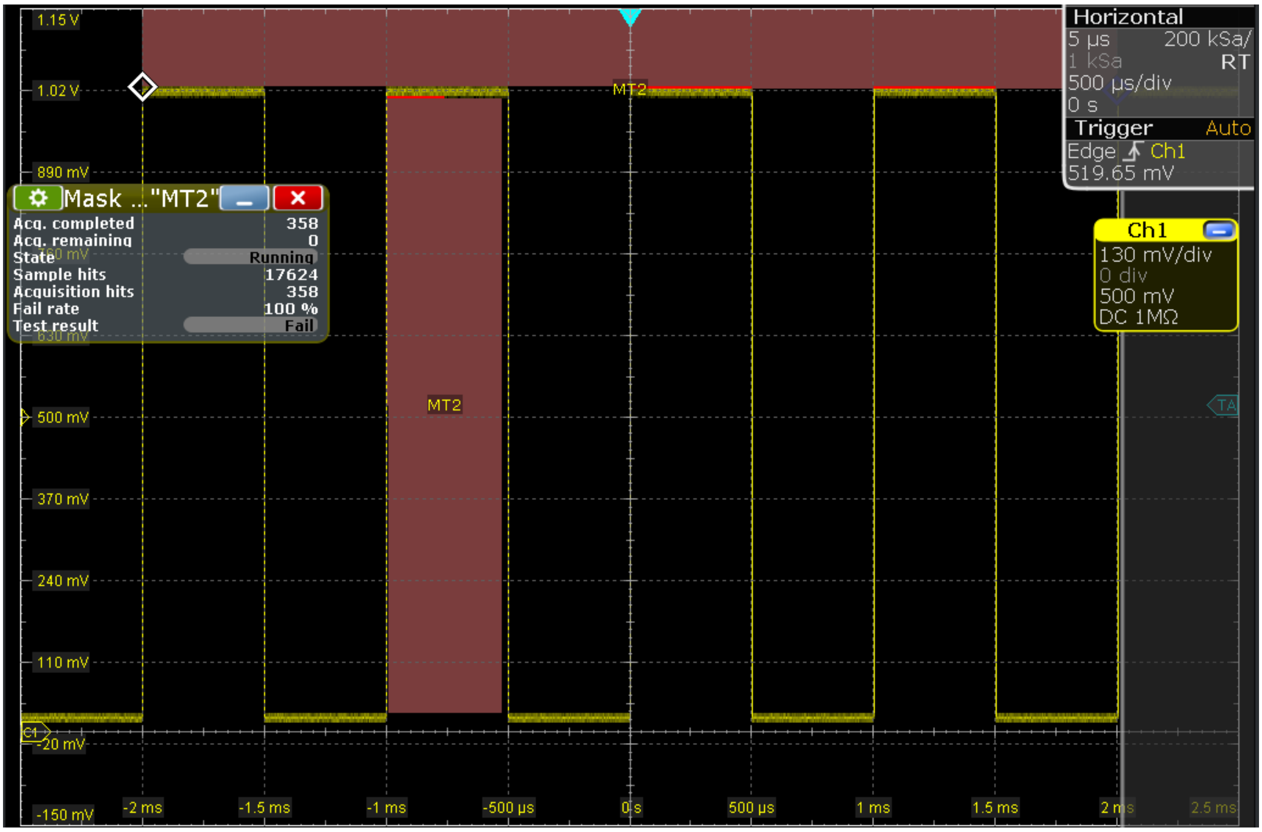
4. To define the further mask test settings, press the [MASKS] key on the front panel (in the [ANALYSIS] area), and select the "Mask Definition" tab.
5. If necessary, correct the mask segment points you defined graphically in the "Mask Definition" tab. In the "Region" column of the mask segment, "Inner" is selected. That means, a mask hit is detected if the signal is inside the segment.
6. Insert another mask segment above the positive pulse:
   a) Tap the "Append" button under "Mask segments".
   b) In the "Region" column of the new mask segment, select "Upper". In this case, a mask hit is detected if the signal is above the mask limit line.
   c) Under "Definition of segment", tap "Insert" twice to insert two points.
   d) Enter the x and y-values to define a line beneath which the values of the positive pulse should remain.



7. Select the "Test Definition" tab.
8. Select channel 1 as the "Source".
9. Define the number of tolerable sample hits in the "Tolerance" field.

   A test has failed if the number of sample hits exceeds the limit of "Violation tolerance" hits.
10. Select the "Event Action / Reset" tab.
11. For the "Stop acquisition" action, select *On violation*. If the violation tolerance is exceeded, acquisition stops.

    The results of the mask test are shown in the "MaskTest" results box. Mask hits are also indicated as red points in the mask segment in the diagram.



12. Press [RUN CONT] to start the next acquisition and watch the screen.
13. Close the "MaskTest" results box by tapping the red cross in the label.

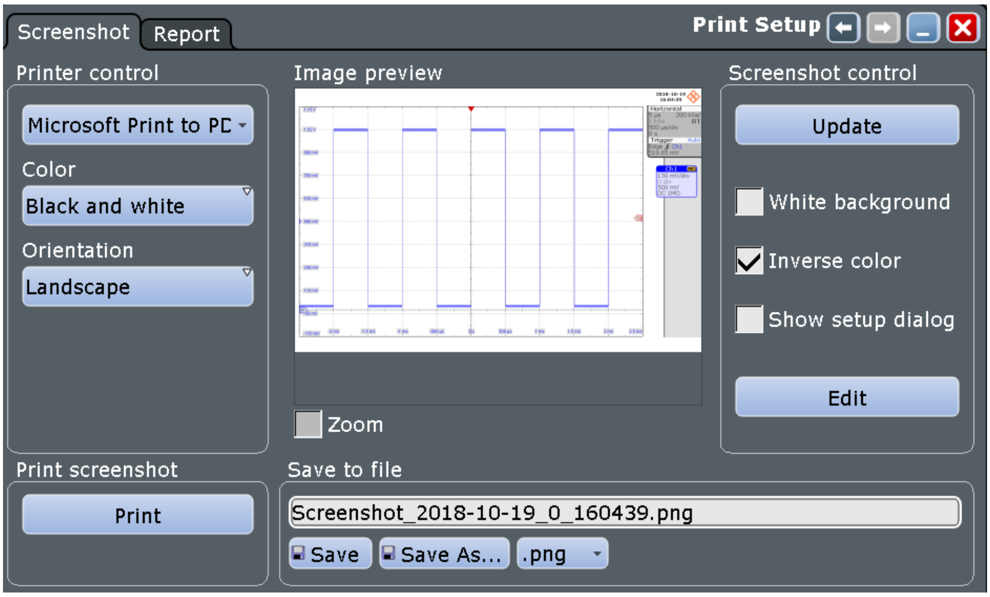
2.3.11 Printing and Saving Screenshots

You can print and save screenshots of the current display to document your results. In the following examples, you print the current display as a black and white graphic with inverted colors, i.e. a black waveform is printed on a white background. Then you save some screenshots using the PRINT key.

To print a screenshot

You need a printer that is connected to the instrument. If the instrument is connected to the network, you can also use a network printer.

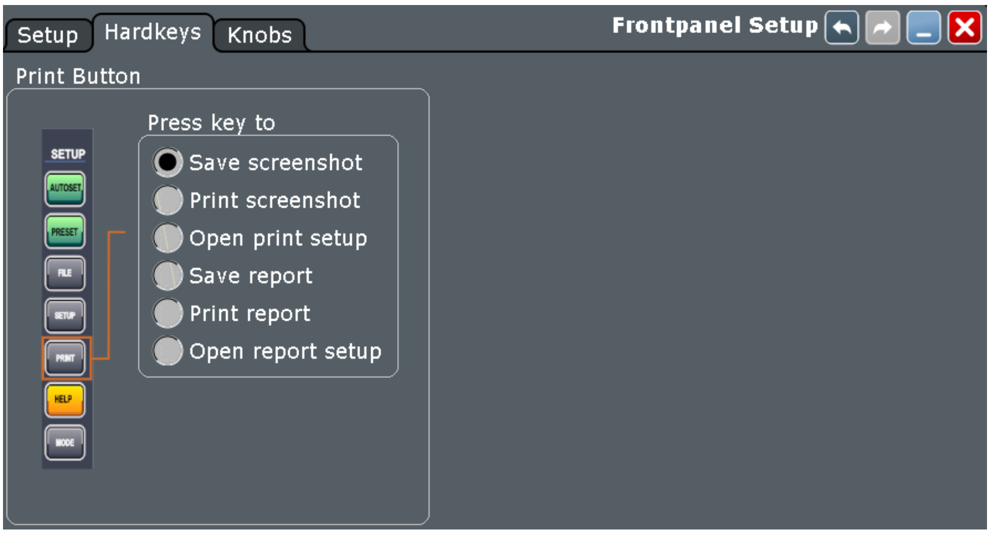
1. Open the "File" menu and tap "Print Setup".
2. Tap "Color" and select "Black and white".
3. Enable "Inverse color".



4. Tap the upper left button and select the printer.
5. Tap "Print". The result is a monochrome image.
6. Close the dialog box.

To configure the PRINT key

1. Open the "File" menu and tap "Frontpanel Setup".
2. Select the "Hardkeys" tab.
3. Select "Save screenshot".



4. Close the dialog box.

To save screenshots

1. Press the PRINT key.
2. Change the horizontal scale.
3. Press the PRINT key.

   The files are saved to the following directory:
   C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\ScreenShots
   The default file name is Screenshot_<date>_<index>_<time>.png.
4. To access the saved files, open the "File" menu and tap "Minimize Application".
5. Double-tap the "User" folder icon on the desktop.
6. Open the ScreenShots folder.

2.3.12 Saving Data

After a measurement with the R&S RTE, you can save the resulting waveform data for further evaluation or comparison. You can also save measurement results, and device settings to repeat or restore previous measurements.

- "Saving waveform data" on page 65
- "Saving data of an acquisition series" on page 66
- "Saving measurement results" on page 66
- "Saving and restoring device settings" on page 67

Saving waveform data

1. Press the [FILE] key on the front panel (in the SETUP area on the left).
2. Select "Waveform".
3. Check the "Source".
4. Set "Scope" to "Full Waveform".

   Tip: If a cursor, zoom or measurement gate is defined, you can use these settings to export only a part of the waveform.
5. Under "Save to file", tap "Save As".
6. The file selection dialog box shows the default storage directory:
   C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms
7. Tap the keyboard icon on the right of the "File Name" field.

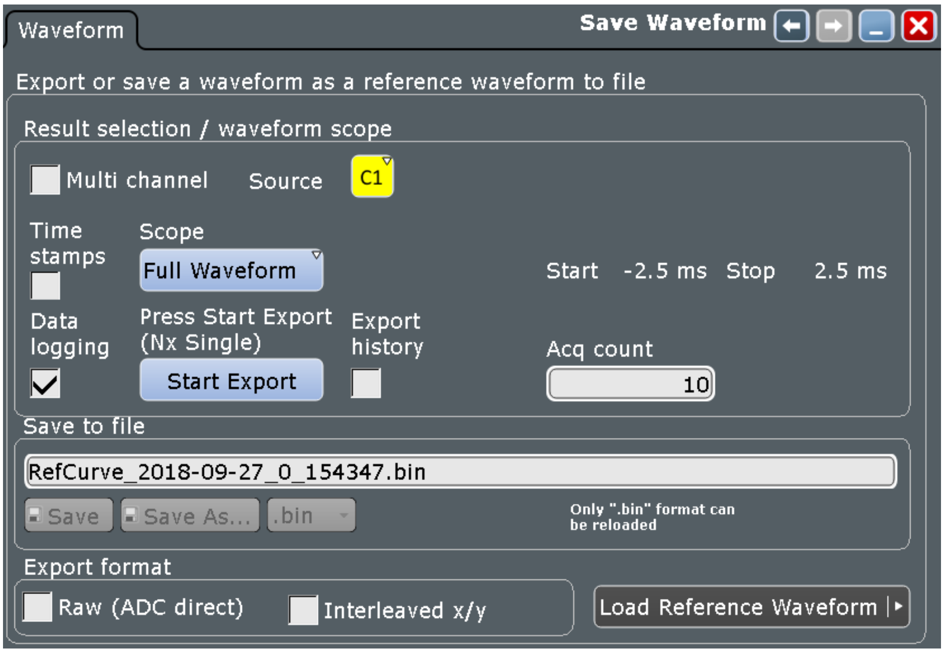
   

8. Enter *Waveform1* on the online keyboard.
9. Tap "ENTER" to close the online keyboard.
10. Select the file type: "*.bin".
11. Tap "Save".

    The waveform data is saved to the files `Waveform1.Wfm.bin` and `Waveform1.bin` in the selected directory.

Saving data of an acquisition series

1. Press the [FILE] key on the front panel.
2. Select "Waveform".
3. Set the export scope of the waveform:
    a) Check the "Source".
    b) Set "Scope" to "Full Waveform".
    c) Tap "Data logging" to enable export all waveforms of a running acquisition.
    d) Enter "Acq count" = *10*, the number of subsequent waveforms that the instrument acquires and saves.



4. Tap "Start Export" to save the waveforms to file. You can change the filename in "Save to file".

Saving measurement results

1. Perform a measurement as described in Chapter 2.3.6.2, "Performing Automatic Amplitude Measurements", on page 49.
2. Press the [FILE] key on the front panel.

3. Select "Numeric results".
4. Select the results that you want to save.
5. For further usage of the results, select the "CSV-Delimiter" that is used to convert the values in columns. For MS Excel, select the semicolon.
6. Tap "Save".

   The results are saved to the following folder:
   `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\ResultBoxExport`
   The filename is created according to the autonaming settings.

Saving and restoring device settings

1. Press the [FILE] key on the front panel.
2. Select "Savesets".
3. Tap "Save As". Enter the path and filename.
   `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\SaveSets\Settings_Meas1.dfl`
4. Tap "Save".
5. To restore the default instrument settings, press the [PRESET] key.
6. To repeat the initial measurement, tap the "Load saveset" icon on the toolbar.

   
7. Select the "Type" = "Savesets".
8. Use the buttons on the left and the right to scroll the stored savesets. The filename and a screenshot help identify the correct saveset.
9. Tap "Load".

   The device and measurement settings are restored and you can repeat the measurement.

2.4 Operating the Instrument

There are three ways to operate the R&S RTE.

Manual operation

Use the touchscreen, keys and rotary knobs, or an optional mouse and/or keyboard. The principles of manual operation are explained in this section.

Remote control

Create programs to automatize repeating settings, tests, and measurements. The instrument is connected to a computer that runs the program.

This way of operation is described in: Chapter 17, "Remote Control Commands", on page 1016

Remote operation

The remote desktop connection of Windows can be used for instrument control and file transfer. Even on computers with non-Windows operating systems, a remote desktop connection is possible using RDP applications.

For details, refer to the user manual, chapter "Remote Desktop Connection".

Remote monitoring and control of the instrument from a connected computer is also possible with a standard web browser and a LAN connection.

For details, refer to the user manual, chapter "Web Control".

Alternatively, you can use Virtual Network Computing (VNC), which requires installation of the VNC server on the R&S RTE. Installation and configuration are described in the application note "Remote Monitoring and Control of the R&S RTE with a Web Browser", available on the Rohde & Schwarz internet site.

2.4.1 Means of Manual Interaction

The R&S RTE provides the following means of manual interaction, which you can use alternatively or complementary:

- Touchscreen:
  Using the touchscreen is the direct interaction way. Use your finger to place waveforms on the screen, mark areas for zoom and histograms, set parameters in dialog boxes, enter data, and much more. The control elements and actions on the screen are based on common concepts, and you will easily become familiar with the user interface.

  Tapping the screen works like clicking mouse buttons:
  - Tap = click: Selects a parameter or provokes an action.
  - Double-tap = double-click has the same effect as touch and hold = right-click: Opens the on-screen keyboard or keypad, or a specific editor if available
- Function keys and rotary knobs:
  The front panel provides nearly all functions and controls to operate the instrument in the classic ways, without touchscreen. As an exception, the signal bar cannot be used with front panel controls.
- Optional mouse and/or keyboard:
  These devices work conform to Windows standards. The navigation keys on the front panel correspond to the keys on the keyboard.

The usage of the touchscreen and navigation keys is described in detail in the following sections.

2.4.2 Touchscreen Display

2.4.2.1 Information on the Display

The touchscreen of the instrument shows the waveforms and measurement results, and also information and everything that you need to control the instrument. All waveform-related display elements are shown in Figure 2-3. An overview of control elements - like dialog box, toolbar - is given in Figure 2-6 .

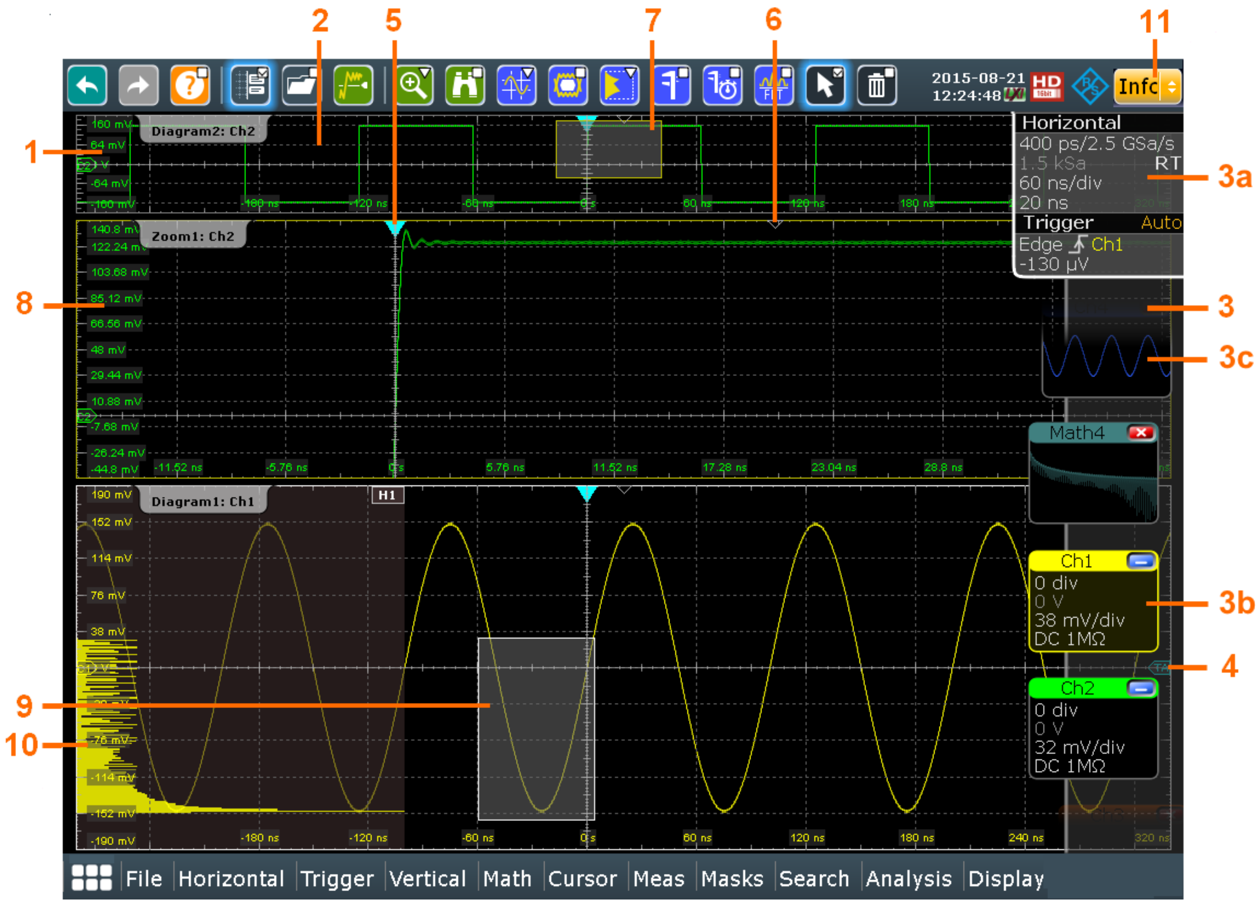


Figure 2-3: Display information

1 = Diagram
2 = Grid
3 = Signal bar with horizontal and trigger label (3a), signal icon with waveform settings (3b) and signal icon with minimized live waveform (3c)
4 = Trigger level
5 = Trigger position
6 = Reference point (distance from trigger position to reference point = horizontal position)
7 = Zoom area
8 = Zoom diagram
9 = Histogram area
10 = Histogram
11 = Messages

Diagram (1)
A diagram shows one or more waveforms: channel, reference, and math waveforms together with histograms, masks etc. Zoom details, XY-waveforms, spectra and other special waveforms are shown in separate diagrams.

By default, the diagram name contains the diagram number and the short names of the waveforms shown inside. To change the diagram name, touch and hold the tab name. The on-screen keyboard opens to enter the new name. Names must be unique.

To arrange the diagrams on the screen, the Rohde & Schwarz SmartGrid function helps you to find the target place simply and quickly. For details, see Chapter 2.4.5, "Rohde & Schwarz SmartGrid", on page 76. You can also adjust the diagram size by dragging the diagram border.

Grid (2)
The grid shows the vertical and horizontal divisions. The division lines are labeled with the correspondent values. The grid labels have the color of the waveform to which they belong. If several waveforms are shown in one diagram, the grid has the color of the selected waveform.

Signal bar (3)
The signal bar is the control center for all enabled waveforms. On the top, the horizontal and trigger labels show the main timebase and trigger settings. If you tap a label, the relevant dialog box opens with the tab used at last.

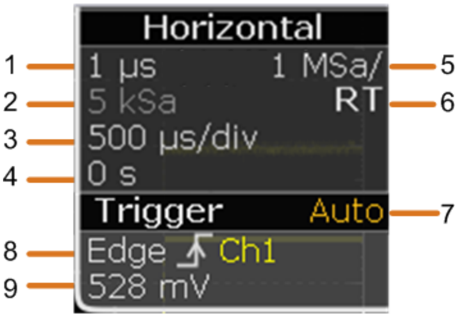


Figure 2-4: Horizontal and trigger label on top of the signal bar

1 = Resolution
2 = Record length
3 = Timebase, horizontal scale
4 = Horizontal position
5 = Sample rate
6 = RT - real time, IT - interpolated time
7 = Trigger mode
8 = Trigger type, slope, and source
9 = Trigger level

Below, each waveform is represented by a signal icon. If the waveform is shown in a diagram, the signal icon displays its main vertical and acquisition settings. If you tap the "Minimize" icon on the signal icon, the waveform switches from the diagram area to the signal icon: the icon shows the real-time preview of the waveform. If you tap a signal icon, the dialog box with vertical settings for this waveform opens. See Chapter 2.4.4, "Working with Waveforms", on page 74 for a detailed description.

In Figure 2-3, the signal icons Ch1 and Ch2 show the main settings, and the waveforms are displayed in diagrams. All other waveforms are minimized and shown in the signal icon.

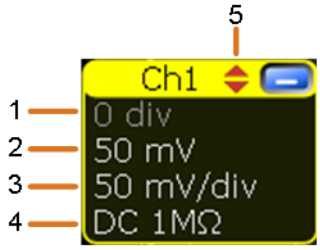


Figure 2-5: Signal label on the signal bar

1 = Vertical scale
2 = Vertical position
3 = Offset
4 = Coupling
5 = Clipping indication is shown if the signal is clipped at the top and/or bottom.

You can also adjust the behavior of the signal bar in various ways, see Chapter 2.4.6, "Using the Signal bar", on page 78.

Trigger position and trigger level (4, 5)
The blue markers show the horizontal position of the trigger and the vertical trigger level. You can touch and move the trigger markers in the diagram to set the positions. The trigger point is the zero point of the diagram.

The trigger position can be moved outside the diagram. A red trigger position marker indicates that the trigger position is not visible.

Reference point (6)
The reference point marks the rescaling center. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.

You can define the position of the reference point (HORIZONTAL), and its time distance from the trigger point of the diagram (POSITION / REF POINT).

Zoom diagram and zoom area (7, 8)
Zoomed waveforms are shown in separate zoom diagrams, in addition to the waveform diagrams. On the original waveform diagram, a rectangle indicates the zoomed section of the waveform - this is the zoom area. You can modify the zoom area by dragging the rectangle as a whole, and by dragging its edges. To toggle between these modes, tap the zoom area. You can also set exact positions.

The frames of the zoom area and of the associated zoom diagram have the same color, different zooms are marked with different colors. So it is easy to assign zoom area and zoom diagram.

As for waveform diagrams, you can change the name of the zoom diagram. A zoom in a zoom and coupled zooms are also possible.

For details, see Chapter 6.1, "Zoom", on page 240.

Histogram and histogram area (9, 10)
A histogram shows the frequency of occurrence of voltage or time values in a bar chart directly in the diagram. The rectangular histogram area indicates the part of the wave-form that is considered in the histogram. The vertical histogram counts the voltage val-ues, and the horizontal histogram counts time values. You can switch between vertical and horizontal mode, and modify the histogram area by dragging the rectangle as a whole, by dragging its edges, or by setting exact positions.

Messages (11)
A yellow or red button on the toolbar points to the status messages of the instrument. To open the message box, tap the button. See also: Chapter 2.4.11, "Messages", on page 91.

2.4.2.2 Control Elements on the Touchscreen

The touchscreen provides everything you need to control the instrument, to analyze waveforms, and to get measurement results. Figure 2-6 shows the control elements on a glance.

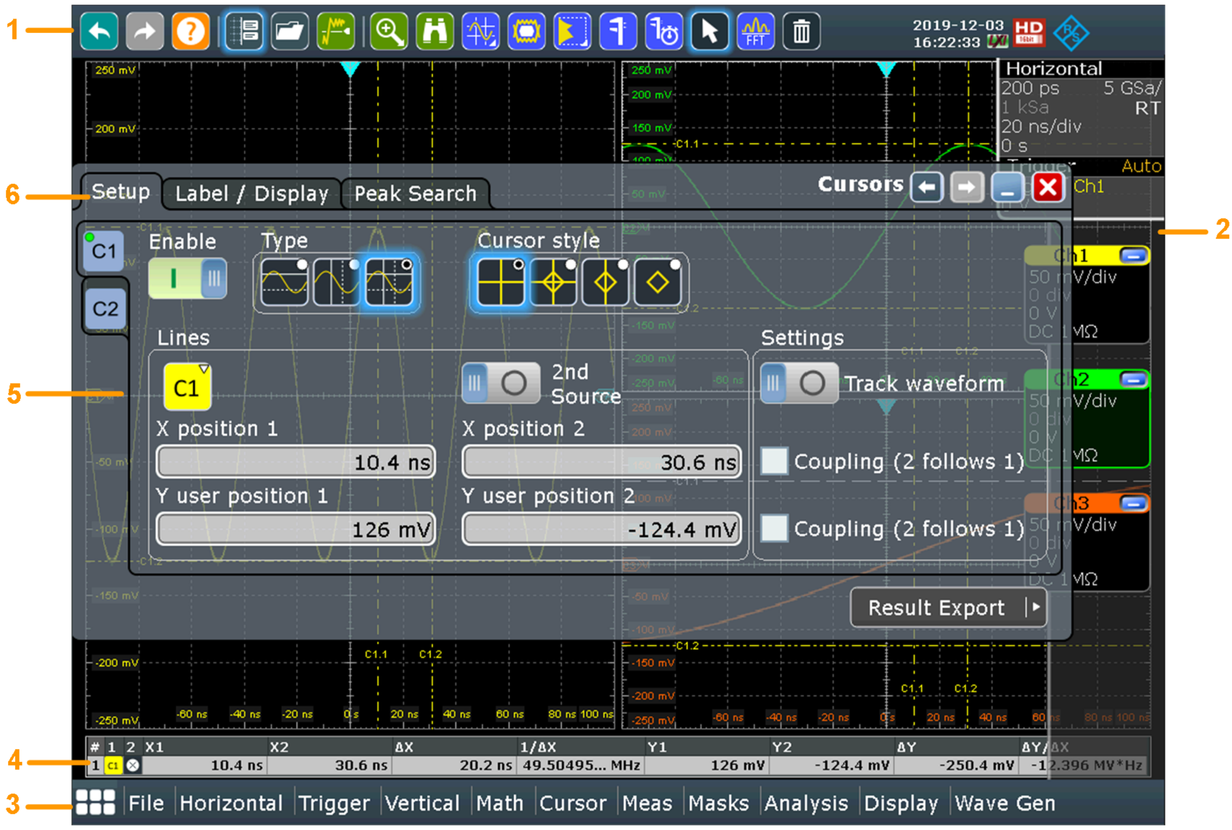


Figure 2-6: Control elements on the touchscreen

1 = Toolbar
2 = Signal bar
3 = Menu bar
4 = Result box (docked)

5 = Dialog box
6 = Tab in a dialog box
Not shown = Input box

Toolbar (1)
The icons on the toolbar provide quick and easy access to the most important functionality. For a detailed description, refer to Chapter 2.4.7, "Toolbar", on page 79.

Signal bar (2)
The signal bar shows all enables waveforms as described in "Signal bar (3)" on page 70.

Menu bar (3)
The menus provide access to the complete functionality of R&S RTE.

Result box (4)
If you perform cursor or automatic measurements, mask testing, or a search, the result box shows the results of the action. The position of the result box is adjustable. It can be docked (default for measurements), floating, minimized to a result icon on the signal bar, or displayed in a separate diagram on the screen.

For details, see Chapter 2.4.8, "Displaying Results", on page 85.

Dialog box (5, 6)
The tabs of the dialog boxes contain all task-oriented settings and operations, and black buttons for calling related tabs. The usage of dialog boxes is described in Chapter 2.4.9, "Using Dialog Boxes", on page 87.

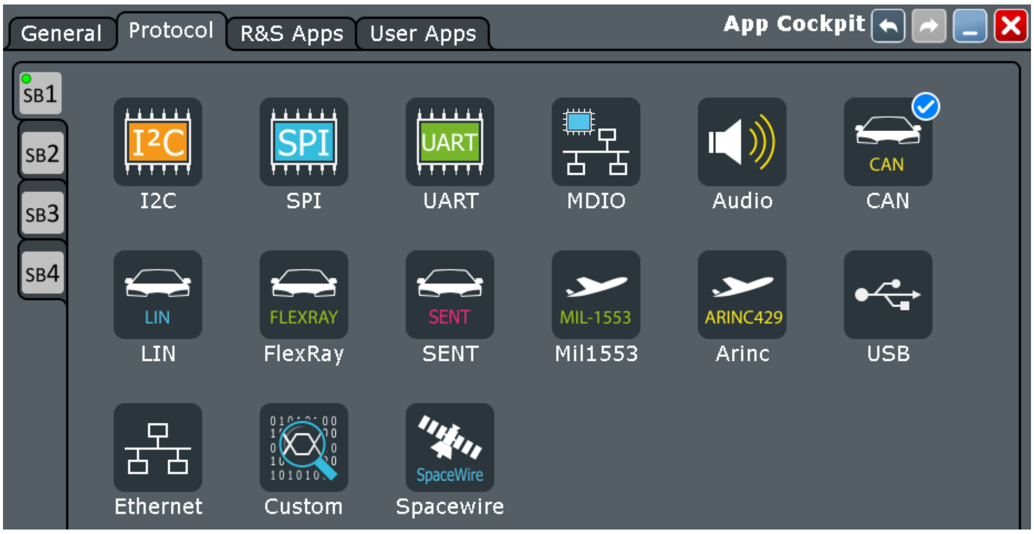
Input box
The input box appears if you adjust a value using one of the rotary knobs, or if you drag an element on the screen, for example, a cursor line. The input box shows the current value of the modified parameter. You can enter the exact numerical value, change the step size, and - if available - autoset the value directly in the input box. The box title shows the name of the currently adjusted parameter. The input box is helpful when using the multi-function rotary knobs, for example, [INTENSITY], and [RESOLUTION / RECORD LENGTH].

See also: Chapter 2.4.10, "Entering Data", on page 89.

2.4.3 App Cockpit

The app cockpit provides fast access to all available applications.

► To open the app cockpit, tap in the menu.



2.4.4 Working with Waveforms

The R&S RTE can create and display several types of waveforms:

- Channel waveforms:
  For each input channel, one channel waveform is shown.
- Reference waveforms:
  Four waveforms can be used as reference for comparison and analysis.
- Math waveforms:
  Eight mathematic waveforms can be created with mathematic operations performed on channel, reference, and other math waveforms.
- Zoom waveforms:
  Show the details of waveforms.
- XY-waveforms:
  Four XY-waveforms can be created. Each XY-waveform is built from the voltage values of two source waveforms.
- Digital waveforms:
  The Mixed Signal Option R&S RTE-B1 provides 16 digital channels grouped in two logic probes (pods) with 8 channels each.

Waveform handling

The R&S RTE can show and analyze many waveforms. To handle this multitude while keeping track of it, the R&S RTE provides intelligent support:

- The color system helps to distinguish the waveforms. The color of the vertical rotary knobs indicates the signal that is focused (selected). The color of each waveform can be changed, the color of its signal icon and of the illuminated keys is adjusted to the new color. Alternatively, a color table can be assigned to a waveform. Settings: [DISPLAY] > "Signal Colors / Persistence" tab.
- Waveforms can be minimized to signal icons showing a small real-time signal view. Thus, more space in the diagram area is available without switching off waveforms.

- Diagrams are displayed on tabs – you can arrange them side by side or one above the other. To change the diagram name, double-tap the tab name.
- The Rohde & Schwarz SmartGrid function helps to arrange the diagrams.

Waveform states

Depending on its place on the screen and the effect of settings, a waveform has one of the following states:

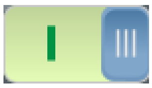
- Off
- Active:
  The waveform is shown in a diagram
- Selected:
  One of the active waveforms that has the focus. In each diagram, one of the assigned waveforms is selected – it appears "on top" in the diagram, and the grid labels have the color of the selected waveform. Some of the toolbar functions, like cursor and histogram measurements are performed on the selected waveform. All waveform-specific settings are applied to the selected waveform of the selected diagram. The vertical [POSITION / OFFSET] and [SCALE] knobs, and the [SIGNAL OFF] key are illuminated with the color of the selected waveform.
  In Figure 2-3, "Ch1" is the selected waveform: The frame of the diagram and the signal icon are highlighted.
- Minimized:
  The waveform is shown as real-time signal view in its signal icon.

To switch a waveform on

A channel waveform is activated as soon as you connect the probe. You can switch it on and off according to your needs.

► Choose one of the following ways:

- Press the channel key.
- In the "Vertical" dialog box, select the channel and tap the "Show channel" button.

The waveform is now active, selected, and is shown in the diagram.

To select a waveform

► Choose one of the following ways:

- Tap the waveform in the waveform diagram.
- To select a channel, reference, or math waveform, press the corresponding key.
- Tap the signal icon.

Note: Zoom waveforms in zoom diagrams cannot be selected.

To minimize a waveform

► Choose on of the following ways:

- Tap the "Minimize" icon in the upper right corner of the waveform's signal label in the signal bar.
- Drag the waveform from the diagram to the signal bar.

The waveform disappears from the diagram and the minimized signal view is shown in the signal icon.

Tip: To set the waveform back to its previous diagram immediately, use "Undo".

To arrange a waveform using the SmartGrid

See Chapter 2.4.5, "Rohde & Schwarz SmartGrid", on page 76.

To switch off a waveform

► Do one of the following:

- Select the waveform, and then press the [SIGNAL OFF] key.
- To switch off a minimized waveform, tap the "Close" icon in the upper right corner of the minimized signal view.
- Disable "Show channel" in the "Vertical" > "Channels" tab.
- Tap the "Delete" icon (Recycle bin) in the toolbar, and then the waveform.
  If several waveforms overlap or lie close together, the upper (selected) waveform is switched off.

2.4.5 Rohde & Schwarz SmartGrid

The Rohde & Schwarz SmartGrid helps to create and arrange the diagrams on the screen with drag&drop. The diagram layout depends on the position where you drop the signal icon, in relation to an existing diagram.

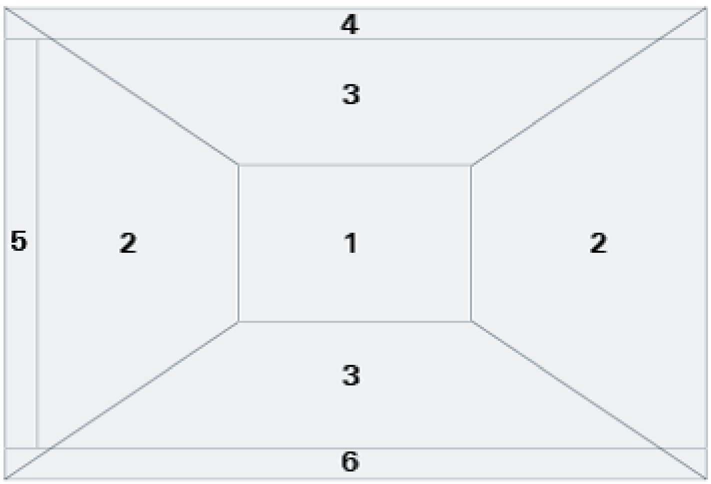


Figure 2-7: SmartGrid positions

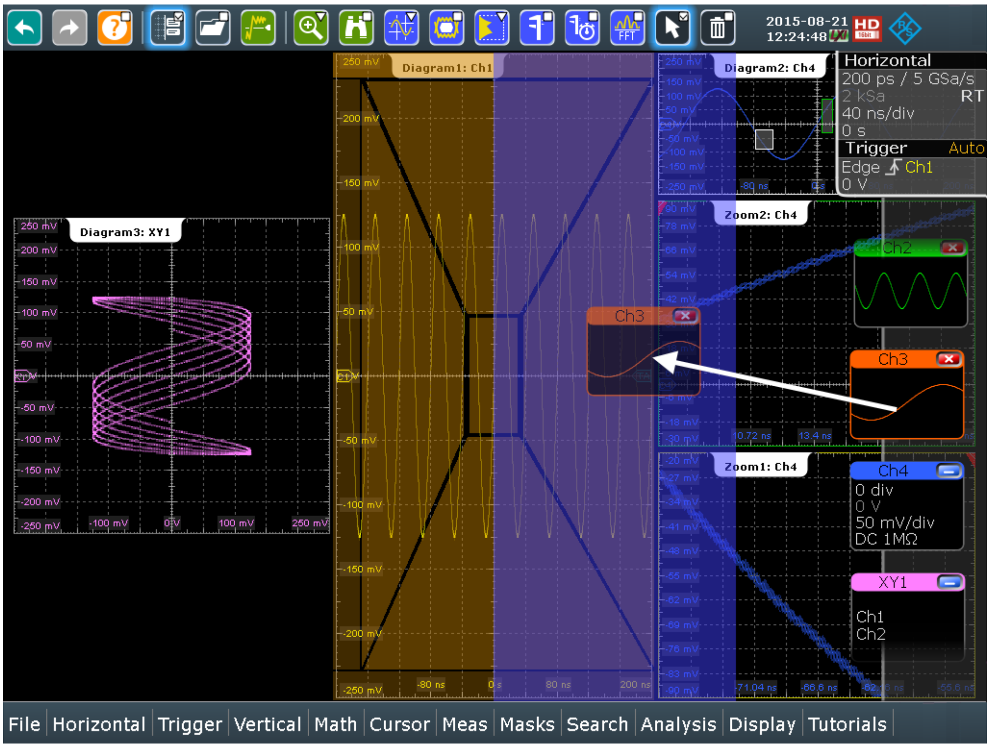
1 = In the existing diagram, overlay of signal
2 = New diagram on the left or right
3 = New diagram above or below
4 = New diagram on top of the existing diagram
5 = XY-diagram
6 = YX-diagram

The diagram configuration is deleted when you use [PRESET] and `*RST`.

To arrange a waveform using the SmartGrid

You can arrange waveforms in one of the existing diagrams, or in a new diagram.

1. Drag the signal icon to the diagram area, and move it around.
   The Rohde & Schwarz SmartGrid appears and a blue area shows where the waveform will be placed.
2. Drop the waveform in the target area.
   The waveform appears in an existing or in a new diagram, and it is selected for further actions.



3. To change the size of the new diagram, drag its edge to the required position.

Remote commands: see Chapter 17.7.2.3, "SmartGrid", on page 1057.

2.4.6 Using the Signal bar

The signal bar can hold a large number of signal and result icons. Signal icons represent the waveforms, serial buses and parallel buses, while result icons are minimized result boxes showing measurement and search results.

To scroll the signal bar

If the signal bar contains more than four icons, not all icons are visible on the display.

► Touch one of the signal icons and move it up or down until the required icon appears.

To switch on and off the signal bar

If you need the complete screen to see the diagrams and results, you can switch off the signal bar completely.

► Tap the "Show signal bar" icon on the toolbar.

Alternatively, tap "Signal Bar" on the "Display" menu.

To change the position of the signal bar

► Touch the "Horizontal" label on the top of the signal bar and drag it to the opposite side of the screen.

To configure auto-hide

The signal bar can be hidden if the displayed information has not changed for a defined time, and is displayed again automatically when a setting in the signal bar changes. The signal bar does not hide entirely, it simply fades and becomes less visible in the display.

1. Press the [DISPLAY] key on the front panel.
2. In the "Display" dialog box, select the "Diagram Layout" tab.
3. Select "Auto-hide".
4. Define the hiding properties:
   - "Hide bar after": the time after which the bar is hidden if no changes occur
   - "Hiding transparency": Transparency of the hidden signal bar on a scale from 20% (low transparency) to 70% (high transparency)
   - Hide head also: the horizontal and trigger labels are also faded

To change the colors

If you want to highlight the signal bar, you can change the "Fill color" and "Border color" of the bar.

1. Press the [DISPLAY] key on the front panel.
2. In the "Display" dialog box, select the "Diagram Layout" tab.
3. Tap "Border color" to change the color of the signal bar frame, or "Fill color" to change the fill color of the bar.
4. In the "Adjust Colors" dialog box, select the color.
5. To use a color that is not yet defined, tap "Userdefined Colors". Define the new color settings.
   To see the effect of a setting change in the "Preview" area, enter the value and press the [ENTER] key.
6. Tap "OK."

   The signal bar is displayed in the new colors.

2.4.7 Toolbar

The toolbar provides direct access to important control and measurement functions. It shows current date and time, and a message button. The selected function is highlighted.

A little triangle in the lower right corner of the icon means, that a menu is available where you can select the required function.

By default, the toolbar shows the most frequently used functions. You can configure the content of the toolbar and hide the date/time display, see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.

2.4.7.1 Using the Toolbar

Using the toolbar is easy and straightforward.

Some of the toolbar functions are one-click actions. These actions are performed immediately when you tap the icon.

Other toolbar functions are analyzing functions. These actions are interactive actions.

To use analyzing functions (interactive actions)

1. If several waveforms are shown in the diagram, select the waveform that you want to analyze.
   See: "To select a waveform" on page 75
2. Tap the icon of the function in the toolbar.

3. To define the analyzed area, do one of the following:
   - Tap the required diagram.
   - Drag a rectangle on the diagram.

To select a function on a toolbar menu

Icons with a little triangle in the lower right corner show the last selected function. A short tap on the icon activates the displayed function. To change the function, proceed as follows:

1. Touch the icon, and drag your finger down.
2. When the menu has opened, remove the finger.
3. Tap the required function on the menu.
   The function is selected, and its icon is shown in the toolbar.

2.4.7.2 Configuring the Toolbar

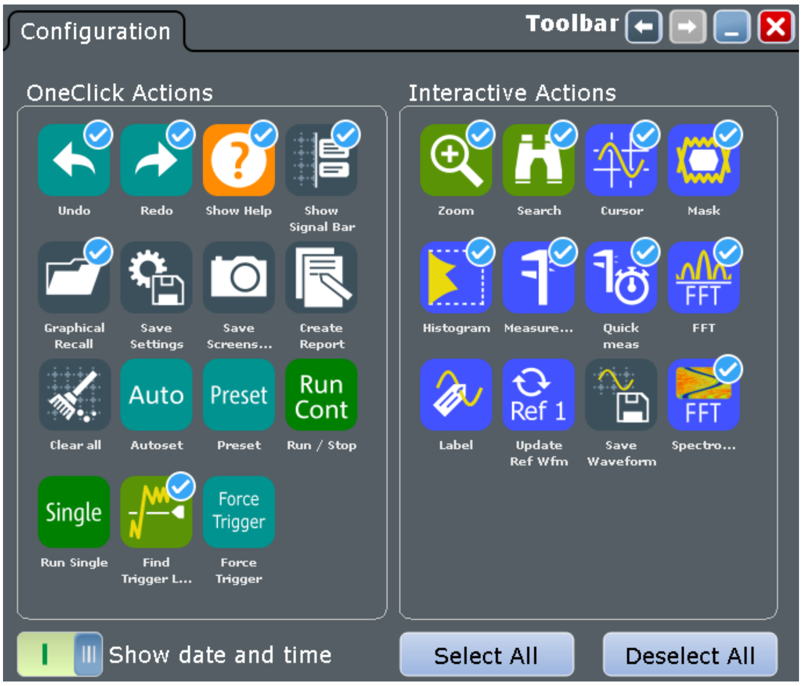
You can configure the contents of the toolbar so that only the required functions are displayed. Furthermore, date and time can be hidden. The toolbar configuration is part of the user preferences. It is retained when you switch off and on the instrument, and you can save it in the user preferences and user-defined preset.

1. On the "Display" menu, select "Toolbar".
2. Select the required toolbar functions:
   - Disable the functions that you do not need.
     Enable the functions that you want to add to the toolbar.
   - To display all available toolbar icons, tap "Show All".
   - To hide all toolbar icons, tap "Hide All".
3. To hide the current date and time on the toolbar, disable "Show date and time".

A detailed description of the toolbar functions is given in Chapter 2.4.7.3, "Toolbar Functions", on page 80.

2.4.7.3 Toolbar Functions

This chapter describes all toolbar functions in detail.



One-click actions	Interactive actions
Undo	Zoom
Redo	Search
Show Help	Cursor
Graphical Recall (load saveset)	Masktest
Save Settings	Histogram
Save Screenshot	Measurement
Create Report	Quick meas
Clear all	FFT
Show Signal Bar	Select
Autoset and Preset	Delete
Run / Stop and Run Single	Label
Find Trigger Level	Update Ref Waveform
Force Trigger	Save Waveform
	Spectrogram (option R&S RTE-K18)

You can configure the content of the toolbar and hide the date/time display, see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.

The following list describes at first the default toolbar functions and then the additional functions.

Undo
Undoes the last setting actions step by step. Some actions cannot be revoked: locking the touchscreen with [T-SCREEN LOCK], and saving data. The undo stack is deleted during the following actions: Reloading settings from file, and reference waveform actions (save, load and preset with active reference waveform).

Redo
Recovers the undo steps in reverse order.

Show Help
Enables the tooltip display. A short description appears when you tap a parameter in a dialog or result box. To open the corresponding help topic, tap the "Show Help" button in the lower right corner of the tooltip. See also: Chapter 2.4.12, "Getting Information and Help", on page 91.

Show Signal Bar
Shows and hides the signal bar.

The look and the behavior of the signal bar can be configured, see Chapter 2.4.6, "Using the Signal bar", on page 78.

Graphical Recall (load saveset)
Opens a window to select and load instrument settings that were previously stored in a saveset. A graphical preview helps you to find the required settings.

Zoom
The zoom icon on the toolbar shows the last selected zoom type. A short tap on the icon activates the selected zoom.

If you touch the icon and drag your finger down, a menu opens where you can select another zoom type.

Standard zoom ← Zoom
Displays a magnified section of the diagram in an additional zoom diagram. It is a display zoom, instrument settings are not changed.

Touch and hold the zoom area to open the "Zoom" dialog box.

Remote command:
`LAYout:ZOOM:ADD` on page 1174

Hardware zoom ← Zoom
Changes the instrument settings - horizontal and vertical scales as well as trigger level and offset - to display a part of the diagram in greater detail.

Coupled zoom ← Zoom
Creates a coupled zoom area and its related zoom diagram. If you change the size of one zoom area, the size of all coupled zoom areas is changed as well.

Remote command:
`LAYout:ZOOM:ADDCoupled` on page 1175

Fingertip zoom ← Zoom
Magnifies the waveforms around your fingertip.

Tap the icon and put your finger on the waveform. The touched part of the waveform is displayed in a magnifier. Drag your finger on the screen to move the magnifier. You can change the zoom factor using the Navigation knob.

Search
Performs a search according to the settings in the "Search Setup" dialog box. Tap the icon and then tap the diagram with the waveform to be searched, or drag a rectangle to define a search gate. The search is performed on the selected waveform.



Cursor
The cursor icon shows the last selected cursor type. A short tap on the icon activates the selected cursor.

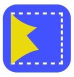
If you touch the icon and drag your finger down, a menu opens where you can select another cursor type: horizontal cursors, vertical cursors, or both.

Tap the diagram where you want to set the cursors, or draw a rectangle in the diagram to position the cursor lines. The resulting cursor lines measure the selected waveform. The results appear in the "Cursor Results" box. You can adjust the cursor source, type and position in the result box. Move the cursor lines by dragging them in the diagram, or by turning the navigation knob. Pressing the knob switches the parameter to be changed.

Masktest
Starts the on-screen mask definition and the testing against the defined mask.

Tap the icon and then tap the points that build the mask. Double-tap the last point to finish mask definition. To create a rectangular mask, draw a rectangle on the screen. You can move the mask on the screen.

To configure the mask test settings, tap the icon in the "Mask" result box.




Histogram
The histogram icon on the toolbar shows the last selected histogram type. A short tap on the icon activates the selected histogram.

If you touch the icon and drag your finger down, a menu opens where you can select another histogram type: horizontal histogram, or vertical histogram.

Tap the icon and then drag a rectangle on the diagram to mark the histogram area. The histogram for the selected waveform appears.

Touch and hold the histogram area to open the "Histogram" dialog box.

Measurement
Starts an automatic measurement.

You can run up to 8 automatic measurements in parallel. The "Automatic measurment" icon starts the measurements one after the other.

Tap the icon and then tap the diagram with the waveform to be measured. To define a measurement gate, draw a rectangle on the screen.

To modify the measurement, tap the icon in the "Measurement" result box.

Quick meas
Performs a set of measurements on the selected waveform. You can configure up to 8 measurement type to be included in quick measurement.

Tap the icon and then tap the diagram with the waveform to be measured.

FFT
Transforms a waveform to the frequency spectrum by fast Fourier transform (FFT). The FFT trace is shown in a new diagram.

Tap the icon and then tap diagram with the waveform to be transformed. The FFT diagram is created from the selected waveform.

To adjust FFT settings, double-tap the FFT diagram.

Select
Enables the select mode to move and modify objects on the touchscreen. The select mode is activated automatically when an analyzing function is completed.

Delete
Removes zoom and histogram areas and their diagrams; measurement areas and their associated results; and mask segments. The icon also switches off a waveform.

Tap the icon and then tap the area or diagram to be deleted, or the waveform to be switched off.

Save Settings
Saves the current instrument settings in a saveset. The filename is created according to the autonaming pattern. You can reload the saveset using the "Load saveset (Graphical recall)" toolbar icon, or using [FILE] > "Save/Recall" > "Settings".

Save Screenshot
Saves a screenshot of the current display using the settings defined in "File" menu > "Print Setup".

Create Report
Creates a report of the current measurement settings and results using the settings defined in "File" menu > "Report Setup".

Clear all
Deletes all measurement results including long term measurement and statistics, all waveforms, and the history.

Remote command:
`DISPlay:CLR` on page 1068

Autoset and Preset
Performs an autoset, or a preset to a default state. The icons have the same functionality as the corresponding keys on the front panel. They are useful when you control the instrument remotely.

Run / Stop and Run Single
Starts and stops the continuous acquisition, or starts a defined number of acquisition cycles. The icons have the same functionality as the corresponding keys on the front panel. They are useful when you control the instrument remotely.

Find Trigger Level
Analyses the signal and sets the trigger level to the middle of the signal peaks.

Force Trigger
Starts an immediate single acquisition. If the acquisition is running in normal mode and no valid trigger occurs, use "Force Trigger" to confirm that a signal is available. Then you can use the displayed waveform to determine how to trigger on it.

Label
Defines a waveform label that names or explains the waveform. Tap the icon and then tap the waveform to be labeled. If you tap the display background, the label is assigned to the selected waveform. Enter the label text using the onscreen keyboard. The text is shown in the same color as the waveform. You can drag the label to another position.

Update Ref Waveform
Copies the selected source waveform with all its settings to the reference waveform. If the acquisition is running, the reference waveform is a snapshot. You can configure up to four reference waveforms.

Select the required reference waveform (R1 to R4) in the toolbar menu of the icon.

Save Waveform
Exports the waveform data to file using the settings defined in [FILE] > "Waveforms / Results" > "Waveforms". The filename is created according to the autonaming pattern.

Tap the icon and then tap the waveform to be exported. If you tap the display background, the selected waveform is exported, or a multichannel export is performed if configured.

Spectrogram (option R&S RTE-K18)
Starts an FFT and the spectrogram. The FFT trace and the spectrogram are shown in separate diagrams.

Tap the icon and then tap diagram with the waveform to be transformed. The diagrams are created from the selected waveform.

2.4.8 Displaying Results

The results of automatic and cursor measurements, mask tests, and searches are displayed immediately in a result box.

There are several places to display the results:

- In a floating result box in front of the diagrams, which you can move on the display
- In a minimized view (result icon) on the signal bar
- In a separate tab

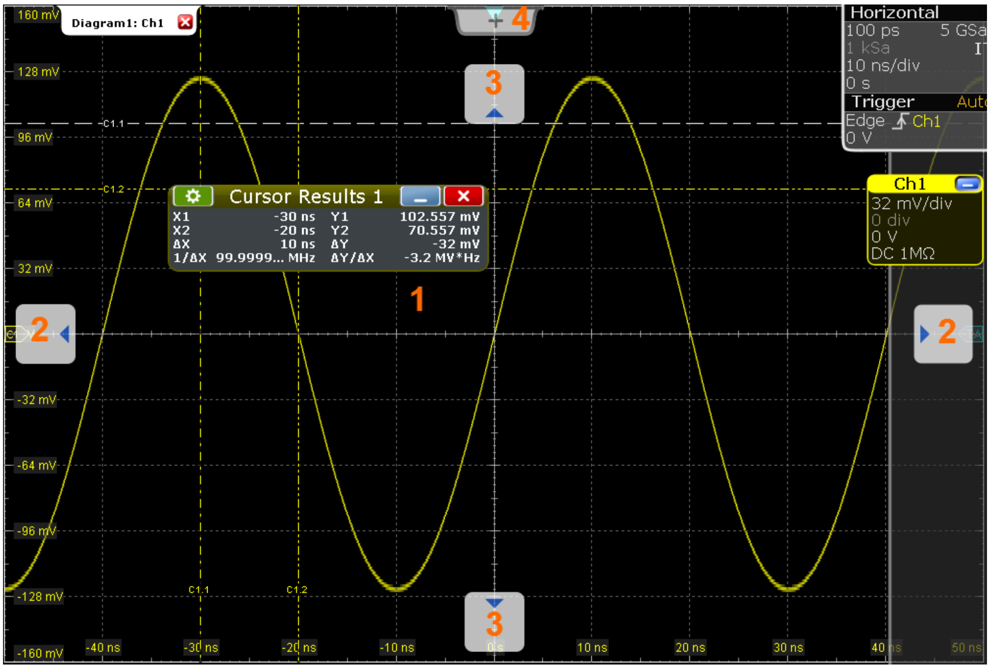
- In a docked tab below the diagram (available only for automatic and cursor measurements)

The default position and the font size can be adjusted.

To arrange a result box on the display

► Touch the title of the result box and drag the box on the screen. If the results are docked, touch a result line and drag it. The SmartGrid indicates where the result box will be placed.

- If you drop the box on one of the buttons, the results are shown in a separate tab besides, above, or below the diagram.
- If you drop the box on the signal bar, a result icon is created.
- If you drop the box somewhere else, a floating result box is created.



1 = Floating result box
2 = Table in a tab on the left or right
3 = Table in a tab above or below
4 = New tab

To open the corresponding setup dialog box

► In the result box, tap the ⚙ icon. Alternatively, double-tap the results.

The dialog box with corresponding settings opens.

To define the default position of measurement results

For results of automatic and cursor measurements, the docked position below the diagram is the initial default position.

► To change the default position of measurement results:

- Automatic measurements: select "Meas" menu > "Gate/Display" > "Default position".
- Cursor measurements: select "Cursor" menu > "Label/Display" > "Result position"

Note: Once you undock the measurement results, you cannot move the result box back to the docked position.

To define the default position of other results (mask test, search)

1. Press the [DISPLAY] key on the front panel.
2. In the "Display" dialog box, select the "Diagram Layout" tab.
3. Under "Result box", select the "Result position":
   - "Preview": result icon on the signal bar
   - "Floating": floating result box in front of the diagrams

To adjust the font size in result boxes

1. Press the [SETUP] key.
2. Select the "Screen" tab.
3. Set the "Result dialog font size".

2.4.9 Using Dialog Boxes

All functionality is provided in dialog boxes as known from computer programs. You can control the instrument intuitively with the touchscreen. This section provides an overview of the accessing methods and describes how to use the dialog boxes.

Each dialog box has four icons in the upper right corner:

	Go back: opens the previously opened dialog box.
	Go forward: opens the next dialog box.
	Minimizes the dialog box to a small box that only contains the last selected function.
	Closes the dialog box.

For direct access to important control and measurement functions use the toolbar, see Chapter 2.4.7, "Toolbar", on page 79.

To open a dialog box

► Perform one of the following actions:

- Tap the required menu, and then the menu entry.
- Press the function key on the front panel.
- If a results box is open, tap the icon to open the corresponding dialog box.
- To open the "Vertical" dialog box of a waveform, tap its signal icon. For XY-waveforms, the "XY Diagram" tab opens.
- Tap the horizontal or trigger label to open the "Horizontal" or "Trigger" dialog box, respectively.

To minimize a dialog box

If you want to change only one setting during analysis, and you need to change it often, you can display a small box that only contains the required setting.

1. Tap the function that you need to modify.
2. Tap the "Minimize" icon in the upper right corner of the dialog box.

   The dialog box turns into a small box that contains only the "Wfm Arithmetic" setting.

3. To restore the complete dialog box, tap the "Maximize" icon in the small box.

To close a dialog box

► Tap the "Close" icon in the upper right corner.
Or:
Press the [ESC] key on the front panel.

To select an option in a dialog box

► Tap the required option.
Or:
Press the [⇤ Field] and [⇥ Field] keys to navigate to the required option. Then press the [☑] key.

To select an option in a list

If many options are available - for example, for the trigger type - the options are provided in a list. The current selection is shown on the list button.

► Tap the list button. Then tap the required option.
Or:
Use the front panel keys:

a) Press the [⇤ Field] and [⇥ Field] keys to navigate to the list button.
b) Press the [☑] key to open the list.

c) Press the [⇧] and [⇩] keys to navigate to the required option in the list.
d) Press the [☑] key to select the marked option.

2.4.10 Entering Data

Most important parameters have their own rotary knobs on the front panel. When you turn a knob, the input box appears the upper left corner of the screen, showing the parameter name and current value.

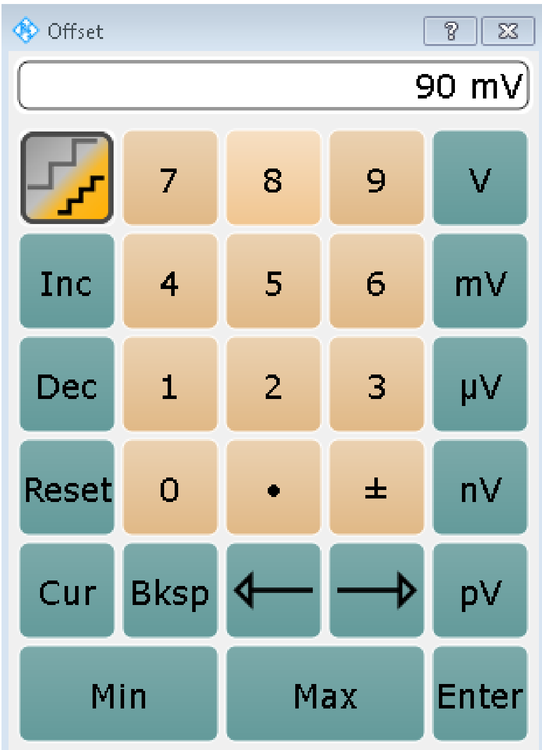
Using rotary knobs

1. Turn the knob to change the value.
2. To toggle the increment, tap the "Steps" icon in the input box.
3. To set the parameter to the autoset value (if available), tap the "RESET" icon.

For data input in dialog boxes, the touchscreen provides an on-screen keypad to enter numeric values and units. For text input, the on-screen keyboard with English key layout is used.

To enter values with the on-screen keypad

1. Double-tap the entry field.
The on-screen keypad opens.



2. Enter the numeric value using the following methods:
   - To use the default value, tap "Reset" (if available).
   - To use the minimum or maximum value, tap "Min" or "Max", respectively.
   - To increase the displayed value in fixed steps, tap "Inc".
     To decrease the value in fixed steps, tap "Dec".
     To toggle between small steps and large steps, tap the "Steps" icon.
     
   - To get the value that was used before the keypad was displayed, tap "Cur".
   - To enter a user-defined value, tap the numbers and complete the entry by tapping the unit button.
     – The arrow buttons move the cursor left or right.
     – "Bksp" deletes the last character before the cursor.
     – "±" changes the sign of the value.

To enter data with the on-screen keyboard

1. Double-tap the entry field to open the on-screen keyboard.
   If available, you can also tap the keyboard icon on the right of the entry field.

2. Enter the text as you would on a normal keyboard.
   - To enter a series of capital characters, tap "Caps".
     To enter one capital character, tap "Shift".
   - To use the currently defined value, tap "Cur". This is the value that was used before the keyboard was displayed.
   - The arrow buttons move the cursor left or right.
   - "Bksp" deletes the last character before the cursor.
3. Tap "Enter" to complete the entry.

To enter numeric data in a dialog box with navigation controls

1. To navigate to the entry field, press the [⇤ Field] and [⇥ Field] keys.

2. To change the value with a small step size, turn the rotary knob.
   Alternatively, press the [⇧] and [⇩] keys for a larger step size.

If you edit numeric data in tables, the entry field must be in edit mode. To activate the edit mode, press ENTER, or the [☑] key, or the navigation rotary knob.

2.4.11 Messages

Status messages of the instrument are displayed for a few seconds. Then they are saved in a message box in the upper right corner of the screen. By default, the message box is closed. You can open it to read the messages and to delete them.

► To open and close the message box, tap the "Info" button.

If no messages are available, the "Info" button is hidden.

Important messages are indicated by a red "Info" button. These messages cannot be deleted, they remain until the problem is solved.

2.4.12 Getting Information and Help

In many dialog boxes, graphics are included to explain the way a setting works. For further information, you can use the following sources:

- Tutorials are silent movies, which demonstrate the general usage of the R&S RTE, for example, how to use the SmartGrid. They are available on the internet product page, in the R&S®RTE1000 - Media Center.
- Tooltips give a short description of the parameter.
- The context help provides functional description on a setting, and the corresponding remote command.
- The general help explains a dialog box, provides instructions, and general information.

2.4.12.1 Displaying Help

To display tooltips and context help

1. Enable the "Tooltip" icon on the toolbar.

2. Tap the parameter for which you need information.

   The tooltip opens.

3. To open the corresponding help topic, tap the "Show Help" button in the lower right corner of the tooltip.

   The "Help" window opens and displays the comprehensive description and the corresponding remote command. You can browse the help for further information.

   Note: The tooltip icon disables automatically when you tap a parameter. To show another tooltip, tap the tooltip icon again.

To open general help

► Press the yellow [HELP] button on the left side of the screen.

  If a dialog box is open, the help topic for the current tab is shown. Otherwise the "Contents" page appears.

2.4.12.2 Using the Help Window

The Help window contains several tabs:

- "View" - shows the selected help topic
- "Contents" - contains a table of help contents
- "Index" - contains index entries to search for help topics
- "Search" - provides text search

The Help toolbar provides some buttons:

- To browse the topics in the order of the table of contents: Up arrow = previous topic, Down arrow = next topic
- To browse the topics visited before: Left arrow = back, Right arrow = forward
- To increase or decrease the font

To navigate the Help, use the touchscreen. Alternatively, you can also use the navigation keys on the front panel.

To search for a topic in the index

The index is sorted alphabetically. You can browse the list, or search for entries in the list.

1. Switch to the "Index" tab.

2. Select the "Keyboard" icon besides the entry field.

3. Enter the first characters of the keyword you are interested in.

   The entries containing these characters are displayed.

4. Double-tap the suitable index entry.

   The "View" tab with the corresponding help topic is displayed.

To search topics for a text string

1. Switch to the "Search" tab.
2. Select the "Keyboard" icon besides the entry field.
3. Enter the string you want to find.
   If you enter several strings with blanks between, topics containing all words are found (same as AND operator).

   For advanced search, consider the following:
   - To find a defined string of several words, enclose it in quotation marks. For example, a search for *"trigger qualification"* finds all topics with exactly *"trigger qualification"*. A search for *trigger qualification* finds all topics that contain the words *trigger* and *qualification*.
   - Use "Match whole word" and "Match case" to refine the search.
   - Use operators AND, OR, and NOT.

To close the Help window

► Select the "Close" icon in the upper right corner of the help window.
  Or: Press the [ESC] key.

3 Instrument Setup

You can adapt various instrument settings to your requirements, such as language, display appearance, and assign functions to some keys.

The chapter describes also the handling of software options.

The following setup procedures are described in other chapters of the documentation:

- Chapter 2.1.4, "Connecting External Devices", on page 19
- Chapter 16.3, "Web Interface", on page 1001
- Chapter 16.6, "Remote Settings", on page 1013
- The firmware update is described in the release notes.

The following settings and procedures are described in the current chapter:

- System Setup....94
- Screen Setup....99
- Frontpanel Setup....101
- Display Configuration....104
- External Application....120
- Self-alignment....120
- Self-test....122
- Firmware Update....123
- Options....123

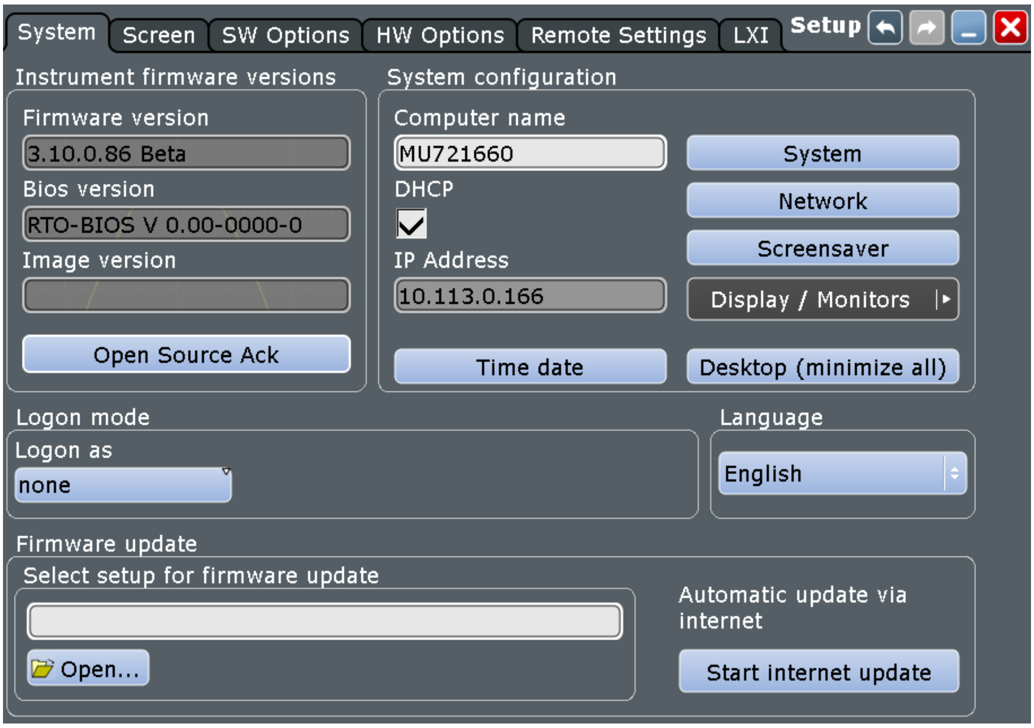
3.1 System Setup

- System Settings....94
- Setting the Display Language....98

3.1.1 System Settings

Access: [SETUP] > "System" tab

The settings on this tab are related to the basic instrument and system configuration.



Firmware version....................95
Bios version....................95
Image version....................95
Desktop (minimize all)....................96
Computer name, IP Address, DHCP....................96
System....................96
Network....................96
Screensaver....................96
Display / Monitors: Display Settings....................96
Time, date....................97
Log on as....................97
Language....................97
Select setup for firmware update....................97
Start internet update....................98

Firmware version

Indicates the firmware version currently installed on the instrument.

Remote command:

`DIAGnostic:SERVice:FWVersion?` on page 1050

Bios version

Indicates the BIOS version currently installed on the instrument.

Image version

Indicates the image version currently installed on the instrument.

Desktop (minimize all)
Minimizes all displayed application windows on the instrument, so that the desktop becomes visible on the screen to access the Windows functionality.

This function is also available from the "File" menu.

Computer name, IP Address, DHCP
Indicates the currently defined computer name, the defined IP address and DHCP address enabling. These values are required to configure the instrument for work in a network.

NOTICE! Risk of network problems. All parameters can be edited here; however, beware that changing the computer name has major effects in a network. For details, see Chapter 16.2, "Setting Up a Network (LAN) Connection", on page 996.

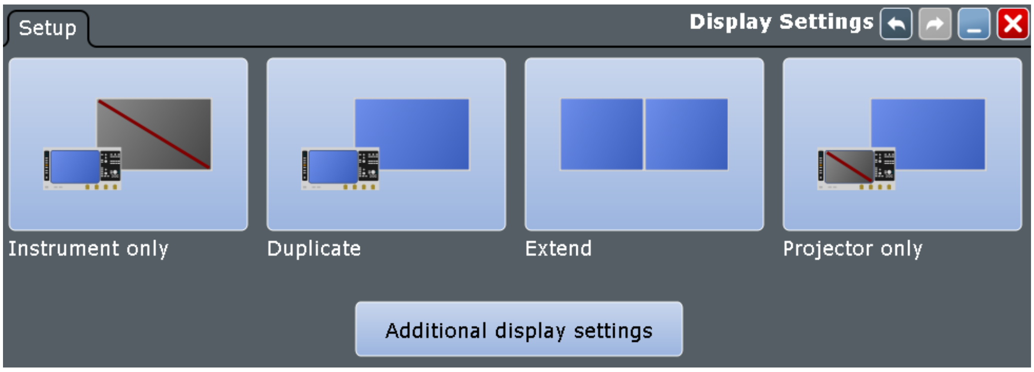
Remote command:
`DIAGnostic:SERVice:COMPutername` on page 1050

System
Opens the standard Windows "System Properties" dialog box to configure system settings. Only users with administrator rights can fulfill this task.

Network
Opens the standard Windows "Network Connections" dialog box to configure a network connection. Only users with administrator rights can fulfill this task.

Screensaver
Opens the standard Windows "Display Properties" dialog box to configure a screen saver.

Display / Monitors: Display Settings
The "Display / Monitors" button opens the "Display Settings" dialog box, where you can extend or duplicate the instrument display to a second monitor or projector (external display).



"Instrument only"	The instrument display is on, the external display is off.
"Duplicate"	The external display shows the same content as the instrument display.

"Extend"	The instrument display and the external display show different content.
"Projector only"	The instrument's user interface is only shown on the external display, the instrument display is off.
"Additional display settings"	Opens the Windows configuration for display settings.

Time, date
Opens the standard Windows "Date and Time Properties" dialog box to set the correct date and time. Only users with administrator rights can fulfill this task.

Note: Usually date and time are set correctly. To adjust your regional time, select the correct time zone rather than changing the time.

Remote command:
`SYSTem:DATE` on page 1049
`SYSTem:TIME?` on page 1049

Log on as
Sets the user that is automatically logged on during the startup process of the instrument. The change of this setting takes effect at the next instrument startup.

See Chapter 16.1.1, "Logon", on page 991 for restrictions of the standard user and how to change the auto-logon.

"User autologon"	Auto-logon as standard user with limited access. Enter the "User name": *NormalUser* and the "Password" of the standard user.
"Admin autologon"	Auto-logon with unrestricted access to the instrument and network. The setting is only available for administrators. Enter the "User name": *Instrumen* and the administrator's "Password" to enable the auto-logon.
"None"	No auto-logon, user name and password are requested at instrument startup.

Language
Selects the language in which the dialog boxes, result boxes and other screen information is displayed.

Select setup for firmware update
Your instrument is delivered with the latest firmware version. Firmware updates are provided on the internet at:

www.rohde-schwarz.com/firmware/rte.

The "Release Notes" describe the improvements and modifications of all firmware versions ans also how to update the firmware. They are available along with the firmware on the same web page.

Only users with administrator rights can fulfill this task.

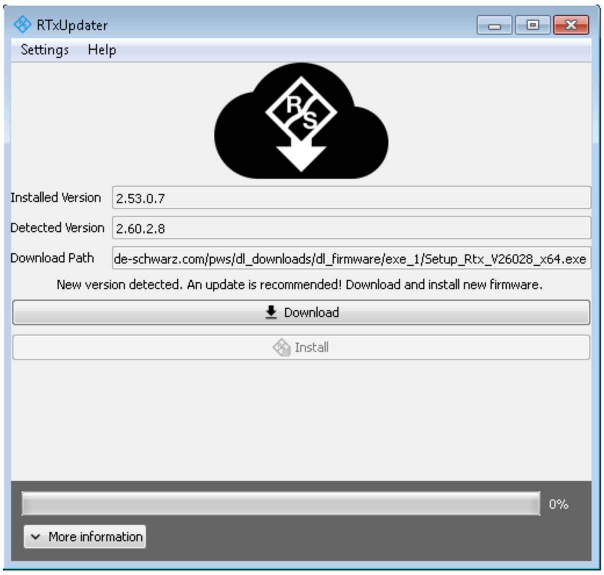
"Load"	Loads the specified file.
"Open"	Opens a file selection dialog box and loads the selected file.

"Explore" Opens the Windows file explorer where you can navigate and search for files and folders as usual.

Start internet update

Starts the "RTxUpdater", which connects to the internet, checks for newer versions, downloads the firmware file, and installs the firmware. Only users with administrator rights can fulfill this task.

Make sure that your device is connected to the Internet. If your corporate network uses a proxy server, enter the proxy settings in "Settings" menu > "Proxy Settings". Ask your administrator for correct proxy settings.



A short instruction is available under "Help" > "Help".

3.1.2 Setting the Display Language

You can change the language in which the dialog boxes, result boxes and other screen information is displayed. A reboot of the instrument is not necessary.

1. Press the [SETUP] key.
2. Select "System".
3. Tap the "Language" button. The button shows the current language.
4. Select the required language.

   The instrument changes the language after a few seconds.

3.2 Screen Setup

- Screen Settings..99
- Aligning the Touchscreen..100

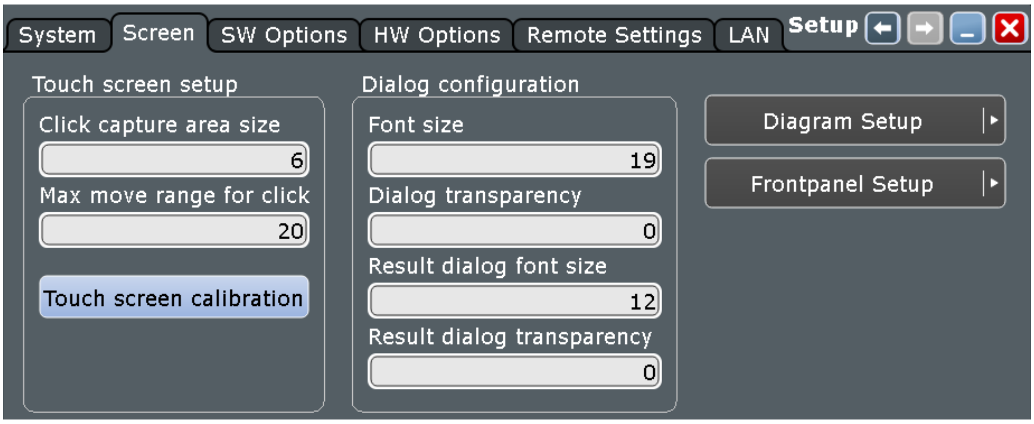
3.2.1 Screen Settings

Access: [SETUP] > "Screen" tab

The settings on this tab are related to the screen display.

Note for "Dialog configuration", "Front panel setup" and "Navigation rotary knob" settings:

These settings are user-specific, they are *not* reset by [PRESET] and `*RST`. You can reset them to default values using [FILE] > "Save/Recall > User defined preset > Factory defaults" or using the `SYSTem:PRESet` command.



Click capture area size..99
Max move range for click...100
Touchscreen calibration...100
Font size..100
Dialog transparency...100
Result dialog font size..100
Result dialog transparency...100

Click capture area size

Defines the number of pixels around each element (e.g. button, icon, data point) that create a capture area. If you tap your finger or click the mouse pointer within this capture area, this element is considered to be selected. If you tap or click outside this area, a different or no element is selected.

The larger the area, the easier is it to select an element. However, when selecting data points, for example, a large frame does not allow you to select precisely.

Max move range for click
Defines the maximum number of pixels around an element (e.g. data point) within which your pointing device must stay to "click" the element. When you tap or click a specific element and move your finger or the mouse outside this range, it is considered to be a "moving" or "dragging" operation.

Touchscreen calibration
Opens the touchscreen calibration application, see Chapter 3.2.2, "Aligning the Touchscreen", on page 100. Only users with administrator rights can fulfill this task.

Font size
Defines the font size of the text in dialog boxes.

Dialog transparency
Defines the transparency of the dialog box background. For high transparency values, you can see the waveform display in the background, and possibly check the effect of the changed setting. For lower transparency values, readability in the dialog box improves.

Result dialog font size
Defines the font size of the text in result boxes. The size of the result box is adapted to the font size.

Result dialog transparency
Defines the transparency of the measurement result boxes in the same way as Dialog transparency.

3.2.2 Aligning the Touchscreen

When the device is delivered, the touchscreen is initially calibrated. However, to ensure that the touchscreen responds to the finger contact correctly, a touchscreen alignment is required. Only users with administrator rights can fulfill this task.

Alignment of the touchscreen is useful:

- At first use
- If the position of the instrument has been changed, and you cannot look straight on the screen
- If another person operates the instrument
- If you notice, that touching a specific point on the screen does not achieve the correct response

1. Press the [SETUP] key.
2. Select the "Screen" tab.
3. Tap "Touchscreen Calibration".

   A blinking cross appears in the lower left corner of the screen.
4. Touch and hold the cross until "OK" is shown.

5. Repeat this action for the crosses in the other corners.
6. Tap the R&S logo button in the task bar to display the instrument's user interface.

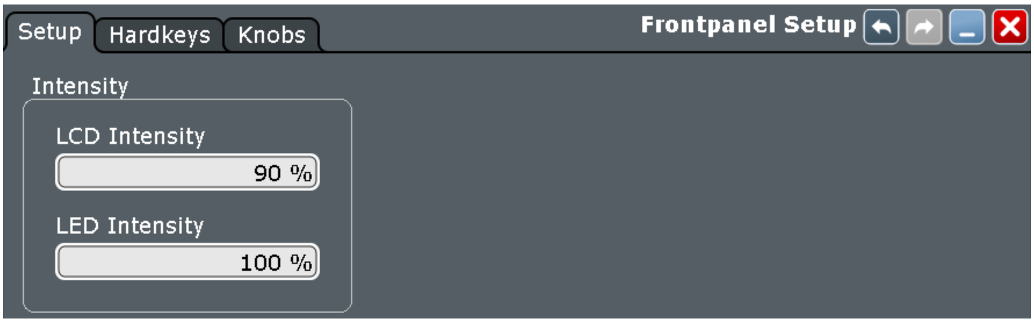
3.3 Frontpanel Setup

In the "Frontpanel Setup" dialog box, you can adjust the luminosity of the screen and luminous keys, assign functions to keys and knobs, and adjust the NAVIGATION knob.

- Setup: Luminosity Settings....................101
- Hardkeys: Function Assignment....................101
- Knobs....................102

3.3.1 Setup: Luminosity Settings

Access: "File" menu > "Frontpanel Setup" > "Setup"



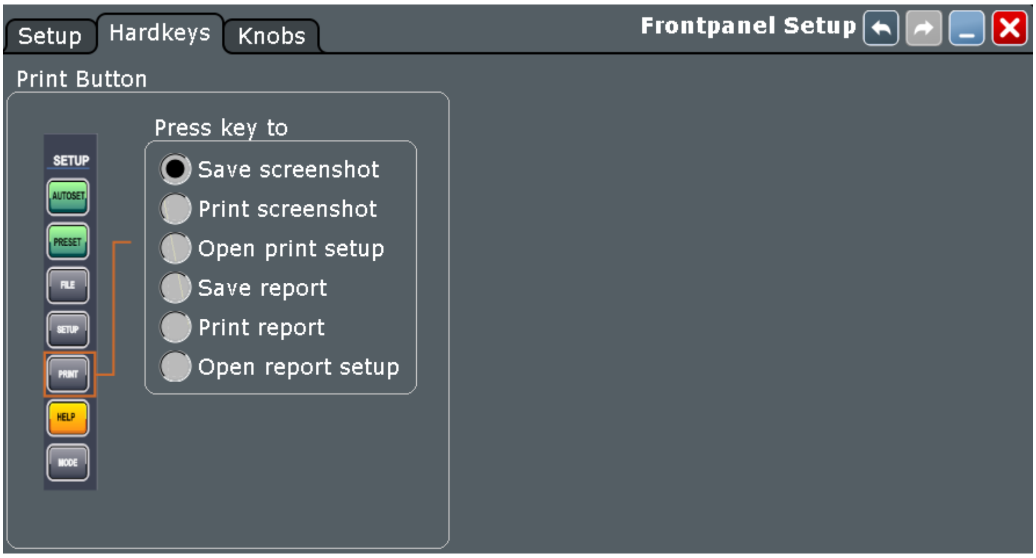
LCD Intensity
Changes the background luminosity of the touchscreen.

LED Intensity
Defines the luminosity of illuminated front panel keys and rotary knobs.

3.3.2 Hardkeys: Function Assignment

Access: "File" menu > "Frontpanel Setup" > "Hardkeys"

You can configure the function of some controls on the front panel to your needs.



Print Button.. 102

Print Button

The PRINT key on the left side of the display is a shortcut key that initiates an associated action.

You can assign one of the following actions to the PRINT key:

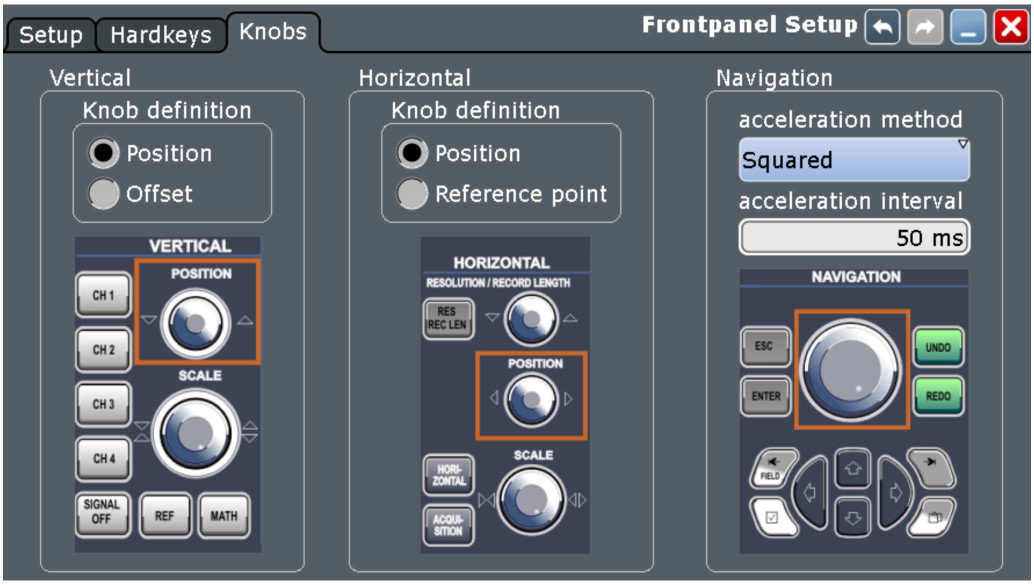
- Save a screenshot
- Print a screenshot
- Open print setup
- Save a report
- Print a report
- Open report setup

Configure the settings for the selected action.

- Screenshots: "File" menu > "Print Setup", see Chapter 11.4.1, "Screenshot Settings", on page 462.
- Report: "File" menu > "Report Setup", see Chapter 11.5.1, "Report Settings", on page 467.

3.3.3 Knobs

Access: "File" menu > "Frontpanel Setup" > "Knobs"



Vertical.......... 103
Horizontal.......... 103
Rotary knob acceleration method.......... 103
Rotary knob acceleration interval.......... 104

Vertical
The vertical POSITION / OFFSET knob can change the waveform position or the offset of the selected waveform. Select the action that you want to perform.

See also: Chapter 2.2.3.4, "VERTICAL Controls", on page 30.

Horizontal
The horizontalPOSITION / REF POINT knob can change the horizontal position or the reference point. Select the action that you want to perform.

See also: Chapter 2.2.3.3, "HORIZONTAL Controls", on page 29.

Rotary knob acceleration method
Selects a method to accelerate the movement of the element on the screen compared to the actual movement of the rotary knob. Acceleration is useful if you need to move from one end of the screen to the other, for example. Without acceleration, you have to turn the knob quite a while to reach the other end. On the other hand, acceleration can make precise selection difficult, since a small movement of the knob causes a relatively large movement on the screen.

"None"	No acceleration method used.
"Squared"	Moderate acceleration method used.
"Exponential"	Strong acceleration method used.

Rotary knob acceleration interval
Defines the delay time during which the movement of the rotary knob is analyzed before acceleration is applied. For short intervals, acceleration sets in quickly, but is not as effective. For long intervals, acceleration is more effective. However, it takes longer until the instrument reacts on the knob's input. Furthermore, when you turn the knob slowly during fine-tuning, subsequent movements that occur during the same interval are accelerated, making precise selection difficult.

3.4 Display Configuration

- Adjustable Display Elements....................104
- Display Settings....................105
- Adjusting the Display....................115

3.4.1 Adjustable Display Elements

You can customize the various elements on the screen according to your needs:

Signal bar

The signal bar can be manually switched on and off, it can be automatically hided, and you can adjust color and transparency of the bar.

Toolbar

The toolbar contains icons that start frequently used functions. You can define which tools are displayed on the toolbar.

Diagrams

The basic diagram elements can be shown or hidden: grid, crosshair, label, and tab titles. You can also enter user-defined diagram names.

Waveforms

For waveforms, you can adjust the persistence, the waveform style, and color. You can also annotate the waveforms by adding screen texts.

To set the color, you can select it from a color palette or assign color tables defining the color of waveform pixels depending on the cumulative occurrence of the associated values. You can assign a different color or color table to each waveform.

The following default color tables are provided:

- "False colors"
- "M-Hot"
- "M-Hsv"
- "M-Jet"
- "Spectrum"

- "Single Event"
- "Temperature"

Dialog boxes and result boxes

You can configure the font size, contrast and transparency in dialog and result boxes. Thus, you can optimize readability or keep track of the waveforms while changing settings in dialog boxes.

Clear results

To delete all results, statistics, waveforms, and also the history, select "Display" menu > "Clear all".

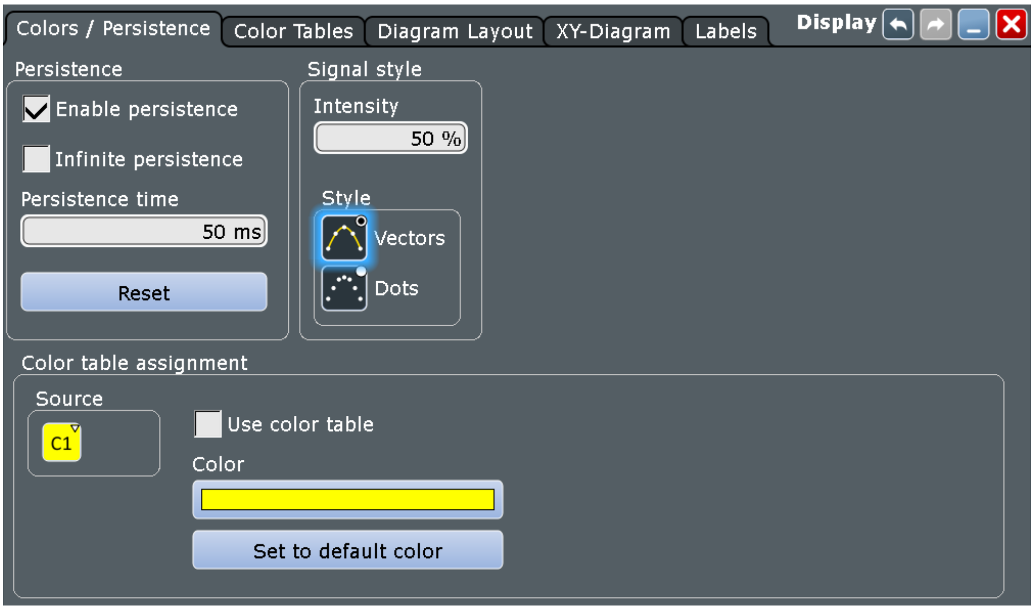
3.4.2 Display Settings

Display settings are configured in the "Display" dialog box, which is opened when you press the [DISPLAY] key or select an item from the "Display" menu.

3.4.2.1 Colors / Persistence

Access: [DISPLAY] > "Colors / Persistence" tab

The "Colors / Persistence" tab contains settings for the general display of waveform data.



Enable persistence....106
Infinite persistence....106
Persistence time....106
Reset....106
Intensity....106

Style......106
Color......107
Set to default color......107
Use color table......107
Source......107
Assigned color table......107

Enable persistence
If enabled, each new data point in the diagram area remains on the screen for the duration that is defined using Persistence time, or as long as Infinite persistence is selected.

If disabled, the waveform points are displayed only for the current acquisition.

Remote command:
`DISPlay:PERSistence[:STATe]` on page 1053

Infinite persistence
If infinite persistence is enabled, each new waveform point remains on the screen until this option is disabled. Use infinite persistence to display rare events in the signal.

Remote command:
`DISPlay:PERSistence:INFinite` on page 1053

Persistence time
Sets a time factor that controls how long the waveforms points fade away from the display. Thus, the R&S RTE emulates the persistence of analog phosphor screens.

Remote command:
`DISPlay:PERSistence:TIME` on page 1053

Reset
Resets the display, removing persistent all waveform points.

Remote command:
`DISPlay:PERSistence:RESet` on page 1054

Intensity
This value determines the strength of the waveform line in the diagram. Enter a percentage between 0 (not visible) and 100% (strong). The default value is 50%.

You can also use the [INTENSITY] knob on the left side of the screen to adjust the waveform intensity directly.

Note: Use of color tables. The exact mapping of the cumulative value occurrences according to the assigned color table is guaranteed only if the intensity is set to 50%. All other intensity values falsify the mapping but can improve the visibility of the signal. See also: Chapter 3.4.3.2, "Changing Waveform Colors", on page 115.

Remote command:
`DISPlay:INTensity` on page 1054

Style
Select the style in which the waveform is displayed:

"Vectors"	The individual waveform points are connected by a line. Define the strength of the line using the [INTENSITY] knob on the left side of the screen.
"Dots"	Only the individual waveform points are displayed. Waveform sample points are the ADC sample points and additional interpolated points if "Interpolated time" is used for resolution enhancement. To see the dots of one waveform, perform one acquisition with [RUN N× SINGLE] and N=1 ("Average count" = 1). During continuous acquisition, or a [RUN N× SINGLE] acquisition with N > 1, the dots of multiple subsequent waveforms are displayed on the screen, and the waveform looks like a line.

Remote command:
`DISPlay:DIAGram:STYLe` on page 1054

Color
Shows the current color of the selected waveform. To change the color, tap the button and select a color. The color of the waveform, its signal icon, channel icon, and of the illuminated keys is adjusted to the new color.

Remote command:
`DISPlay:COLor:SIGNal<m>:COLor` on page 1054

Set to default color
Resets the color of the selected waveform to the factory default.

Use color table
If enabled, the selected waveform is displayed according to its assigned color table.

If this option is disabled, the selected color is displayed, and the intensity of the specific signal color varies according to the cumulative occurrence of the values.

Remote command:
`DISPlay:COLor:SIGNal<m>:USE` on page 1055

Source
Selects the waveform to which the color table and the labels are assigned.

A spectrogram (option R&S RTE-K18) always has the same color as the math (spectrum) waveform from which it is created.

Assigned color table
Adjust the waveform colors to suit your preferences. For each of the following waveform types you can assign a suitable color table:
- Analog and digital channels
- Reference waveforms
- Results of a mathematical function, also for FFT and derived spectrogram.
- Measurements and tracks
- XY-traces
- Serial buses if a protocol option is activated
- Parallel buses if MSO option is installed

See also: Chapter 3.4.2.2, "Color Tables", on page 108.

Remote command:
`DISPlay:COLor:SIGNal<m>:ASSign` on page 1055

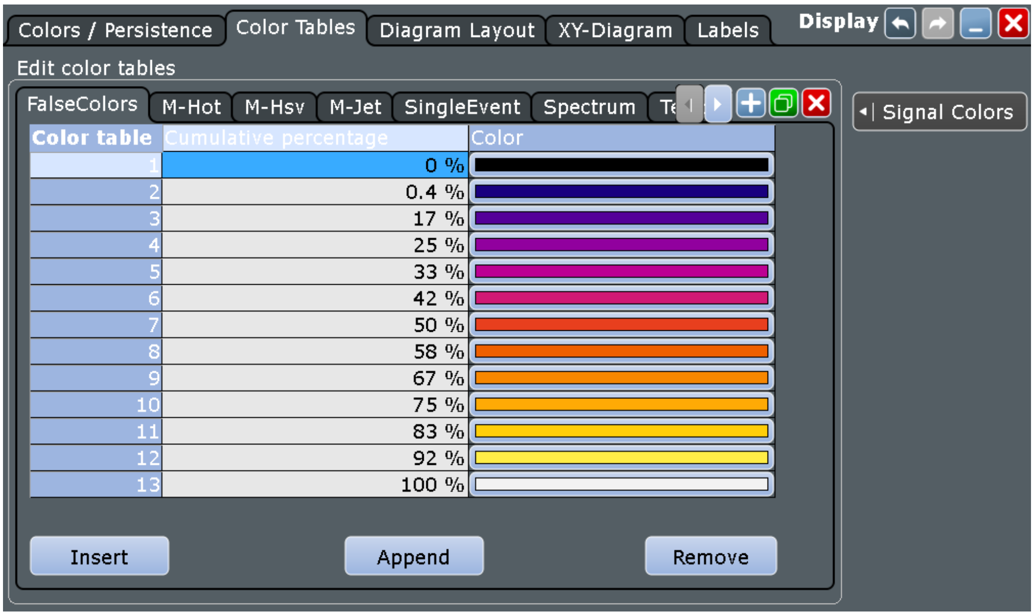
3.4.2.2 Color Tables

Access: [DISPLAY] > "Color Tables" tab

Color tables define the color of the waveform pixels depending on the cumulative occurrence of the associated values. By default, the intensity of the specific waveform color varies according to the cumulative occurrence of the values. The more often a value occurs, the darker the color of the data point is displayed.

The following default color tables are provided:

- "False colors"
- "M-Hot"
- "M-Hsv"
- "M-Jet"
- "Spectrum"
- "Single Event"
- "Temperature"



The editing table allows you to edit existing color tables or add new ones that can then be assigned to the waveforms. To assign a color table to a waveform, use the "Signal colors / Persistence" tab.

See also:

- Chapter 3.4.3.2, "Changing Waveform Colors", on page 115
- Assigned color table

Remote commands

The following remote commands are used to configure color tables:

DISPlay:COLor:PALette:COUNt? on page 1056

DISPlay:COLor:PALette:ADD on page 1055

DISPlay:COLor:PALette:REMove on page 1056

DISPlay:COLor:PALette:POINt:INSert on page 1056

DISPlay:COLor:PALette:POINt:ADD on page 1056

DISPlay:COLor:PALette:POINt[:VALue] on page 1056

DISPlay:COLor:PALette:POINt:COUNt? on page 1057

DISPlay:COLor:PALette:POINt:REMove on page 1056

DISPlay:COLor:PALette:COUNt? on page 1056

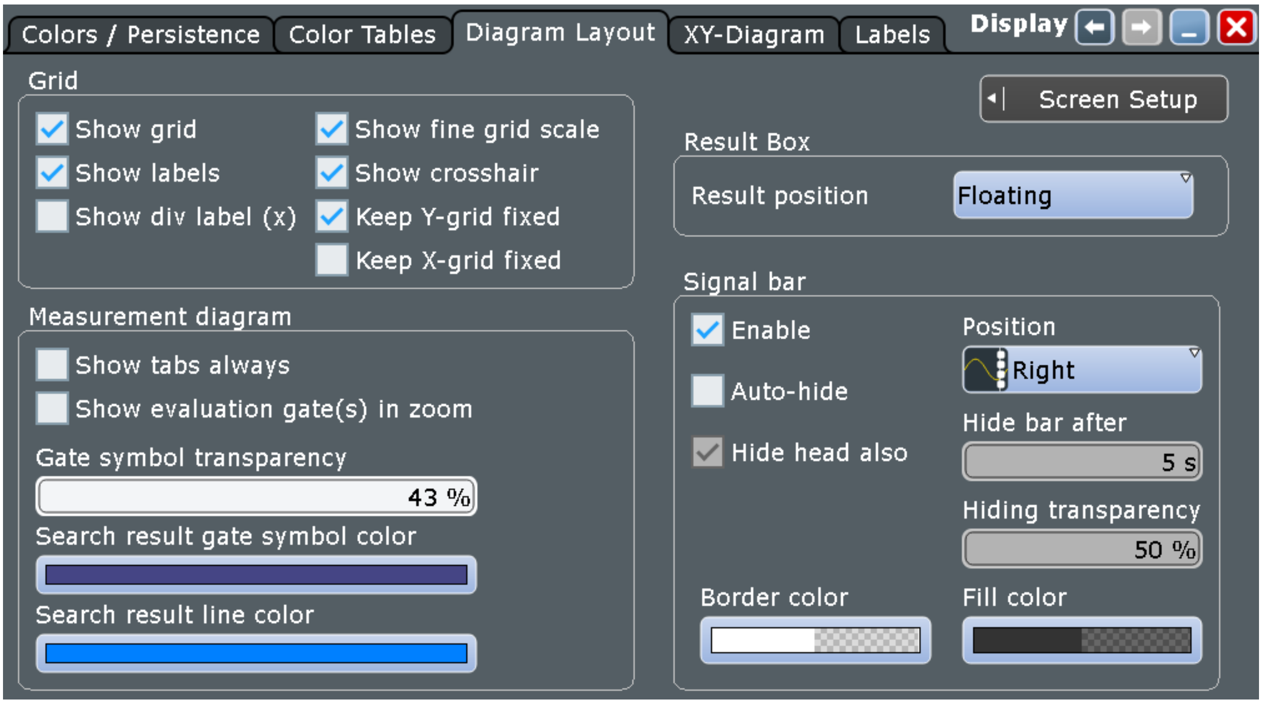
3.4.2.3 Diagram Layout

Access: [DISPLAY] > "Diagram Layout" tab

On the "Diagram Layout" tab, you define the elements to be shown in the diagrams, and the position of result boxes.

These settings are user-specific, they are *not* reset by [PRESET] and *RST. You can reset them to default values using [FILE] > "Save/Recall > User defined preset > Factory defaults" or using the SYSTem:PRESet command.

The diagrams and tabs are created and arranged with drag&drop and the SmartGrid, see Chapter 2.4.5, "Rohde & Schwarz SmartGrid", on page 76.



Show grid..110
Show labels.. 110
Show div label (x).. 111
Show fine grid scale..111
Show crosshair... 111
Keep Y-grid fixed..111
Keep X-grid fixed... 111
Show tabs always... 111
Show evaluation gate(s) in zoom..111
Gate symbol transparency..112
Search result gate symbol color... 112
Search result line color...112
Result position.. 112

Show grid

If selected, a grid is displayed in the diagram area. A grid helps you associate a specific data point to its exact value on the x- or y-axis.

Remote command:

`DISPlay:DIAGram:GRID` on page 1060

Show labels

If selected, labels mark values on the x- and y-axes in specified intervals in the diagram.

Remote command:

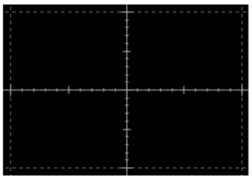
`DISPlay:DIAGram:LABels` on page 1061

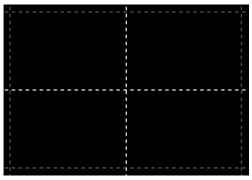
Show div label (x)

If selected, the time scale value is shown at the diagram bottom instead of the horizontal grid labels. For example, 10 ns/div is shown instead of the values 0, 10, 20, 30... ns.

Show fine grid scale

If selected, the crosshair is displayed as a ruler with scale markers. If disabled, the crosshair is shown as dashed lines.





Remote command:
`DISPlay:DIAGram:FINegrid` on page 1061

Show crosshair

If selected, a crosshair is displayed in the diagram area. A crosshair allows you to select a specific data point by its co-ordinates.

Remote command:
`DISPlay:DIAGram:CROSshair` on page 1061

Keep Y-grid fixed

If enabled, the horizontal grid lines remain in their position when the position of the curve is changed. Only the values at the grid lines are adapted. Fixed horizontal grid lines correspond to the behavior of traditional oscilloscopes.

Remote command:
`DISPlay:DIAGram:YFIXed` on page 1061

Keep X-grid fixed

If enabled, the vertical grid lines remain in their position when the horizontal position is changed. Only the values at the grid lines are adapted.

Remote command:
`DISPlay:DIAGram:XFIXed` on page 1062

Show tabs always

If selected, the tab titles of all diagrams are displayed: "Diagram1", "Diagram2" ...

If cleared, the tab titles are not shown except for titles in a tabbed diagram. In tabbed diagrams, the tab titles are required to change the tabs.

Remote command:
`DISPlay:DIAGram:TITLe` on page 1061

Show evaluation gate(s) in zoom

If enabled, the available histogram areas, masks, and measurement gates are shown in the zoom diagrams. If the evaluation gate is within the zoom area, the display helps to move or modify the evaluation gates in the zoom window.

Make sure that the option is disabled if the zoom area and the evaluation gate are of nearly the same size to avoid conflicts in operation.

Gate symbol transparency
Sets the transparency of the area that is defined as measurement or search gate. The setting only takes effect if "Show gate" is enabled.

Remote command:
`DISPlay:GATE:TRANsparency` on page 1062

Search result gate symbol color
Sets the color of the search zoom area. The search zoom area is displayed if "Show search zoom windows" is enabled. See also: "Search zoom window" on page 426.

Search result line color
Sets the color of the search result markers. The markers are displayed if "Show search zoom windows" is enabled.

Result position
Defines where a new result box opens.

Results of automatic and cursor measurement, and also decoding results, have their own position setting.

"Floating"	The result box opens as a box similar to a dialog box in front of the diagrams. It can be moved and shows all results.
"Preview"	The result box opens as a minimized result icon on the signal bar. It shows only two columns and a few rows of the results.

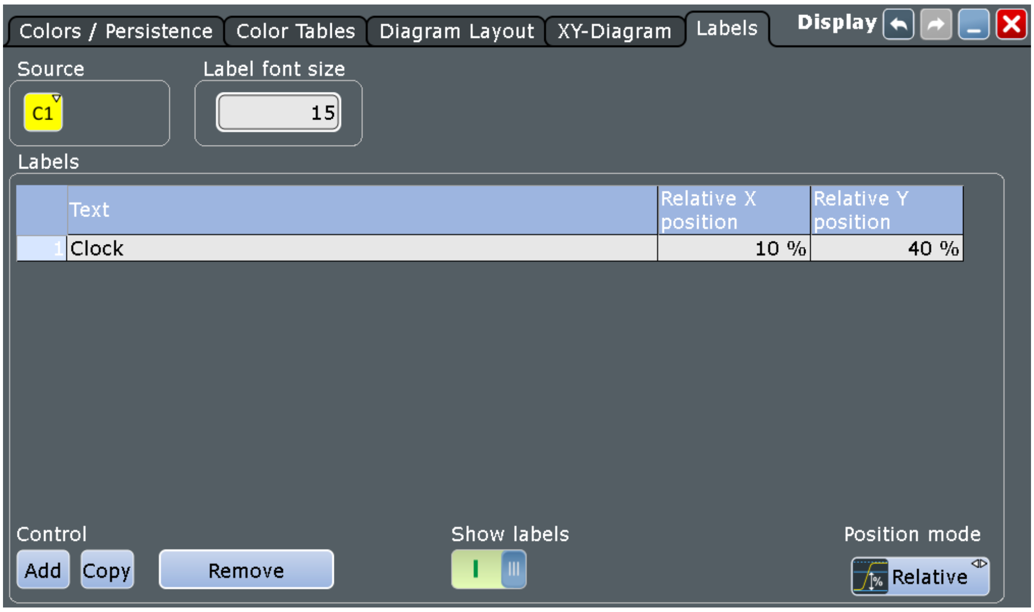
Remote command:
`DISPlay:RESultboxes:DEFaultpos` on page 1062

3.4.2.4 Waveform Labels

Access: [DISPLAY] > "Labels" tab

Using labels, you can annotate the waveforms to name or explain each waveform. The text is shown in the same color as the assigned waveform. Each label has its individual position. You can enter exact positions in the dialog box, or drag the labels on the screen to the required position. The position can be a fixed one (relative to the screen), or a flexible position (absolute, assigned to the axes).

To add labels quickly, you can add the "Label" icon to the toolbar and use it.



Make sure that the correct waveform tab is selected before you enter the labels.

Labels..113
Show labels..113
Position mode..114
Label font size..114

Labels

For each waveform, the "Labels" table shows the assigned texts and their positions. Enter the label text and the horizontal and vertical positions for each label.

"Add" Adds a line at the end of the list.

"Copy" Copies the selected line in a new line.

"Remove" Deletes the selected line. Only single lines can be removed.
You can also delete a label by using the toolbar: Tap the "Delete" icon and then the label.

Remote command:
`DISPlay:SIGNal:LABel:ADD` on page 1065
`DISPlay:SIGNal:LABel:REMove` on page 1066
`DISPlay:SIGNal:LABel:TEXT` on page 1066
`DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition` on page 1067
`DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition` on page 1067
`DISPlay:SIGNal:LABel:HORizontal:RELative:POSition` on page 1068
`DISPlay:SIGNal:LABel:VERTical:RELative:POSition` on page 1068

Show labels

Enables or disables the label display.

Position mode
Defines the label position either relative to the diagram or with absolute values according to the units of the waveform. Relative positions are fixed, whereas absolute positions move with the waveform display when the scales, the vertical position or offset, or the reference point are changed.

The position mode applies to all labels of the selected waveform. For different waveforms, different position modes can be selected.

"Relative"	Sets a fixed position in percent of the screen counting from the upper left corner.
"Absolute"	Sets the position in time and voltage values, or in other units depending on the waveform character.

Remote command:
`DISPlay:SIGNal:LABel:POSMode` on page 1066

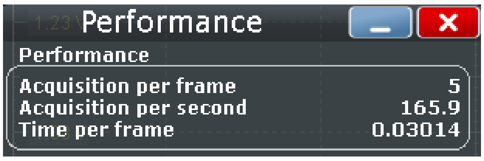
Label font size
Defines the size of the labels in the diagram.

Remote command:
`DISPlay:SIGNal:LABel:FONTsize` on page 1068

3.4.2.5 Performance

Access: "Display" menu > "Show Performance"

The "Performance" result box shows information on the current acquisition performance values of the R&S RTE.



The instrument groups acquired waveforms together in a frame, and displays the frame content. The maximum number of frames displayed per second is about 30. The current number of frames per second is indicated as reciprocal "Time per frame". If the time scale decreases, and thus the number of acquisitions per second also decreases, the number of acquisitions per frame can drop to 1.

3.4.2.6 Clear Results

"Display" menu > "Clear all"

"Clear all" resets all results in all measurement result boxes including long-term measurement and statistic results, and deletes all waveforms and the history.

If you need this function frequently, you can add the corresponding icon to the toolbar, see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.

Remote command:
- `DISPlay:CLR` on page 1068

3.4.3 Adjusting the Display

You can adjust the appearance of several display elements:
- Diagram name, see Chapter 3.4.3.1, "Changing the Diagram Name", on page 115.
- Waveform colors, see Chapter 3.4.3.2, "Changing Waveform Colors", on page 115.
- Icons on the toolbar, see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.
- Font size and transparency of dialog boxes and result boxes, see Chapter 3.4.3.3, "Configuring the Appearance of Dialog and Result Boxes", on page 117.
- Behaviour and position of the signal bar, see Chapter 2.4.6, "Using the Signal bar", on page 78.

3.4.3.1 Changing the Diagram Name

The diagram name is shown on the diagram tab.

► Double-tap the diagram tab name. The on-screen keyboard opens to enter the new name.

In remote operation, use`DISPlay:DIAGram:REName`.

3.4.3.2 Changing Waveform Colors

For each waveform, you can set a waveform color, or you define a color table that specifies which waveform points are displayed in which color. You can use one of the default color tables, or define your own table according to your needs. You can also edit the default color tables.

After you define a color table, you must assign it to the waveform it is to be used for, and enable its use.

The exact mapping of the cumulative value occurrences according to the assigned color table is guaranteed only if the intensity is set to 50%. All other intensity values falsify the mapping but may improve the visibility of the signal.
See also: "Intensity" on page 106.

For details on signal color settings, see Chapter 3.4.2.2, "Color Tables", on page 108.

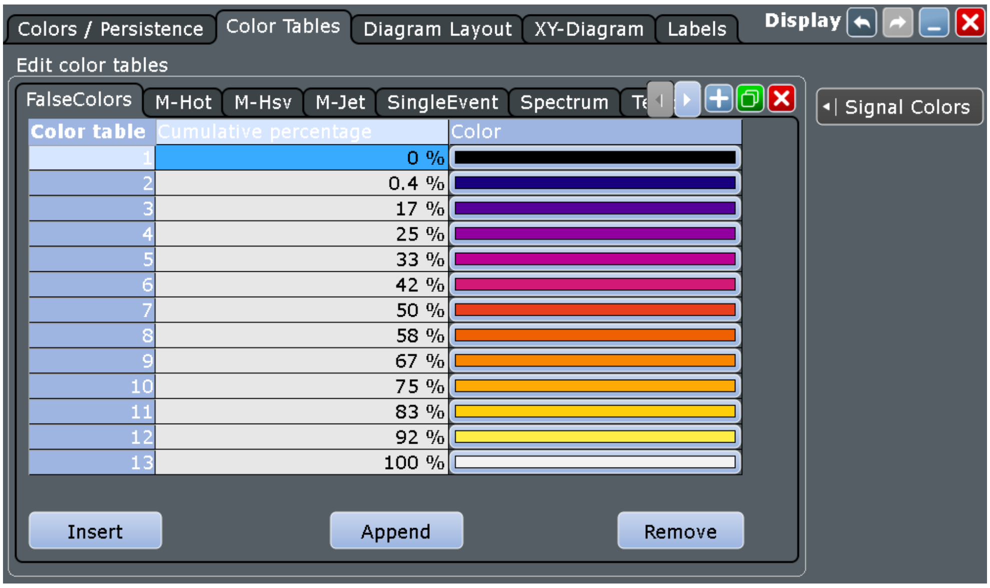
To change a waveform color

1. On the "Display" menu, tap "Signal Colors / Persistence".

2. Under "Color table assignment", select the waveform for which you want to change the color.
3. Tap the "Color" button.
4. In the "Adjust Colors" dialog box, select a predefined color, or define any other RGB color with "User defined Colors".

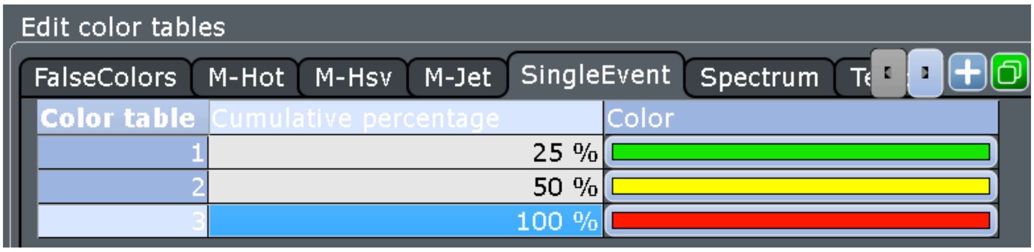
To edit a color table

1. On the "Display" menu, tap "Color Tables".
2. Under "Edit Color Tables", select the color table you want to edit.



3. For each range of cumulative occurrence of the values, insert an entry in the color table:
   - To insert an entry at the end of the color table, tap "Append".
   - To insert an entry before an existing entry, tap the existing row. Then tap "Insert".
   - To remove an entry, tap the entry. Then tap "Remove".
4. Assign a color to each entry: Tap the "Color" cell.
   Select a predefined color, or define your own color.

Example:



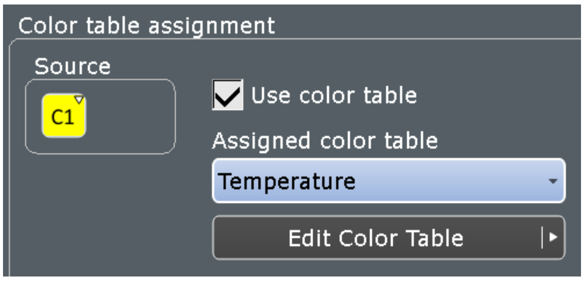
In this example, values with a cumulative occurrence under 25% (very short or rare display) are displayed green. Values with an occurrence of 40% are yellow-green. Values with an occurrence of 90% (displayed almost for the entire duration of the signal) are a deep shade of orange.

To create a color table

1. On the "Display" menu, tap "Signal Colors".
2. **To create an empty color table:** tap the "Add" button and enter a name for the new color table using the on-screen keyboard.
   To copy an existing color table: select the color table you want to copy, and tap the "Copy" button. Enter a name for the new color table using the on-screen keyboard.

To assign the color table and enable its use

1. Open the "Signal Colors/ Persistence" tab of the "Display" dialog box.
2. Under "Color Table Assignment", select the "Source" for the waveform.
3. Enable "Use Color table".

4. Under "Assign color table", select the color table you want to assign to the waveform.

   The waveform colors are displayed according to the definition in the color table.

3.4.3.3 Configuring the Appearance of Dialog and Result Boxes

You can optimize the display of dialog and result boxes so they do not interfere with the waveform display and you can still analyze the results and settings.

To change the font size in dialog and result boxes

1. Press [SETUP].
2. Select the "Screen" tab.
3. To set the font size in points for text in all dialog boxes, change "Font size". Most dialog boxes are optimized for a font size of 19 pt.
4. To set the font size in points for result boxes, change "Result dialog font size". The default is 12 pt.

To change the transparency of dialog boxes and result boxes

The transparency of the dialog box background lets you see the waveforms behind the box. You can configure the transparency separately for dialog boxes and result boxes.

1. Press [SETUP].
2. In the "Screen" tab, in the "Dialog box transparency" field, enter the transparency value for dialog boxes.
   For high transparency values, you can see the waveform display in the background, and possibly check the effect of the changed setting. For lower transparency values, readability in the dialog box improves.
3. In the "Result box transparency" field, enter the transparency value for result boxes.

Alternatively, you can press the [INTENSITY] knob until the required parameter is shown in the data input box, and then turn the knob to set the transparency.

3.4.3.4 Using the Signal bar

The signal bar can hold a large number of signal and result icons. Signal icons represent the waveforms, serial buses and parallel buses, while result icons are minimized result boxes showing measurement and search results.

To scroll the signal bar

If the signal bar contains more than four icons, not all icons are visible on the display.

► Touch one of the signal icons and move it up or down until the required icon appears.

To switch on and off the signal bar

If you need the complete screen to see the diagrams and results, you can switch off the signal bar completely.

► Tap the "Show signal bar" icon on the toolbar.

Alternatively, tap "Signal Bar" on the "Display" menu.

To change the position of the signal bar

► Touch the "Horizontal" label on the top of the signal bar and drag it to the opposite side of the screen.

To configure auto-hide

The signal bar can be hidden if the displayed information has not changed for a defined time, and is displayed again automatically when a setting in the signal bar changes. The signal bar does not hide entirely, it simply fades and becomes less visible in the display.

1. Press the [DISPLAY] key on the front panel.
2. In the "Display" dialog box, select the "Diagram Layout" tab.
3. Select "Auto-hide".
4. Define the hiding properties:
   - "Hide bar after": the time after which the bar is hidden if no changes occur
   - "Hiding transparency": Transparency of the hidden signal bar on a scale from 20% (low transparency) to 70% (high transparency)
   - Hide head also: the horizontal and trigger labels are also faded

To change the colors

If you want to highlight the signal bar, you can change the "Fill color" and "Border color" of the bar.

1. Press the [DISPLAY] key on the front panel.
2. In the "Display" dialog box, select the "Diagram Layout" tab.
3. Tap "Border color" to change the color of the signal bar frame, or "Fill color" to change the fill color of the bar.
4. In the "Adjust Colors" dialog box, select the color.
5. To use a color that is not yet defined, tap "Userdefined Colors". Define the new color settings.
   To see the effect of a setting change in the "Preview" area, enter the value and press the [ENTER] key.
6. Tap "OK."

   The signal bar is displayed in the new colors.

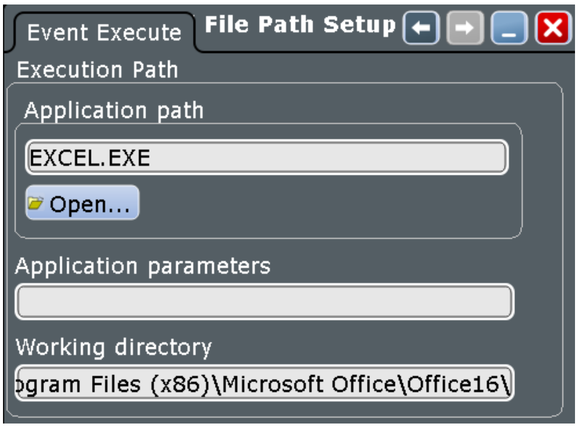
3.5 External Application

Access: "File" menu > "External Setup"

The R&S RTE can start an external application on the instrument or in the network (if connected) when an event occurs.

The following events can start an application:

- Trigger event
- Mask violation
- Successful completion of mask test



Set the path of the application executable, optional parameters, and the working directory as in a Windows shortcut definition. The setup is valid for all events.

Remote commands:

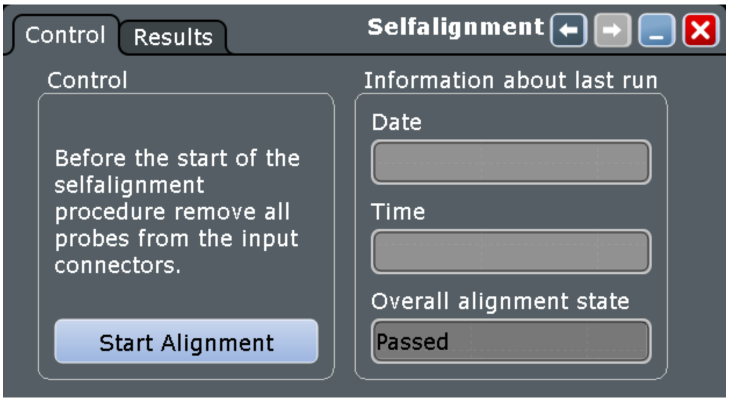
- `EXECutable:NAME` on page 1069
- `EXECutable:PARameter` on page 1069
- `EXECutable:WDIRectory` on page 1069

3.6 Self-alignment

When data from several input channels is displayed at the same time, it may be necessary to align the data vertically or horizontally to synchronize the time bases or amplitudes and positions. This is the case, for example, when strong temperature changes occur (> 5°).

3.6.1 Control

Access: "File" menu > "Selfalignment"



Start Alignment
Starts the self-alignment procedure for all channels.

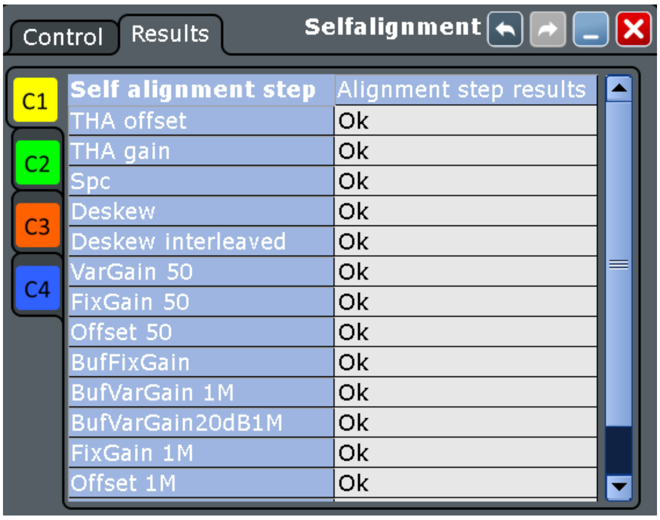
Remote command:
*CAL? on page 1040

Date / Time / Overall alignment state
Show the date, time and the summary result of the self-alignment process: not aligned, passed or failed. Detailed results are provided on the "Results" tab.

Remote command:
CALibration:DATE? on page 1946
CALibration:TIME? on page 1947
CALibration:RESult? on page 1947

3.6.2 Results

For each channel, the results of the individual alignment steps are shown for all technical channel component. In case you require support, you may be asked to provide this information.



3.6.3 Performing a Self-alignment

The self-alignment aligns the data from several input channels vertically and horizontally to synchronize the timebases, amplitudes and positions. The self-alignment process includes a basic hardware check.

Recommendation on performing the self-alignment:

- When putting the instrument into operation for the first time
- After a firmware update
- Once a week
- When major temperature changes occur (> 5°)

1. Warm up the instrument before you start the self-alignment. The minimum warm-up time is indicated in the data sheet.
2. Remove the probes from the input connectors.
3. On the "File" menu, select "Selfalignment".
4. On the "Control" tab, tap "Start Alignment".

   The alignment is performed, the process might take several minutes. A message box informs you about the running process, wait until this message box closes. The overall pass/fail result is shown in the "Overall alignment state" field. The results of the individual alignment steps for each input channel are indicated in the "Results" tab. This information is required if problems arise.

3.7 Self-test

The instrument's self-test checks the hardware for correct operation. Perform the self-test if you suspect problems in hardware operation.

► "File" menu > "Selftest".

The test can take several minutes. The summary result is shown in the "State" field, which can be helpful in case you need support.

Selftest
Starts the self-test.

Remote command:
`*TST?` on page 1044

State
Shows the summary result of the self-test: Pass or Fail.

Remote command:
`DIAGnostic:SERVice:STST:STATe?` on page 1947

Result
Opens a log file with detailed information on the self-test steps and operation of hardware components, which can be helpful in case you need support.

3.8 Firmware Update

Your instrument is delivered with the latest firmware version. Firmware updates are provided on the internet at:

www.rohde-schwarz.com/firmware/rte.

The "Release Notes" describe the improvements and modifications of all firmware versions ans also how to update the firmware. They are available along with the firmware on the same web page.

3.9 Options

Additional options for the R&S RTE can be enabled using a license key. To obtain the license key, consult your sales representative.

The license type defines the duration of applicability and the portability of a license. The following license types are provided: evaluation, permanent, portable, quantified, timed with duration of 1, 3, 6 or 12 months. A license can also be in the states deactivated and expired.

Unregistered licenses
Unregistered licenses are not assigned to a particular instrument. The instrument accepts only registered licenses. If your license is delivered unregistered, use the online tool R&S License Manager to register the license for your instrument. The registration of a permanent license is irreversible, so ensure that you register it for the correct instrument. The address of the tool is https://extranet.rohde-schwarz.com/service.

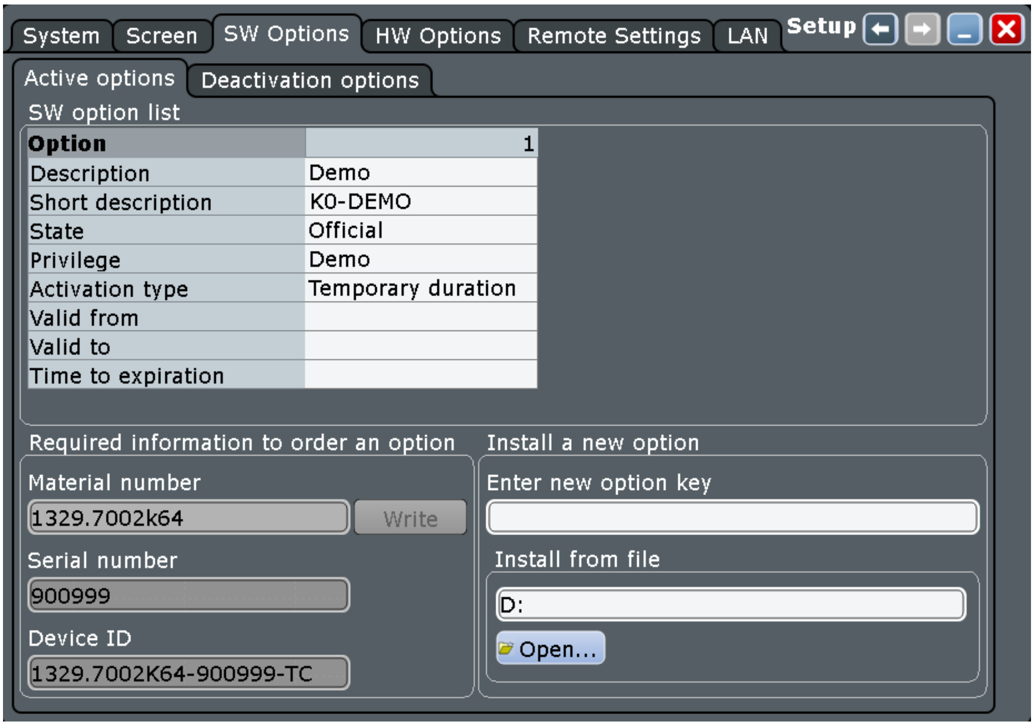
The R&S License Manager also allows you to move a portable license to another instrument.

3.9.1 SW Options

3.9.1.1 Active Options Settings

Access: [SETUP] > "SW Options" tab > "Active options" subtab

The "Active options" tab provides information on installed software options. Here you can install new options or deactivate existing options using license keys.



SW option list

Shows the activated options. This information provided for administration and trouble-shooting purposes. If you need support for an option, provide this information to the service representative.

The "State" of the option indicates whether the installed option is a normal or a beta-release version. Beta-release versions must be activated explicitly in the "Mode" dialog box (see Chapter 3.9.4, "Options in Beta State", on page 127).

Material number, Serial number, Device ID

Indicates the material number, serial number and the device identification string (device ID) of your instrument. These numbers, in particular the device ID, are required to order a new option, or to move a portable option.

Remote command:

`DIAGnostic:SERVice:PARTnumber` on page 1050
`DIAGnostic:SERVice:SERialnumber?` on page 1050
`SYSTem:DEVice:ID?` on page 1049

Enter new option key

Enter the license key here to activate the option. For license keys delivered as a file, use Install from file. Only users with administrator rights can activate options.

Install from file

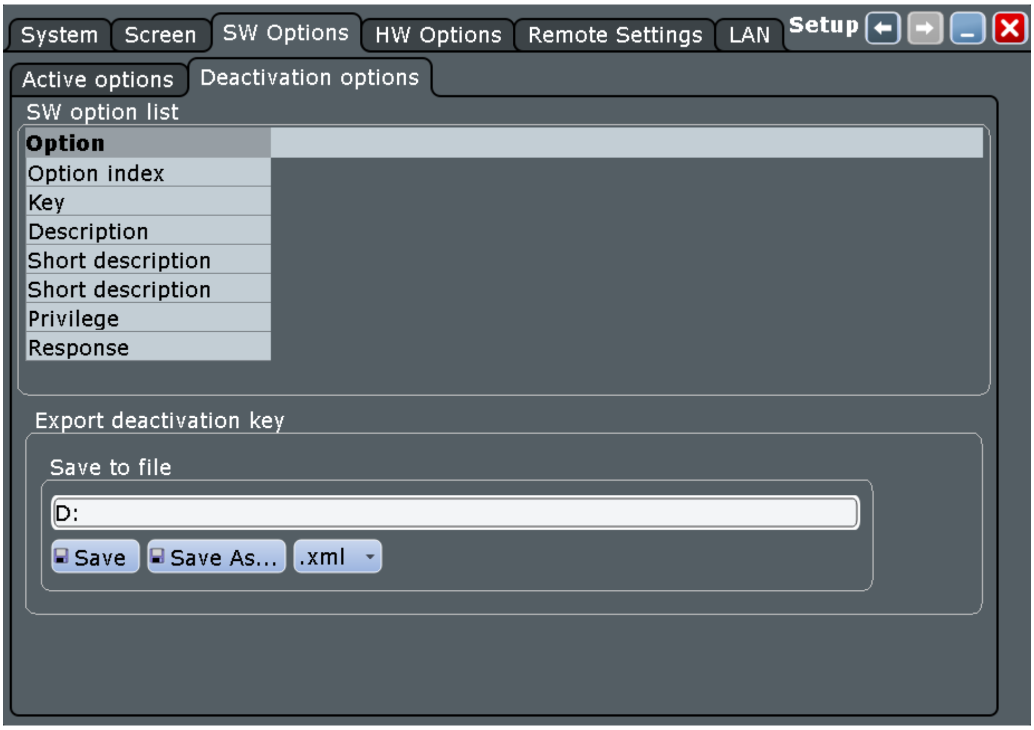
If you got a license file, install the license here. Tap "Open" to open the file selection dialog, or enter the complete path and filename. For details, see Chapter 11.7, "File Selection Dialog", on page 471. Only users with administrator rights can activate options.

When you move a portable license, use this function to import the deactivation key that is generated by the "R&S License Manager". See also Chapter 3.9.1.3, "Using a License Server", on page 125.

3.9.1.2 Deactivation Options

Access: [SETUP] > "SW Options" tab > "Deactivation options" subtab

The "Deactivation options" tab shows all deactivated options and provides a function to export the deactivation response.



Export deactivation key
When you move a portable license, or deactivate an option, you have to note the response key, or to save the response to a file. The "R&S License Manager" needs the response key.

3.9.1.3 Using a License Server

Software licenses can also be provided by the R&S®License Server. In this case, all available licenses are stored on a specified server. When you need a particular application, you can obtain a license from the server. When you no longer need the license, you return it to the server and it becomes available to other users again. These licenses are also referred to as *floating licenses*, as opposed to permanent licenses.

The R&S®License Server must be set up as described in the R&S®License Server - Managing Floating Licenses - User Manual.

You can find the manual in the Windows menu of your instrument: "Start > All Programs > R&S License Server > Open License Server Manual".

To open the License Server

► In the Windows notification area, select , and then "Open Manager".

a) Optional: If the status icon (, or) in the Windows notification area is not shown yet, click "Start > All Programs > R&S License Server > License Server Manager"".
The icon is now available in the Windows notification area.

b) Optional: If the status is (stopped), select the icon, and then select "Start License Server". Then, select > again.

The license server opens in the default web browser. The program adds all available Rohde & Schwarz products (smart cards or devices) to a list and automatically shows the first product that is detected. By default, the "Licenses" list opens with an initial filtering for showing only active licenses.

3.9.2 HW Options

This tab informs about the availability of hardware options.

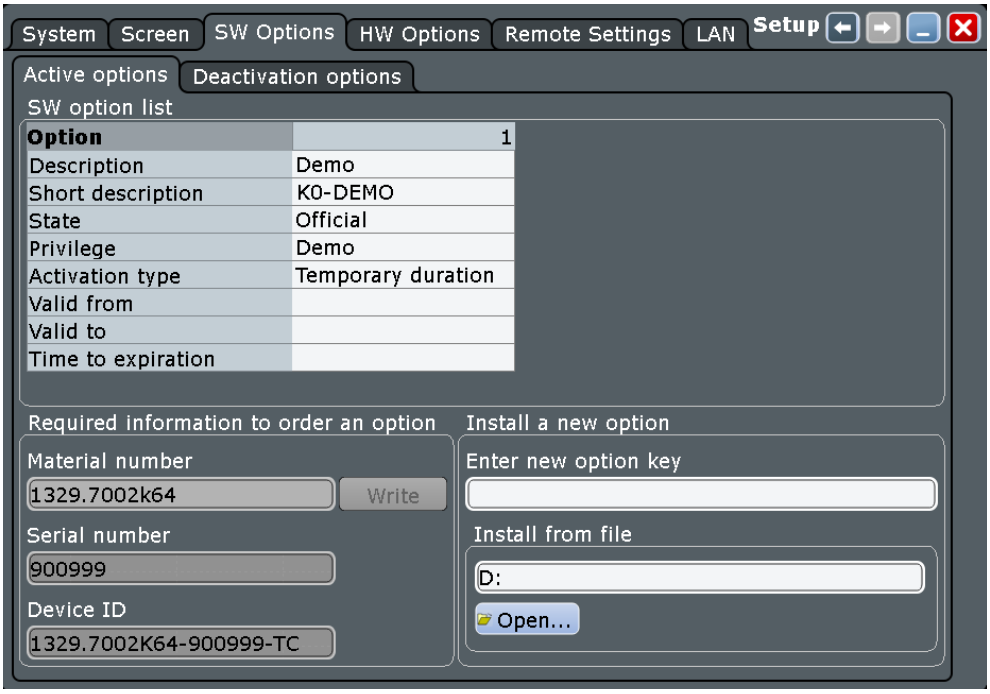
3.9.3 Activating Options

Options are activated by license keys. No additional installation is required. Consult your sales representative and provide the material number and serial number (or the device ID) of your instrument to get a license key. The license key is provided in written form or in a file. Unregistered licenses must be registered in the R&S License Manager before they can be activated on the instrument.

If the option has a portable license, keep the license file or option key at a save place. You need the license to move it to another instrument.

Only users with administrator rights can activate options.

1. Press the [SETUP] key and select the "SW options" tab.



2. If you received a key in written form, enter the key in the "Enter new option key" field.
   If you received a key in digital form as a file, tap "Open", navigate to the directory that contains the file, and select the option key file.
3. If you want to activate several options, repeat step 3 for each option.
4. Restart the instrument or the firmware.

See also: Chapter 3.9.1, "SW Options", on page 123

3.9.4 Options in Beta State

Options may be released in beta state. These options require a license key and an additional activation.

To activate a beta option:

1. On the "File" menu, select "Mode".
2. Select the "Operation Mode" tab.
3. Enable "Enable options in beta state".

The activation is effective immediately until the next shut-down of the firmware.

4 Acquisition and Waveform Setup

This chapter describes the horizontal and vertical settings as well as the acquisition and probe setup.

4.1 Basics

This chapter provides background information on the essential settings in the vertical and horizontal systems, on acquisition setup and probing.

4.1.1 Vertical System

The controls and parameters of the vertical system are used to scale and position the waveform vertically.

4.1.1.1 Input Coupling

The input coupling influences the signal path between input connector and the following internal signal stage. The coupling can be set to DC, AC, or ground.

- DC coupling shows all parts of an input signal. DC coupling is available with 1 MΩ input impedance to connect standard passive probes. DC 50 Ω coupling is the default for 50 Ω input impedance to connect, for example, active probes.
- AC coupling is useful if the DC component of a signal is of no interest. AC coupling blocks the DC component of the signal so that the waveform is centered on zero volts.
- Ground coupling disconnects the input signal from the vertical system to see the ground level (zero volts) on the screen. Ground coupling is useful for reference purposes.

4.1.1.2 Vertical Scale and Position

Vertical scale and vertical position directly affect the resolution of the waveform amplitude. The vertical scale corresponds to the ADC input range. To get the full resolution of the ADC, set up the waveforms to cover most of the height of the diagram.

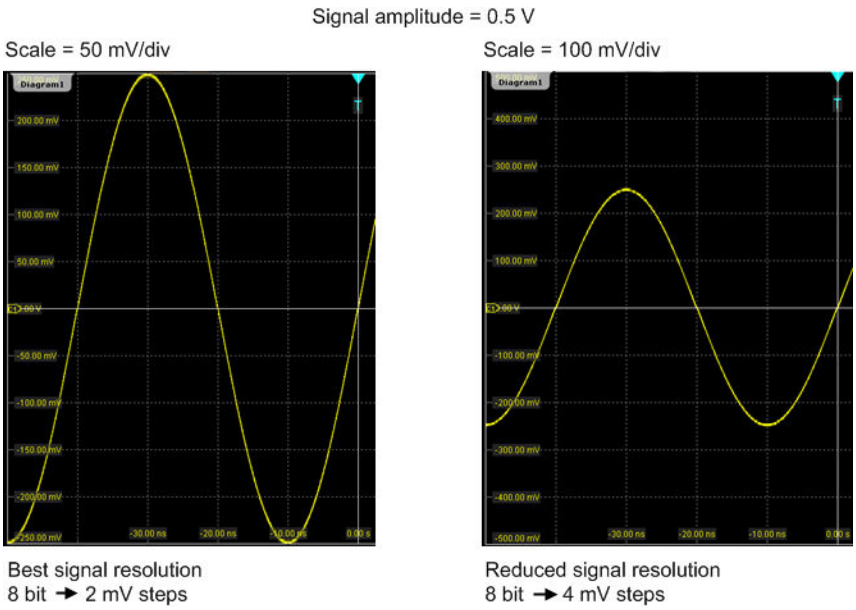


Figure 4-1: Input range and resolution of the ADC

With R&S RTE, you can work with multiple diagrams, and each diagram obtains the full vertical resolution, no matter where the diagram is placed. Therefore, use a separate diagram for each waveform instead of the traditional setup that arranges the waveforms side by side in one diagram.

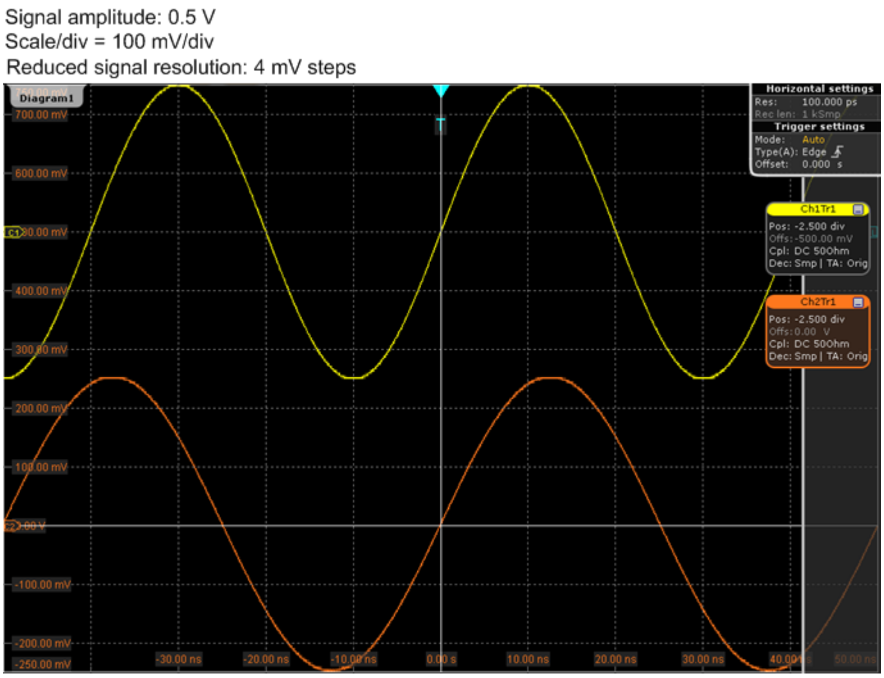


Figure 4-2: Traditional setup of multiple waveforms in one diagram: reduced resolution

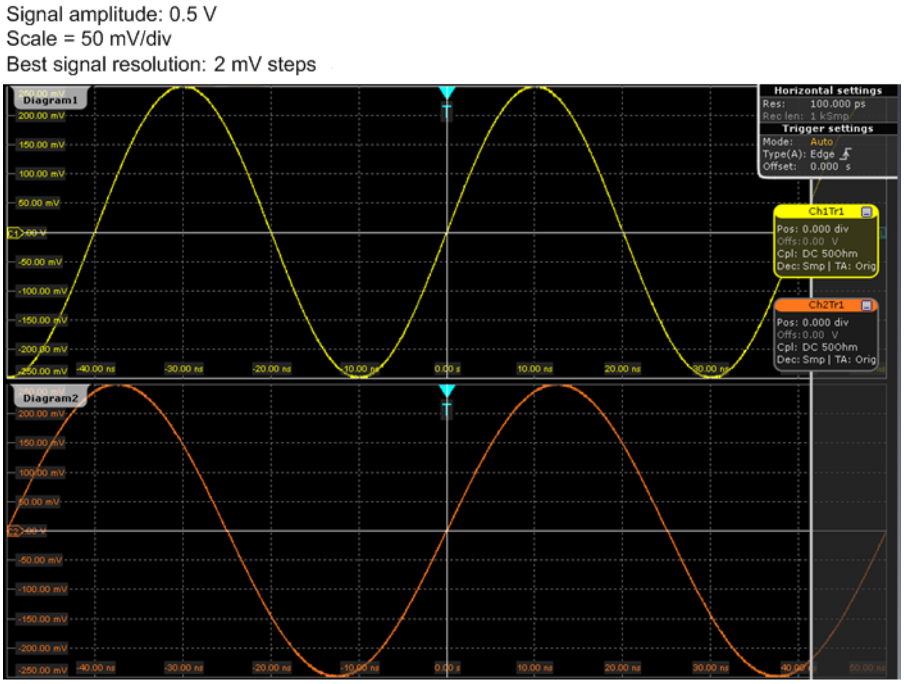


Figure 4-3: R&S RTE setup of multiple waveforms in separate diagrams: best resolution

4.1.1.3 Bandwidth

For analog applications, the highest signal frequency determines the required oscilloscope bandwidth. The oscilloscope bandwidth should be slightly higher than the maximum frequency included in the analog test signal to measure the amplitude with very little measurement error.

Most test signals are more complex than a simple sine wave and include several spectral components. A digital signal, for example, is built up of several odd harmonics. As a rule, for digital signals the oscilloscope bandwidth should be 5 times higher than the clock frequency to be measured.

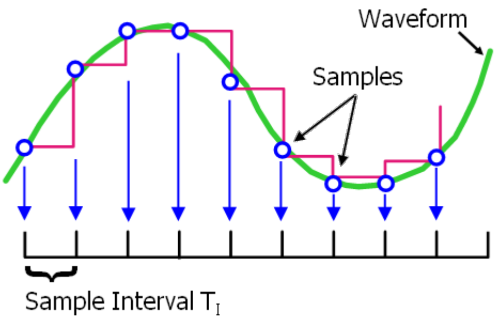
The oscilloscope is not a stand-alone system. You need a probe to measure the signal of interest, and the probe has a limited bandwidth, too. The combination of oscilloscope and probe creates a *system bandwidth*. To reduce the effect of the probe on the system bandwidth, the probe bandwidth must exceed the bandwidth of the oscilloscope, the recommended factor is 1.5 x oscilloscope bandwidth. See also: Chapter 4.1.4.1, "Voltage Probes", on page 135

4.1.2 Sampling and Acquisition

The vertical system of a digital oscilloscope conditions the test signal in a way that the following A/D converter (ADC) can transform the measured voltage into digital data.

4.1.2.1 Sampling and Processing

The A/D converter samples the continuous signal under test at specific points in time and delivers digital values called **ADC samples**. The rate at which the converter is working is the **ADC sample rate**, a constant value specified in GHz: $f_{ADC} = 1 / T_I$



The digital ADC samples are processed according to the acquisition settings. The result is a waveform record that contains **waveform samples** and is stored in the **waveform memory**. The waveform samples are displayed on the screen and build up the waveform.

The number of waveform samples in one waveform record is called **record length**. The rate of recording waveform samples - the number of waveform samples per second - is the **sample rate**. The higher the sample rate, the better the resolution is and the more details of the waveform are visible.

A sufficient resolution is essential for correct reconstruction of the waveform. If the signal is undersampled, aliasing occurs - a false waveform is displayed. To avoid aliasing and accurately reconstruct a signal, Nyquist theorem postulates that the sample rate must be at least twice as fast as the highest frequency component of the signal. However, the theorem assumes ideal conditions, so the Nyquist sample rate is usually not sufficient.

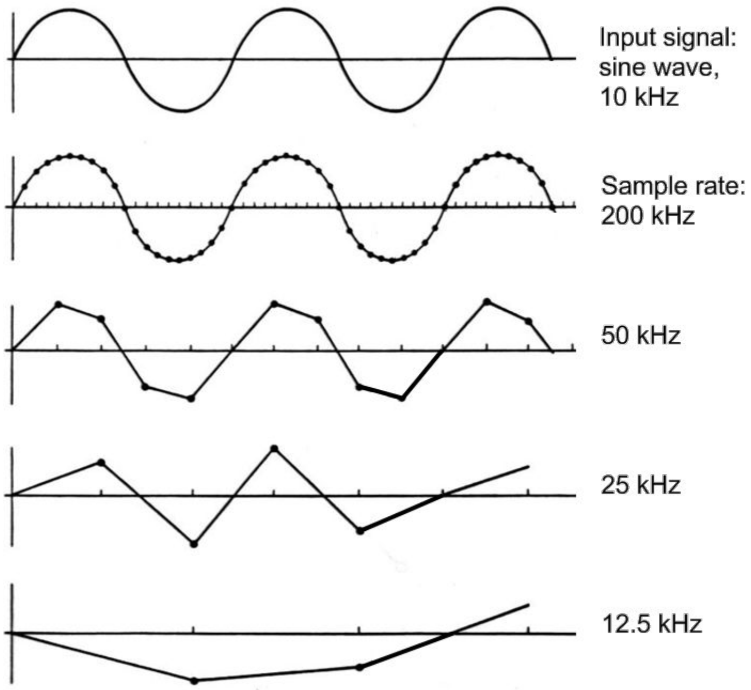


Figure 4-4: Waveforms acquired with different sample rates

To avoid aliasing, the sample rate must be set to a value 3 to 5 times the fastest frequency component of the signal. A higher sample rate increases signal fidelity, increases the chance to capture glitches and other signal anomalies, and improves the zoom-in capabilities.

4.1.2.2 Acquisition Settings

The sample rate can be the same as the constant ADC sample rate, or higher, or lower. To get a higher sample rate, interpolation as method of **resolution enhancement** is used. To reduce the sample rate, **decimation** methods help: sample,peak detect, high resolution and RMS.

As digital waveform data is stored in the memory, and the memory can save many waveform records, further **waveform arithmetic** processing is possible: average and envelope waveforms are resulting waveforms, created from a composite of sample points taken from multiple acquisitions.

You can combine interpolation and waveform decimation modes with waveform arithmetic.

4.1.2.3 Acquisition Control

You can run the R&S RTE in two ways:

- RUN CONT: the instrument acquires data until you stop it manually.
- RUN N× SINGLE: the instrument samples and processes a specified number of acquisitions.

The determining point of an acquisition is the trigger. The instrument acquires continuously and keeps the sample points to fill the pre-trigger part of the waveform record. When the trigger occurs, the instrument continues acquisition until the post-trigger part of the waveform record is filled. Then it stops acquiring and waits for the next trigger. When a trigger is recognized, the instrument does not accept another trigger until the acquisition is complete.

The trigger modes define how the instrument triggers:

- Normal: The instrument acquires a waveform only if a real trigger occurs, that is, if all trigger conditions are fulfilled.
- Auto: The instrument triggers repeatedly after a fixed time interval if the trigger conditions are not fulfilled. If a real trigger occurs, it takes precedence. If the real trigger is faster than the auto trigger, both modes are virtually the same.

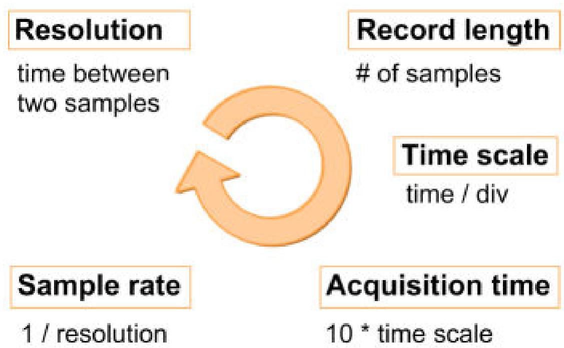
In practice, both trigger modes are useful: The auto mode lets you see the signal with little adjustment, while the normal mode selects the interesting part of the waveform. If you want to acquire a specified number of waveforms, make sure to select the normal trigger mode. Thus you get only the required number of interesting acquisitions.

See also: Chapter 5, "Triggers", on page 200

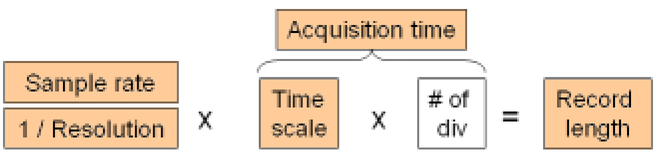
4.1.3 Horizontal System

4.1.3.1 Parameters of the Horizontal System

The control parameters of the horizontal system are tightly connected. Thus, changing one parameter affects the other parameters as well.



The mathematical dependencies can be summarized as follows:



The number of divisions is 10, which is the only constant parameter.

When you set up horizontal parameters, you can choose whether the record length or the resolution remains constant.

- With constant resolution, increasing the time scale also increases the record length, and vice versa. You can limit the record length to a maximum value.
- With constant record length, increasing the time scale coarsens the resolution, that is, the time between two waveform samples gets longer.

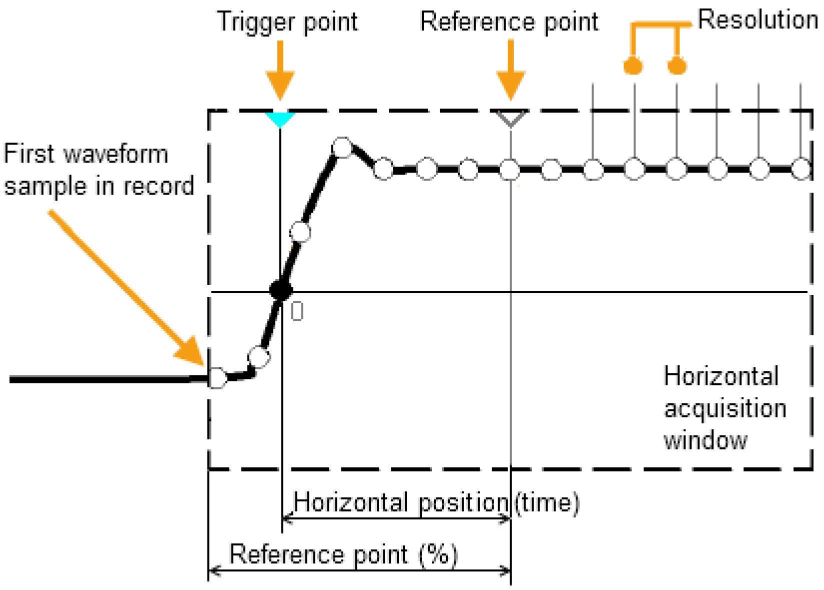
For both settings, the "Auto adjustment" ensures a sufficient resolution to prevent undersampling.

4.1.3.2 Horizontal Position

As described before in Chapter 4.1.2.3, "Acquisition Control", on page 132, the trigger is the determining point of the waveform record.

In many scenarios, you want to analyze the waveform some time before or after the trigger. To adjust the horizontal acquisition window to the waveform section of interest, you can use the following parameters:

- The **horizontal position** defines the time distance from the trigger point (the zero point of the diagram) to the reference point. Changing the horizontal position, you can move the trigger point, even outside the screen.
- The **reference point** is the rescaling center of the time scale on the screen. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.



4.1.4 Probes

A probe connects the signal source (DUT) to the oscilloscope, and delivers the signal to be measured. It is the essential first link in the measurement chain.

An ideal probe fulfills the following requirements:

- Safe and reliable contacts
- Infinite bandwidth
- The probe should not load the signal source and thus impact the circuit operation.
- The connection should not introduce or suppress signal components (hum, noise, filter) and thus degrade or distort the transferred signal.

In reality, the probe can never be an ideal one, it always affects the signal transmission and the signal source, and thus the measured signal. It depends on the frequency to be measured and on the signal source to determine the acceptable loading, and to determine which kind of probe delivers good results.

The solution depends on the quantity to be measured regarding:

- Signal type: voltage, current, power, pressure, optical, etc.
- Signal amplitude: The oscilloscope itself can only display voltages in a limited range. Most probes can adjust the dynamic range to amplitudes from a few mV to 10 V. Smaller or much larger signals require specialized equipment.
- Signal frequency: High frequencies require advanced equipment to get correct results.
- Source characteristic: The source impedance is the decisive factor when choosing the suitable connection.

4.1.4.1 Voltage Probes

The following table provides an overview on common voltage probes and their usage.

Table 4-1: Voltage probes overview

Probe type	Attenuation	Typical bandwidth range	Oscilloscope input	Usage
Passive, high impedance	1:1	10 MHz	1 MΩ	Low-speed signals, low-level signals
Passive, high impedance	10:1	500 MHz	1 MΩ	General purpose
Passive, low impedance	10:1	up to 10 GHz	50 Ω	High frequency
Active, single-ended	10:1	up to 10 GHz	50 Ω	High speed
Active, differential	10:1		50 Ω	Floating

For a list of recommended probes, refer to the R&S RTE product brochure.

Besides the possible input voltage range, two factors are important when selecting a voltage probe: Bandwidth and impedance over frequency.

- **Bandwidth:**
  The combination of probe and oscilloscope builds up a system. The resulting system bandwidth is approximately determined with:

$$\frac{1}{BW_{system}} = \sqrt{\left(\frac{1}{BW_{probe}}\right)^2 + \left(\frac{1}{BW_{scope}}\right)^2}$$

To measure the signal with low measurement error, the system bandwidth should be higher than the highest frequency component of the signal. The probe bandwidth must be even higher than the system bandwidth.

- **Impedance:**
  A minimum impedance is required to keep the circuit loading low. Over frequency, the impedance decreases, in particular with passive probes. The probe impedance should be approximately 10 times the impedance of the circuit test point at the highest signal frequency.

Passive Voltage Probes

Passive probes have the following qualities:

- No active components inside
- BNC connector for universal use
- Compensate the probe when it is connected to a scope input: LF compensation matches the probe (mainly cable) capacitance to the oscilloscope input capacitance.
- With high impedance probes, the impedance varies significantly over frequency.
- With low impedance probes, the impedance variation over frequency is low, but the load on the source is high.

If you use passive probes, remember some recommendations:

- Use a probe recommended for your oscilloscope model.
- Use a ground lead as short as possible to minimize the effect of ground lead inductance. The resonance frequency can be much lower than the system bandwidth and thus can affect the measurement results, in particular, if you measure steep edge rise times.
- Select a probe that has a bandwidth of 5 to 10 times the highest frequency being measured. This bandwidth preserves the harmonics and thus the waveform integrity.

Active Voltage Probes - General

Active probes require operating power from the instrument and have a proprietary interface to the instrument. Their main qualities are:

- Low loading on signal source
- The probe is automatically recognized by the instrument, no adjustment is required.
- Adjustable DC offset at probe tip allows for high resolution on small AC signals which are superimposed on DC levels.
- Connections should be as short as possible to keep the usable bandwidth high.
- Observe the operating voltage range.
- The probe impedance depends on the signal frequency.

RT-ZS single-ended active probes and RT-ZD differential active probes provide special features for easier use and precise measurements. These special features are not available on RT-ZSxxE probes.

- The micro button on the probe head remotely controls important functions on the instrument, like running and stopping the acquisition, autoset, AutoZero and setting the offset to mean value.
- The R&S ProbeMeter measures DC voltages between the probe tip and the ground connection with very high precision. The result is displayed on the instrument's screen. So you can check DC voltages with different levels without having to adjust the measurement range of the oscilloscope. The R&S ProbeMeter also measures the zero error of the probe to optimize measurement results at small signal levels.

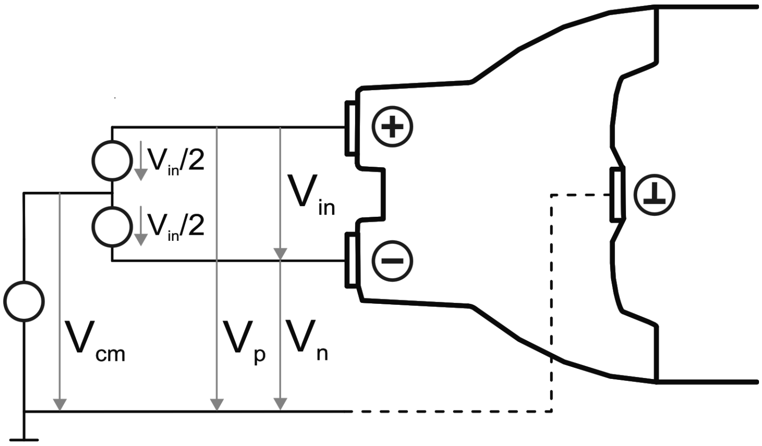
When you connect an R&S RT-ZSxx active probe to a channel input of the R&S RTE, the oscilloscope recognizes the probe. It reads the identification and calibration data from the probe box and shows the result in the "Setup" and "Probe Attributes" tabs. This data together with the deskew time for a given channel is stored and processed by the R&S RTE. If you connect the probe the next time to the same channel, the information is fetched and used.

Differential Active Probes

Differential active probes are designed to measure signals that are referenced against each other, and voltages that are not references to ground, for example twisted-pair signal lines. The R&S RT-ZD probes are differential probes with high input impedance, they can be used to measure voltages between any two test points.

Compared with two-channel measurement setup with single-ended probes, the measurement with differential probes is symmetric due to the same amplification and cable length on both paths. It is also immune to interference and noise and occupies only one input channel.

A differential probe has three sockets: the positive signal socket (+), the negative signal socket (-), and the ground socket.



Multiple input voltages can be defined for a differential probe:

- Differential mode input voltage (V_{in}, V_{dm})

Voltage between the positive and negative signal sockets

- Positive single-ended input voltage (V_p)
  Voltage between the positive signal socket and the ground socket
- Negative single-ended input voltage (V_n)
  Voltage between the negative signal socket and the ground socket
- Common mode input voltage (V_{cm})
  Mean voltage of positive and negative signal sockets referred to the ground socket, respectively

Two of these voltages are independent values, the other two can be calculated:

$$V_{in} = V_p - V_n$$

$$V_{cm} = \frac{V_p + V_n}{2}$$

R&S RT-ZD probes detect only differential input voltages and provide it to the oscilloscope. Common mode signals are suppressed by the probe. This characteristic is described by the common mode rejection ratio (CMRR):

$$CMRR = \frac{DifferentialGain}{CommonMode\,Gain}$$

In addition, the R&S ProbeMeter of R&S RT-ZD differential probes can measure differential and common mode DC voltages. The measurement result is displayed on the oscilloscope's screen. The common mode measurement of the R&S ProbeMeter allows you to check the input voltage relative to ground. Thus, the CM measurement is a convenient way to detect breaches of the operating voltage window, and the reason of unwanted clippings.

4.2 Horizontal Settings

The "Horizontal" menu provides the time base and acquisition configuration for channel and spectrum waveforms:

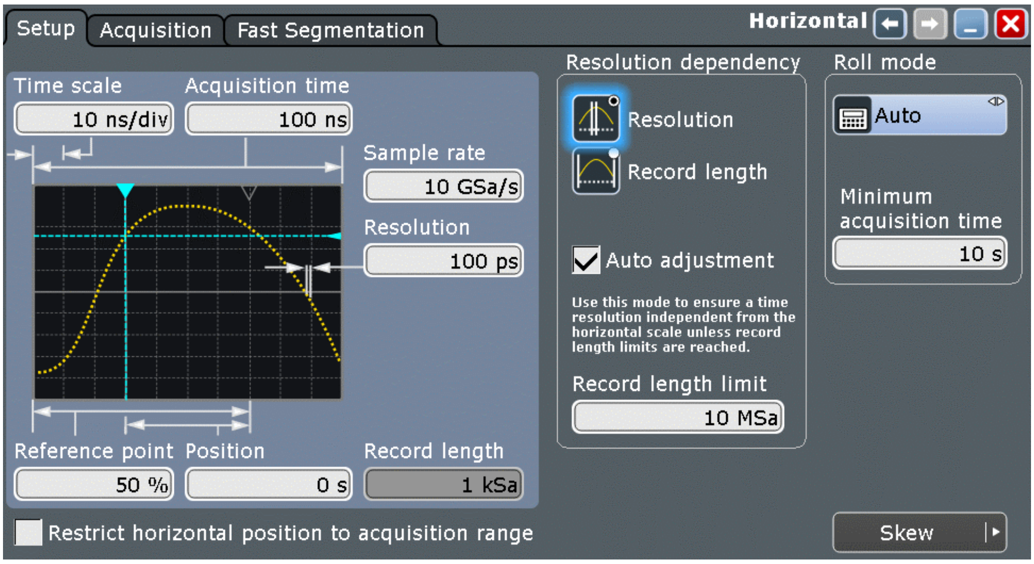
- Setup......138
- Acquisition......143
- Fast Segmentation......145

4.2.1 Setup

Access: HORIZONTAL key

The "Setup" tab in the "Horizontal" dialog box provides the settings for the time axis and the roll mode.

For background information, see Chapter 4.1.3, "Horizontal System", on page 133.



Time scale....139
Acquisition time....139
Reference point....140
Position....140
Restrict horizontal position to acquisition range....140
Sample rate....140
Resolution....140
Record length....140
ZVC Resolution....141
ZVC Record length....141
Resolution / Record length (Resolution dependency)....141
Auto adjustment (Resolution dependency)....141
Record length limit (Resolution dependency)....141
Roll mode....142
└ Minimum acquisition time....142

Time scale
Sets the horizontal scale for all channel and math waveforms in seconds per division. Increase the scale to see a longer time interval of the waveform. Decrease the scale to see it in more detail. The scale has a point that remains fixed on the screen when the scale value is changing - the reference point.

Remote command:
`TIMebase:SCALe` on page 1071

Acquisition time
Shows the time of one acquisition, that is the time across the 10 divisions of the diagram:

Acquisition time = Time scale * 10 divisions

Changing the acquisition time changes the time scale too.

Remote command:
`TIMebase:RANGe` on page 1071

Reference point

Sets the position of the reference point in % of the screen. The reference point marks the rescaling center of the time scale. It is indicated by a grey triangle outline at the top of the diagram. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.

Remote command:
`TIMebase:REFerence` on page 1072

Position

Defines the time distance between the reference point and the trigger point (the zero point of the diagram). If you want to see a section of the waveform some time before or after the trigger, enter this time as horizontal position. The requested waveform section is shown around the reference point. Use positive values to see waveform sections after the trigger - the waveform and the diagram origin move to the left.

See also "Reference point" on page 140.

Remote command:
`TIMebase:HORizontal:POSition` on page 1072

Restrict horizontal position to acquisition range

If enabled, the horizontal position cannot be set outside the visible waveform diagram.

Remote command:
`TRIGger<m>:OFFSet:LIMited` on page 1072

Sample rate

Sets the number of captured waveform points per second. It considers the samples of the ADC, and the reduction of waveform points by decimation.

If interpolation is not active, the sample rate is the reciprocal value of the resolution and thus also depends on the acquisition time and the record length.

If interpolation is active, the sample rate is limited to the ADC sample rate.

See also:
- Chapter 4.1.2, "Sampling and Acquisition", on page 130
- Chapter 4.1.3, "Horizontal System", on page 133

Remote command:
`ACQuire:SRReal` on page 1074

Resolution

Sets the time between two waveform samples. A fine resolution with low values produces a more precise waveform record.

Remote command:
`ACQuire:RESolution` on page 1075

Record length

Indicates the number of waveform samples that build the waveform across the acquisition time.

Remote command:
`ACQuire:POINts[:VALue]` on page 1075

ZVC Resolution
Available only, if a R&S RT-ZVC multi-channel power probe probe is connected to the instrument.

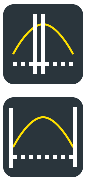
Displays the current resolution, the time between two waveform samples, of the R&S RT-ZVC multi-channel power probe channel.

Remote command:
`ACQuire:ZRESolution?` on page 1118

ZVC Record length
Available only, if a R&S RT-ZVC multi-channel power probe probe is connected to the instrument.

Indicates the number of waveform samples that build the channel's waveform across the acquisition time.

Remote command:
`ACQuire:POINts:ZVALue?` on page 1118



Resolution / Record length (Resolution dependency)
You can choose to keep constant either the resolution or the record length when you adjust the time scale or acquisition time.
- With constant resolution, increasing the time scale also increases the record length, and vice versa. You can limit the record length to a maximum value.
- With constant record length, increasing the time scale coarsens the resolution, that is, the time between two waveform samples gets longer.

Remote command:
`ACQuire:POINts:AUTO` on page 1073

Auto adjustment (Resolution dependency)
Prevents undersampling and ensures a sufficient resolution to acquire the correct waveform if the time scale is changed. The setting takes effect if the changed parameter - resolution or record length - reaches a limit. The instrument automatically keeps this parameter constant at its limit, and changes the other parameter regardless of the "Resolution / Record length" setting.

See also: Resolution / Record length (Resolution dependency)

Remote command:
`ACQuire:POINts:AADJust` on page 1073

Record length limit (Resolution dependency)
Sets a limit for the record length to prevent very large records. This value is only available if "Auto adjustment" is on and a constant resolution is selected. If you increase the time scale, the resolution remains constant and the record length increases until the limit is reached. Further increase of the time scale changes the resolution and keeps the record length limit.

See also:
- Resolution / Record length (Resolution dependency)
- Auto adjustment (Resolution dependency)

Remote command:
`ACQuire:POINts:MAXimum` on page 1074

Roll mode

In roll mode, the instrument shows the waveforms immediately, without waiting for the complete acquisition of the waveform record. If the time base is slow - at long time scale values - the roll mode saves waiting for the waveform display. The instrument displays newly acquired waveform points at the right edge of the display and moves the waveform to the left.

The roll mode has following restrictions:
- Roll mode disables persistence
- History is not available
- Event actions are not possible

The instrument activates the roll mode automatically if the following conditions are fulfilled:

- Acquisition time exceeds the defined "Minimum acquisition time"
- Waveform arithmetic is disabled ("Off")
- All channel waveforms are set to the same decimation mode, and only to one of these values: "Sample", "Peak detect", or "High res"
- All mask tests are disabled
- Fast segmentation is disabled
- Event actions are disabled
- FFT is disabled
- All serial buses are disabled
- All digital channels are disabled (MSO option R&S RTE-B1)

The roll mode depends also on sample rate and record length. In roll mode, the sample rate limit is 1 MSa/s. At 50 s, the resulting record length limit is 50 MSa.

If the acquisition time is >50 s, the record length limit is effective, and the maximum sample rate depends on the acquisition time:

Sample rate ≤ 50 MSa / Acquisition time.

If the acquisition time is <50 s, the maximum sample rate in roll mode depends on the number of active channels:

Sample rate = 1 MSa/s / Number of active wfms.

The corresponding maximum record length is:

*Record length ≤ 1 MSa/s * Acquisition time / Number of active wfms*.

Thus, the roll mode switches off, or it does not activate automatically if:

- The record length exceeds the limit at acquisition times >50 s.
- The sample rate exceeds the limit.
- Too many waveforms are active.

Remote command:
`TIMebase:ROLL:ENABle` on page 1075
`TIMebase:ROLL:STATe?` on page 1076

Minimum acquisition time ← Roll mode

The instrument can activate the roll mode automatically if the Acquisition time exceeds the value given here.

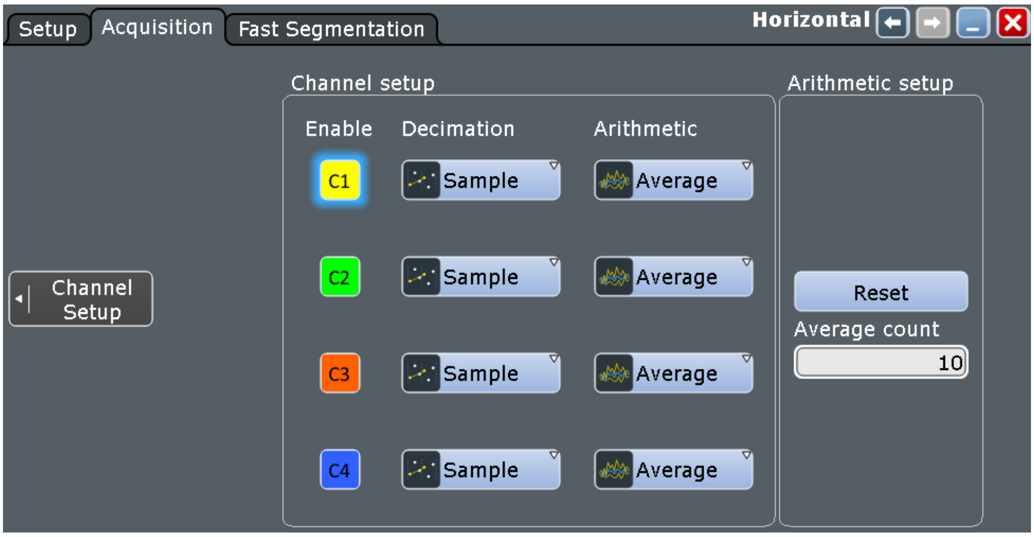
Remote command:
`TIMebase:ROLL:MTIMe` on page 1076

4.2.2 Acquisition

Access: [ACQUISITION] key

Acquisition settings control how the waveform is built from the captured samples.

For background information, see Chapter 4.1.2, "Sampling and Acquisition", on page 130.



The "Decimation" and "Arithmetic" settings are the same for all active channels.

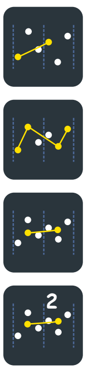
Enable 143
Decimation 143
Arithmetic 144
Average count (N-single count) 144
Reset 145

Enable

Activates or deactivates the channel.

Remote command:

`CHANnel<m>[:WAVeform<n>][:STATe]` on page 1076



Decimation

Selects the decimation mode. Decimation reduces the data stream of the ADC to a stream of waveform points with lower sample rate and a less precise time resolution. The R&S RTE uses decimation, if the waveform "Sample rate" is less than the ADC sample rate. In this case, interpolation is not possible.

There are different methods to define the recorded waveform point out of n sample points:

"Sample"	One of n samples in a sample interval of the ADC is recorded as waveform point, the other samples are discarded. The time between the two adjacent waveform points is exactly the resolution. Very short glitches might remain undiscovered by this method.

"Peak detect" The minimum and the maximum of n samples in a sample interval are recorded as waveform points, the other samples are discarded.

"High res" The average of n sample points is recorded as one waveform sample. Averaging reduces the noise, the result is a more precise waveform with higher vertical resolution.

"RMS" The waveform point is the root mean square of n sample values. Thus, the RMS value reflects the instantaneous power. This arithmetic mode is used to average a measured power waveform. Linear averaging of power signals causes an error dependent on the noise of the signal to be averaged.

Remote command:
`CHANnel<m>[:WAVeform<n>]:TYPE` on page 1077

Arithmetic
Waveform arithmetic builds the resulting waveform from several consecutive acquisitions of the signal. The arithmetic works with interpolated and decimated waveforms.

The methods are:

"Off" The data of only one acquisition is recorded according to the decimation settings. In effect, no waveform arithmetic is processed.

"Envelope" Detects the minimum and maximum values in a sample interval over several acquisitions. Each acquisition is done in the "Peak detect" decimation mode, and the most extreme values for all acquisitions build the envelope. The resulting diagram shows two envelope waveforms: the minimums (floor) and maximums (roof).

Note: If you change from "Envelope" to "Off", make sure to set also the "Decimation" to the required value.

"Average" The average is calculated from the data of the current acquisition and several acquisitions before. The method reduces random noise and other heterodyne signals. It requires a stable, triggered and periodic signal for correct function.
The number of acquisitions for average calculation is defined with "Average count"

Remote command:
`CHANnel<m>[:WAVeform<n>]:ARIThmetics` on page 1077

Average count (N-single count)
Access:
- TRIGGER > "Control" tab > "Average count (N-single count)"
- [ACQUISITION] > "Average count (N-single count)"
- [HORIZONTAL] > "Fast Segmentation" tab > disable "Acquire maximum" > "Required"
- [MATH] > "Setup" tab > "Mode" is not "Off" > "Average count"

The acquisition and average count has several effects:
- It sets the number of waveforms acquired with [RUN N× SINGLE]
- It defines the number of waveforms used to calculate the average waveform.

Thus, the instrument acquires sufficient waveforms to calculate the correct average if "Average" is enabled for waveform arithmetic. The higher the value is, the better the noise is reduced.

- It sets the number of acquisitions to be acquired in a fast segmentation acquisition series. Thus, you can acquire exactly one fast segmentation acquisition series with [RUN N× SINGLE].
  If fast segmentation is enabled and configured to acquire the maximum number of acquisitions, the acquisition count is set to that maximum number and cannot be changed. See also: "Number of acquisitions" on page 146.
- It is the "Finished" criteria for the state of a mask test.

Remote command:
`ACQuire:COUNt` on page 1078

Reset
Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1078

4.2.3 Fast Segmentation

In normal acquisition mode, only a short time is used for sampling; processing and display take most of the time. The processing and display time is blind time causing a gap in the recorded signal. The normal acquisition mode may miss very short-time and infrequent events occurring during the dead time.

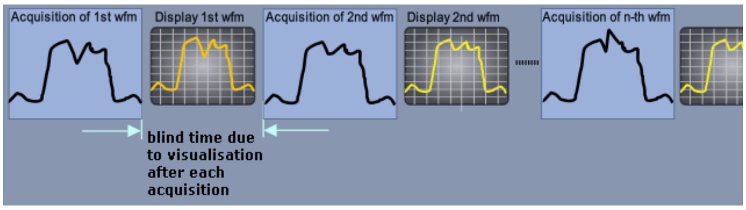


Figure 4-5: Normal acquisition with blind time

With fast segmentation, several triggered acquisitions are captured fast, with hardly any dead time between the acquisitions. The data is processed and the waveforms are displayed when the acquisition of the series has been completed.

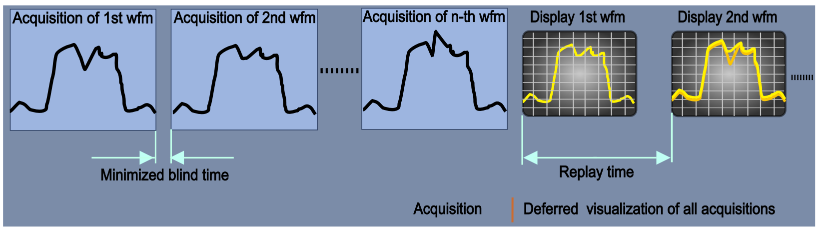


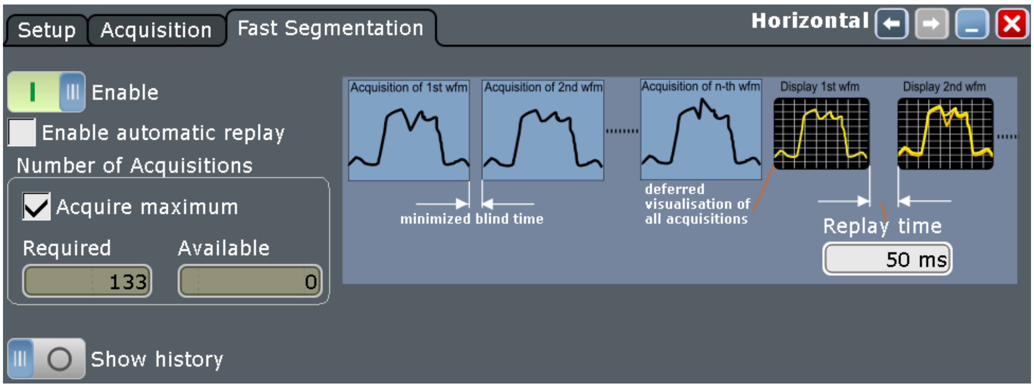
Figure 4-6: Fast segmentation with deferred processing and display

Fast segmentation and history

The acquisition series is written in the sample memory, thus the memory size limits the number of acquisitions in a series. This memory is the memory that is accessed by the history, thus the history function is used to read out the contents of the sample memory.

To use the history functionality, enable "Show history" in the "Fast Segmentation" tab. The history viewer settings are displayed directly in the "Fast Segmentation" tab.

See also: Chapter 6.4, "History", on page 273.



Enable fast segmentation

Switches the fast segmentation mode on and off.

Remote command:
`ACQuire:SEGMented:STATe` on page 1078

Enable automatic replay

If enabled, the instrument starts processing and displaying the data when the acquisition series is captured completely. Depending on the number of acquisitions, it can take some time until the acquisition series is displayed. If the setting is disabled, the instrument only captures the data and stores it in the sample memory.

Remote command:
`ACQuire:SEGMented:AUToreplay` on page 1079

Number of acquisitions

You can define the number of acquisitions to be stored in a fast segmentation acquisition series:

- Acquire the maximum number of acquisitions that can be stored in the sample memory.
  To acquire the maximum number, enable "Acquire maximum". The maximum number of acquisitions is shown in the "Required" field.
- Acquire a given number of acquisitions.
  Enter the number in the "Required" field.

The acquisition count (Average count (N-single count)) is always set to the required number of acquisitions. Thus you can acquire exactly one fast segmentation acquisition series with [RUN N× SINGLE]. The [RUN CONT] key works in the same way as [RUN N× SINGLE], it stops acquisition when the series is completed.

You can stop the running acquisition before the series is completed.

The number of acquired waveforms is shown in "Available" and can be displayed with "Show history".

Remote command:
`ACQuire:SEGMented:MAX` on page 1079

Replay time
Defines the display speed of the fast segmentation acquisition series. Display starts after the series has been captured completely.

See also: "Replay time per acq." on page 276

Show history
Enables the history mode and displays the history viewing functions in the "Fast Segmentation" tab.

See also: Chapter 6.4.2.1, "Viewer", on page 275.

4.3 Vertical Settings

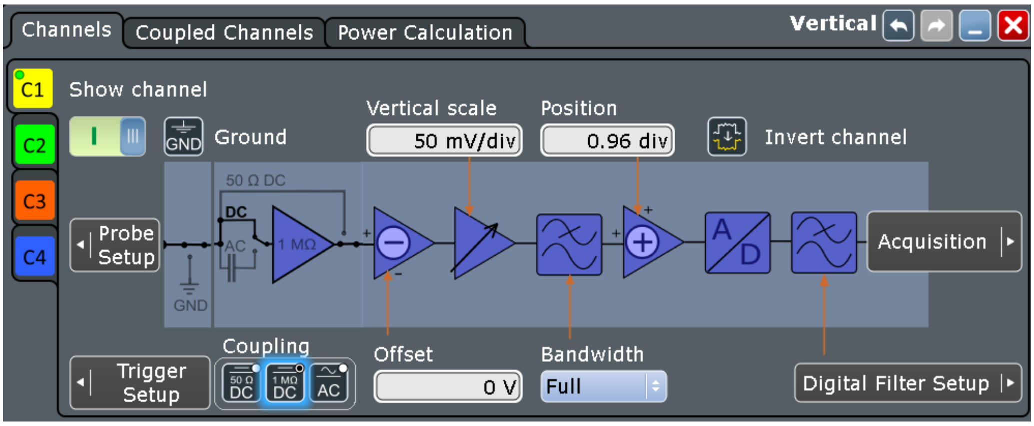
The "Vertical" menu contains all channel-dependent settings and information.

● Channels....................147
● Coupled Channels....................150
● Power Calculation....................150

4.3.1 Channels

Access: "Vertical" menu > "Channels"

The "Channels" tab provides all basic vertical settings. The channels are listed in vertical tabs at the left side of the dialog box.



Make sure that the correct channel tab is selected. The vertical rotary knobs are illuminated in the color of the selected channel.

Enable / Show channel 148
Ground 148
Vertical scale 148
Position 148
Invert channel 148
Coupling 149
Offset 149
Bandwith 149

Enable / Show channel

Switches the channel signal on or off. The signal icon appears on the signal bar. The waveform of the last acquisition is displayed in the diagram.

Remote command:
`CHANnel<m>:STATe` on page 1079

Ground

Connects the input to the ground.

Remote command:
`CHANnel<m>:GND` on page 1080

Vertical scale

Defines the vertical scale in Volts per division. Increasing the scale compresses the display of the signal.

Remote command:
`CHANnel<m>:SCALe` on page 1080

Position

Moves the selected signal up or down in the diagram. The visual effect is the same as for Offset but the waveform is adjusted later in the signal flow. While the offset sets a voltage, position is a graphical setting given in divisions.

By default, the horizontal grid axis remains in the center when the offset is changed. To shift the axis together with the waveform, disable Keep Y-grid fixed in "Display > Diagram Layout".

Remote command:
`CHANnel<m>:POSition` on page 1081

Invert channel

Turns the inversion of the signal amplitude on or off. To invert means to reflect the voltage values of all signal components against the ground level. If the inverted channel is the trigger source, the instrument triggers on the inverted signal.

You can use inversion, for example, to switch the polarity of a differential signal without changing the probe connections.

Remote command:
`CHANnel<m>:INVert` on page 1082

Coupling
Selects the connection of the channel signal. The coupling determines what part of the signal is used for waveform analysis and triggering.

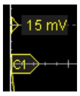
In addition to coupling, the signal can be filtered for high frequency rejection, see .Chapter 4.8, "Digital Filter Setup", on page 192

"DC 50 Ω"	Connection with 50 Ω termination, passes both DC and AC components of the signal.
"DC 1 MΩ"	Connection with 1 MΩ termination, passes both DC and AC components of the signal.
"AC"	Connection with 1 MΩ termination through DC capacitor, removes DC and very low-frequency components. If AC coupling is set, the attenuation of passive probes has no effect, and voltage is applied to the instrument with factor 1:1. Observe the voltage limits, otherwise you can damage the instrument.

Remote command:
`CHANnel<m>:COUPling` on page 1080

Offset
The offset voltage is subtracted to correct a signal with DC offset. The vertical center of the selected channel is shifted by the offset value and the signal is repositioned within the diagram area. Negative offset values move up the waveform, positive values move it down.

The offset of a signal is determined and set by the autoset procedure. The current value is shown in the waveform label, and it is marked by a small triangle in the grid.



If a Rohde & Schwarz differential probe is connected, the offset is the differential offset.

If a Rohde & Schwarz modular probe is connected, the offset of the selected probe mode is used. For example, in CM mode, the offset is the common mode offset.

By default, the horizontal grid axis remains in the center when the offset is changed. To shift the axis together with the waveform, disable Keep Y-grid fixed in "Display > Diagram Layout".

Remote command:
`CHANnel<m>:OFFSet` on page 1082

Bandwith
Selects the bandwidth limit.

The specified bandwidth indicates the range of frequencies that the instrument can acquire and display accurately with less than 3dB attenuation. The probe has also a limited bandwidth and thus affects the resulting system bandwidth.

See also: Chapter 4.1.1.3, "Bandwidth", on page 130

"Full"	At full bandwidth, all frequencies in the specified range are acquired and displayed. Full bandwidth is used for most applications.

"20 MHz, 200 MHz"
Frequencies above the selected limit are removed to reduce noise at different levels.

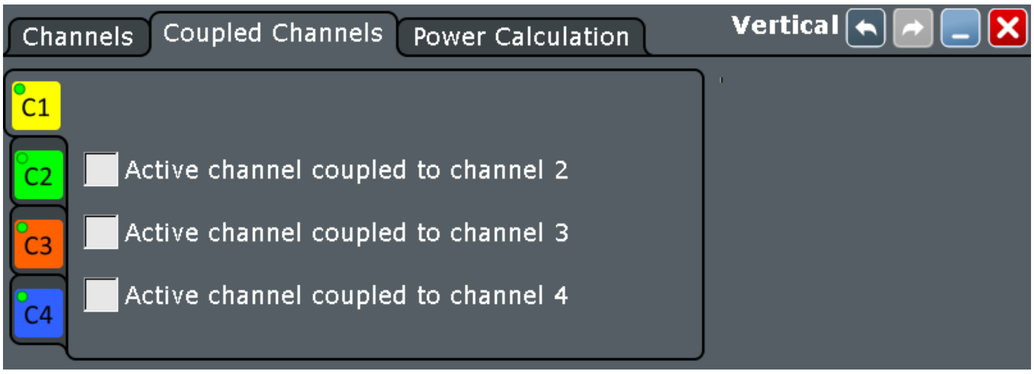
Remote command:
`CHANnel<m>:BANDwidth` on page 1083

4.3.2 Coupled Channels

Access: "Vertical" menu > "Coupled Channels"

Channel coupling sets the vertical settings of the coupled channels to the values of the active channel. If you want to have the same vertical settings for two or more channels, you can set them at once by coupling these channels.

Channel coupling affects all vertical settings that are adjusted in the "Channels" tab: vertical scale, position, offset, bandwidth, coupling, and ground.

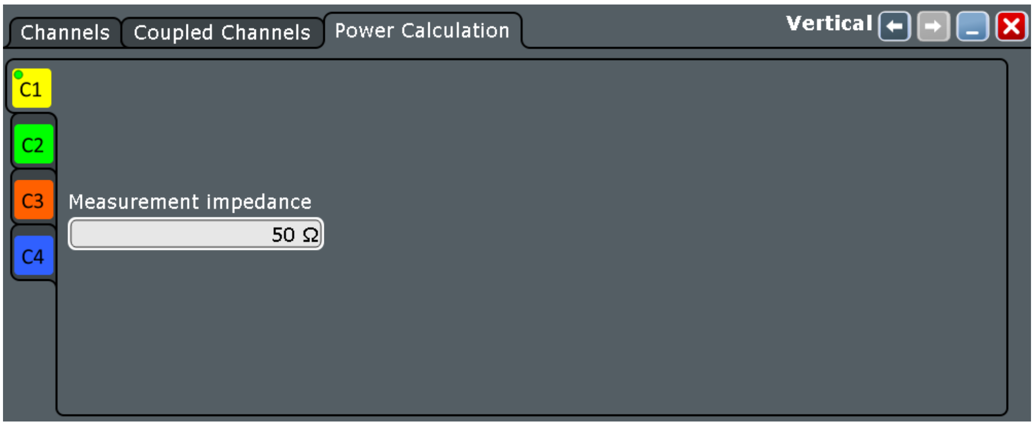


Remote command:
- `CHANnel<m>:CPLing` on page 1083

4.3.3 Power Calculation

Access: "Vertical" menu > "Power Calculation"

Make sure that the correct channel tab is selected.



Measurement impedance

Sets the impedance of the channel for power calculations and measurements.

Remote command:
CHANnel<m>:IMPedance on page 1083

4.4 High Definition Mode

The high definition mode offers up to 16 bits of vertical resolution. Higher vertical resolution reduces quantization noise and acquires waveforms of higher accuracy with finer details of the signal to be seen.

The number of vertical resolution bits defines the number of vertical levels that the acquisition samples are mapped to (quantization). 16 bits of resolution represent 65536 voltage quantization levels, while 8 bits of resolution represent only 256 voltage levels. The waveform values are recorded with 16 bit word length, except for peak detect decimation.

The higher vertical resolution is achieved by applying a digital low pass filter (DSP filter) to the output of the ADC, which reduces the bandwidth of the signal. Increasing the bandwidth reduces the resulting digital resolution. The high definition is also applied to the digital trigger, thus the R&S RTE can trigger with the same high resolution with which they can display signals.

High definition can be used, for example, to measure slow pulses with high accuracy, or to analyze AM signals with very low modilation index, as used in radar.

See also:

- Chapter 4.1.1, "Vertical System", on page 128
- Chapter 4.1.2, "Sampling and Acquisition", on page 130

4.4.1 High Definition Settings

Access: [MODE] > "Acquisition"

High definition is a special acquisition mode of the oscilloscope. This mode has only one setting - the filter bandwidth.

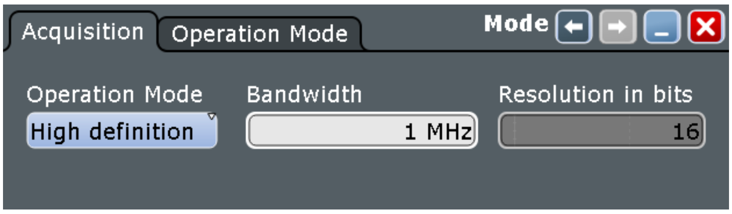


Figure 4-7: Setting the instrument into high definition mode

Operation mode

Sets the operation mode of the instrument.

"Normal"	Usual oscilloscope mode
"High definition"	Mode with higher digital resolution, up to 16 bit.

Remote command:
`HDEFinition:STATe` on page 1125

Bandwidth

Sets the filter bandwidth for the high definition mode.

The maximum filter bandwidth depends on the instrument bandwidth.

Instrument bandwidth	Maximum filter bandwidth
200 MHz	200 MHz
350 MHz	300 MHz
≥ 500 MHz	500 MHz

Remote command:
`HDEFinition:BWIDth` on page 1125

Resolution in bits

Shows the resulting vertical resolution in high definition mode. The higher the filter bandwidth, the lower the resolution. For details, refer to the R&S RTE Specifications.

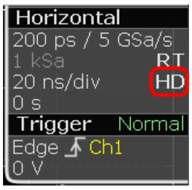
Remote command:
`HDEFinition:RESolution?` on page 1126

4.4.2 Effects of the High Definition Mode

The high definition mode has several effects:

Acquisition

The active high definition mode is indicated by "HD" in the horizontal label.



The high definition mode works with half the realtime sample rate. For FFT, the instrument halves this sample rate again.

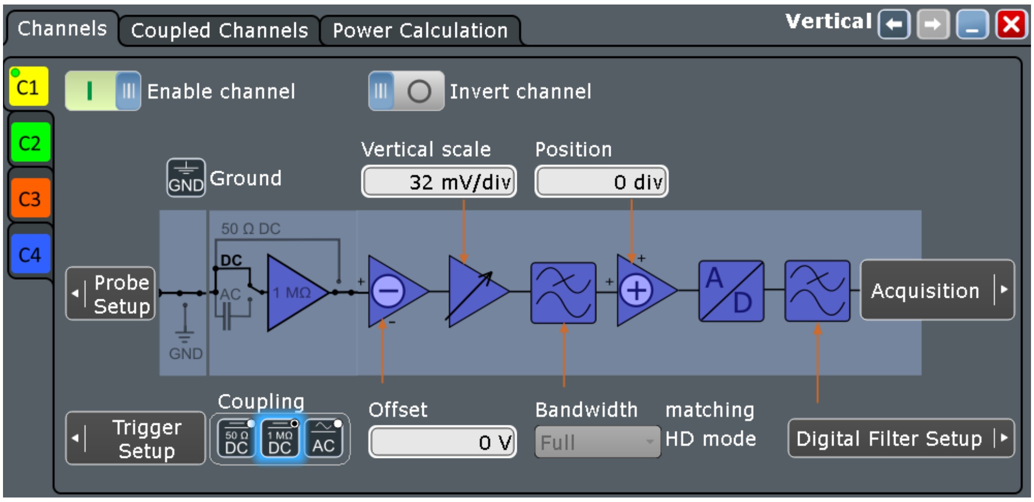
The waveform values are recorded with 16-bit word length, except for peak detect decimation (2 values with 8 bit).

Vertical system

The current bandwidth is shown in the channel label.

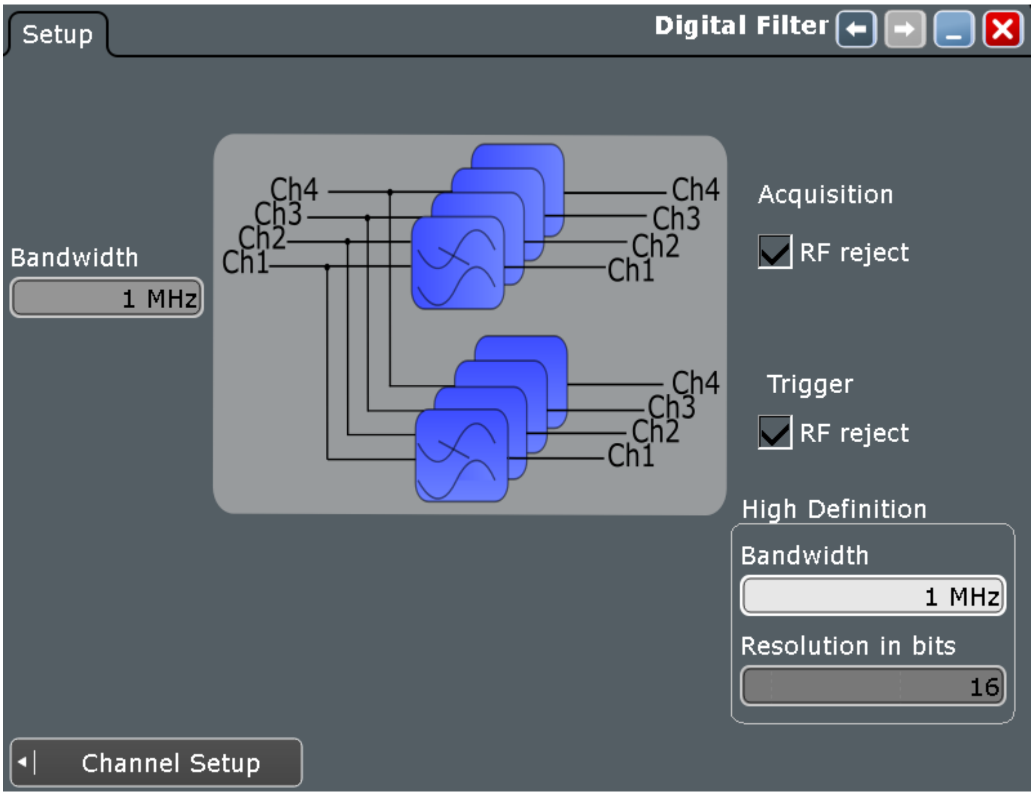
In the "Channels" dialog box (CH<x>), the "Bandwidth" setting is not available because the bandwidth is set by the high definition filter.

For R&S RTE of the 1317.2500.Kxx series only: The minimum vertical scale is 500 µV/div instead of 1 mV in normal mode.



Digital filter

The digital filter settings are set automatically. You can change the high definition "Bandwidth" in the "Digital Filter Setup".



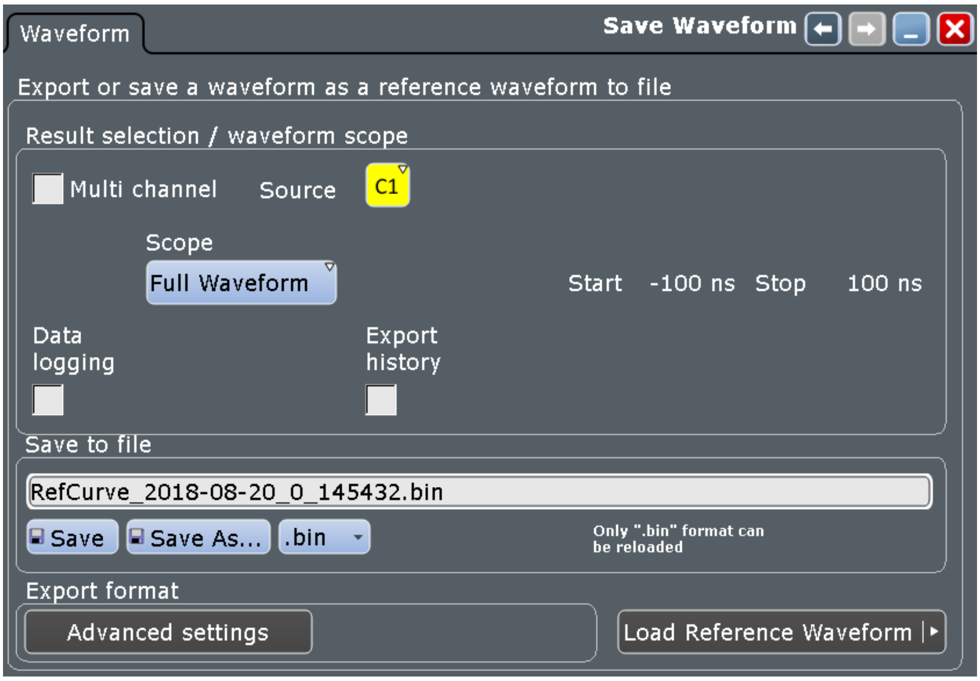
History

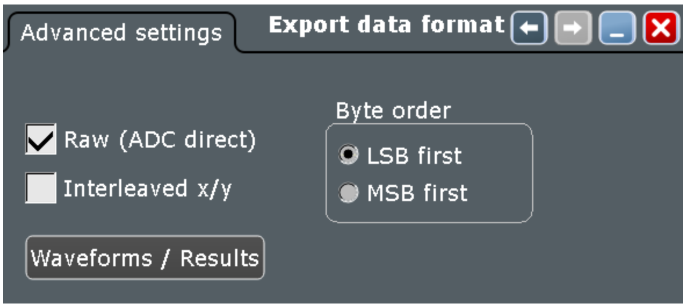
Due to the 16-bit word length, the history depth is reduced, less waveforms are saved than in normal mode.

Export

In high definition mode, waveform data in raw format is exported to file with 16-bit word length, except for peak detect decimation (2 values with 8 bit). In addition, you can define the byte order of the data words.

To define additional export settings, tap "Advanced Settings" in [FILE] > "Waveform".





See:

- "Raw (ADC direct)" on page 449
- "Interleaved x/y" on page 449
- "Byte order" on page 450

If you use remote control commands to transfer data to a controlling computer, set the data format to `INT,16` to transfer the complete data words (see `FORMat[:DATA]` on page 1045).

4.5 Probes

With R&S RTE digital oscilloscopes, you can use various probe types. Mostly these probes are passive and active voltage probes. The "Probes" dialog box provides all probe-relevant information.

The instrument can detect many probes and read out the probe-specific parameters, for example, bandwidth and attenuation.

In the "Setup" tab, you find all settings that are relevant for the connected probe.

Access: "Vertical" menu > "Probe Setup"

The functionality on the "Setup" tab changes according to the type of the attached probe. Probes with Rohde & Schwarz probe interface (probe box), and also many other passive voltage probes, are recognized by the instrument. The R&S RTE reads out the main characteristics of the probe and displays them. Other probes cannot be detected, but their characteristics are known to the instrument. These known probes are called "Predefined probes". Probes that are not recognized automatically and not predefined are unknown probes, they require manual setting of measurement unit and attenuation.

Before you adjust the settings, select the correct channel tab on the left.

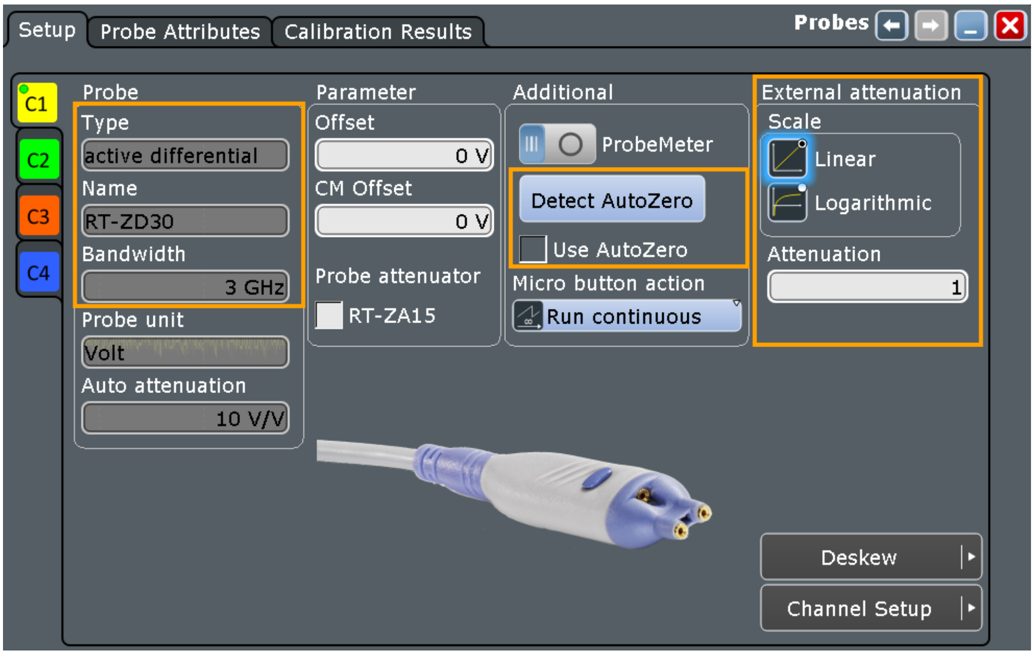
Additional information is given in the "Probe Attributes" and "Calibration Results" tabs. For background information, see Chapter 4.1.4, "Probes", on page 134.

This chapter has the following sections:

- Shared Probe Settings....................156
- Settings for the R&S Probe Interface (Voltage Probes)....................158
- Setup for Passive Probes....................162
- Setup for Active Voltage Probes....................163
- Modular Probes....................168
- Setup for Predefined Probes....................171
- Setup for Current Probes....................173
- Setup for Unknown Probes....................176
- Probe Attributes....................177
- Calibration Results....................178
- Probe Adapter R&S RT-Z2T....................178

4.5.1 Shared Probe Settings

Some of the settings in the "Pobes Setup" tab are available for all probes.



The shared probe settings are:

Type, Name, Bandwidth....................157
Detect AutoZero, Use AutoZero....................157
External attenuation: Scale, Attenuation....................158

Type, Name, Bandwidth

The fields show the characteristics of a recognized or predefined probe for information. If the instrument cannot recognize the probe, and the probe is not known, the "Type" is "None", and the other fields are empty.

Remote command:
`PROBe<m>:SETup:TYPE?` on page 1087
`PROBe<m>:SETup:NAME?` on page 1087
`PROBe<m>:SETup:BANDwidth?` on page 1087
`TRPRobe:SETup:TYPE?` on page 1087 (external trigger input)
`TRPRobe:SETup:NAME?` on page 1087 (external trigger input)
`TRPRobe:SETup:BANDwidth?` on page 1087 (external trigger input)

Detect AutoZero, Use AutoZero

Differences in DUT and oscilloscope ground levels can cause larger zero errors, which affect the waveform. If the DUT is ground-referenced, the AutoZero function corrects the zero error of the probe to optimize measurement results at small signal levels. The validation limit depends on the probe attenuation because probes with high attenuation often have to compensate high offsets. AutoZero detects offset values even when the signal is out of the current measurement range.

To correct the zero error of voltage probes, short the signal pin and the ground pin together and connect them to the ground of the DUT. Then tap "Detect AutoZero". While the alignment is running, the instrument switches to DC coupling to display the waveform correctly.

To include the measured offset in measurement results, enable "Use AutoZero".

If a current probe is connected, the function demagnetizes the probe's sensor head and sets the waveform to zero position. See "Detect AutoZero" on page 175.

Remote command:
`PROBe<m>:SETup:OFFSet:AZERo` on page 1088
`PROBe<m>:SETup:OFFSet:USEautozero` on page 1088

External attenuation: Scale, Attenuation
Consider a voltage divider that is part of the DUT before the measuring point. The external attenuation is included in the measurement, and the instrument shows the results that would be measured before the divider. External attenuation can be used with all probes.

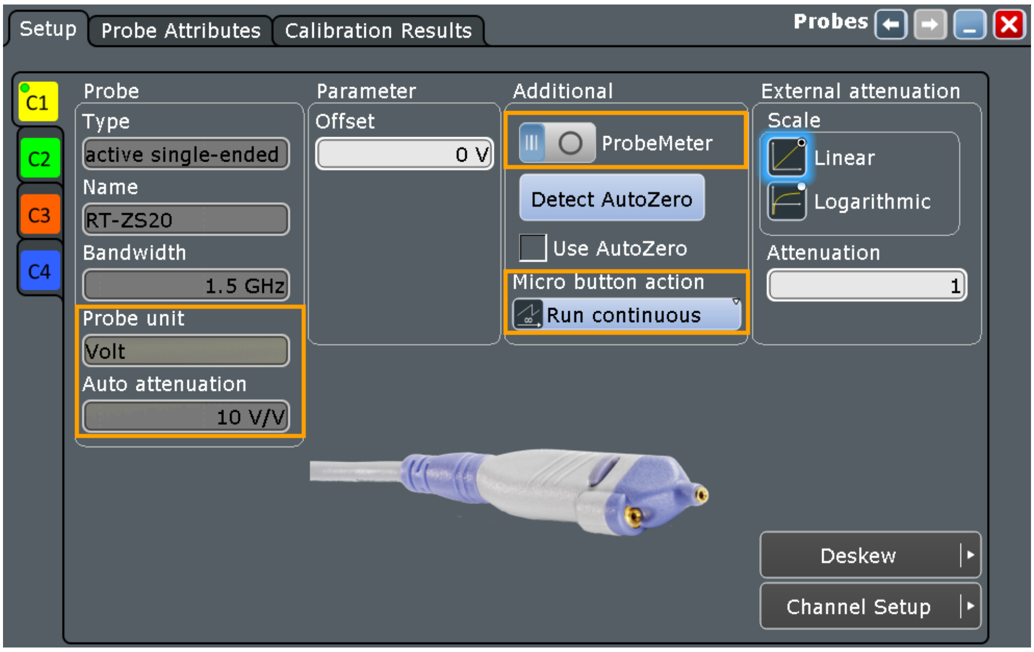
"Scale" Select linear or logarithmic attenuation scale.

"Attenuation" Enter the attenuation of the voltage divider according to the selected scale. The conversion from linear to logarithmic values depends on the "Vertical unit" of the probe:
For voltage-based unit (V and A):
$attenuation\ (dB) = 20 * \log_{10}(attenuation\ factor)$
For power-based unit (W):
$attenuation\ (dB) = 10 * \log_{10}(attenuation\ factor)$

Remote command:
`CHANnel<m>:EATScale` on page 1089
`CHANnel<m>:EATTenuation` on page 1089

4.5.2 Settings for the R&S Probe Interface (Voltage Probes)

Active voltage probes with Rohde & Schwarz probe interface provide special features: the micro button and the ProbeMeter. Furthermore, the R&S RTE can read out the attenuation of the probe.



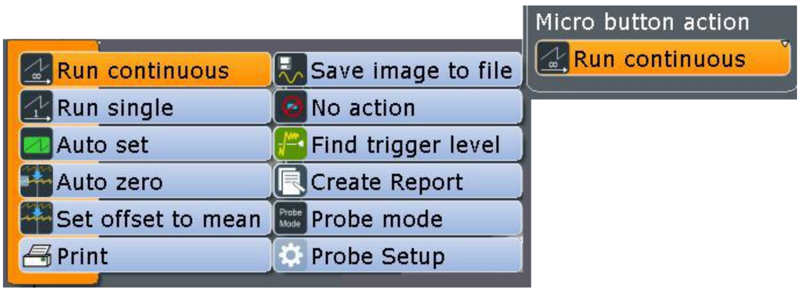
The settings for active voltage probes with Rohde & Schwarz probe interface are:

Probe unit, Auto attenuation..159
Micro button action...159
ProbeMeter... 160

Probe unit, Auto attenuation
If the probe is recognized by the R&S RTE, the instrument reads the attenuation unit and value from the probe an displays them.

Remote command:
`PROBe<m>:SETup:ATTenuation[:AUTO]?` on page 1088
`TRPRobe:SETup:ATTenuation[:AUTO]?` on page 1088 (trigger input)

Micro button action
Active voltage probes with Rohde & Schwarz probe interface have a configurable micro button on the probe head. Pressing this button, you start an action on the instrument directly from the probe. The button is disabled during internal automatic processes, for example, during self-alignment, autoset, and find level.



Select the action that you want to start from the probe:

"Run Continuous"
Is the default assignment. Starts or stops the acquisition (same as RUN CONT key).

"Run single" Starts a defined number of acquisitions (same as [RUN N× SINGLE] key).

"Auto set" Starts the autoset procedure (same as AUTOSET key).

"AutoZero" Starts an auto zero measurement, see "Detect AutoZero, Use AutoZero" on page 157.

"Set offset to mean"
Performs an automatic compensation for a DC component of the input signal using the result of a background mean measurement.
See: "Offset to mean" on page 173.

"Print" Prints the current display according to the "Printer control" settings in the "Print" dialog box. Depending on the selected printer, you can print to a local or network driver, or save to a file. See also: Chapter 11.4, "Screenshots", on page 461.

"Save image to file"
Saves the current display as image according to the image settings in the "Print" dialog box. See also Chapter 11.4, "Screenshots", on page 461.

"No action" Select this option to prevent unwanted actions due to unintended usage of the micro button.

"Find trigger level" Sets the trigger level automatically to *0.5 * (MaxPeak – MinPeak)*.

"Create report" Creates and saves a report using the settings defined in "File" menu > "Report Setup".

"Probe mode" Only available if a R&S RT-ZM modular probe is connected. Sets the measurement mode of the modular probe. See also "Probe Mode" on page 169.

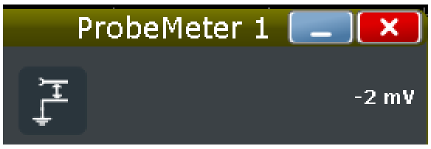
"Probe Setup" Opens the "Probes Setup" dialog box.

Remote command:
`PROBe<m>:SETup:MODE` on page 1090

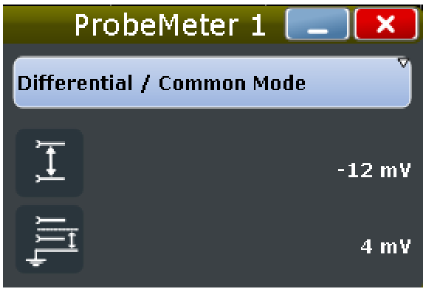
ProbeMeter

The integrated R&S ProbeMeter of active voltage probes with Rohde & Schwarz probe interface is a voltmeter. It measures DC voltages between the probe tip and ground connection or between the probe tips with very high precision. The R&S ProbeMeter enables ground-referenced measurements of voltages. The measurement is performed continuously and in parallel to the measurements of the oscilloscope.

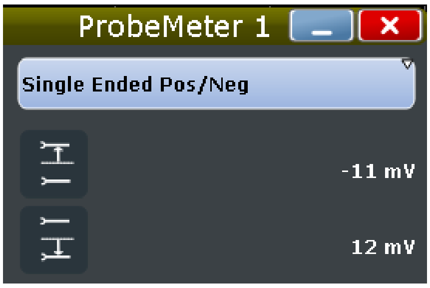
- **"Probemeter"**
  Select "Probemeter" to activate the integrated R&S ProbeMeter of active R&S probes. The measured voltages are displayed in the "ProbeMeter" result box on the screen.
- **ProbeMeter measurement results of single-ended active R&S probes**
  Measures the voltage between the probe tip and the ground.



- **ProbeMeter measurement results of differential and modular R&S probes**
  You can select the voltage to be measured by the differential active probe:
  – "Differential / Common Mode":
    Differential voltage is the voltage between the positive and negative signal sockets.
    Common mode voltage is the mean voltage between the signal sockets and the ground socket. It measures the voltage level relative to ground, for example, to check the operating voltage window.



  – "Single Ended Pos/Neg": Measures the voltage between the positive/negative signal socket and the ground.



The ProbeMeter always measures the common mode and differential voltages. Single-ended voltages are calculated values:
$V_p = V_{cm} + 0.5 * V_{in}$ and $V_n = V_{cm} - 0.5 * V_{in}$

Remote command:
Channel probes:
PROBe<m>:PMETer:VISibility on page 1091
PROBe<m>:SETup:DISPlaydiff on page 1091
PROBe<m>:PMETer:RESults:SINGle? on page 1092
PROBe<m>:PMETer:RESults:POSitive? on page 1093
PROBe<m>:PMETer:RESults:NEGative? on page 1093
PROBe<m>:PMETer:RESults:DIFFerential? on page 1092
PROBe<m>:PMETer:RESults:COMMon? on page 1092
Probe on trigger input:
TRPRobe:SETup:DISPlaydiff on page 1091
TRPRobe:PMETer:VISibility on page 1091
TRPRobe:PMETer:RESults:SINGle? on page 1092
TRPRobe:PMETer:RESults:POSitive? on page 1093

TRPRobe:PMETer:RESults:NEGative? on page 1093
TRPRobe:PMETer:RESults:DIFFerential? on page 1092
TRPRobe:PMETer:RESults:COMMon? on page 1092

4.5.3 Setup for Passive Probes

Passive probes are the most widely used probes for oscilloscope measurements. Passive probes require compensation.

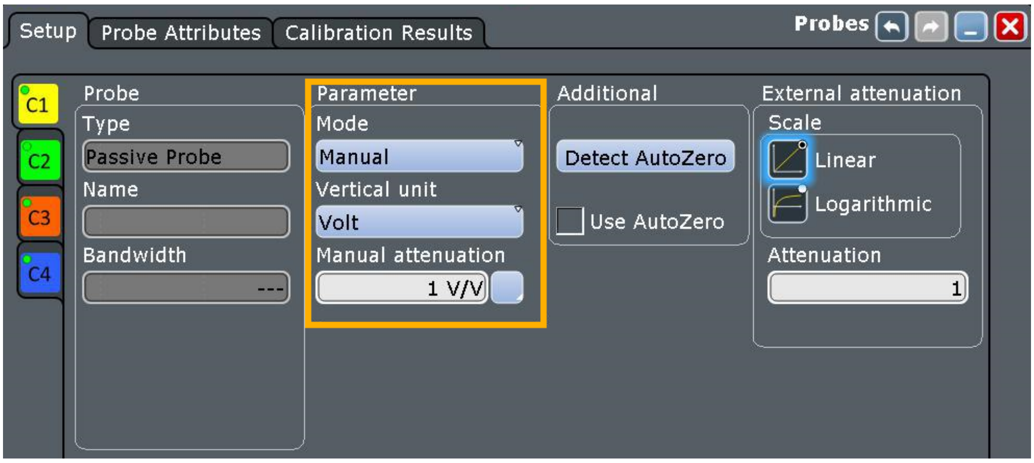


Figure 4-8: Probe setup for passive probe R&S RT-ZP10

The following shared probe settings are available:
- "Type, Name, Bandwidth" on page 157
- "Detect AutoZero, Use AutoZero" on page 157
- "External attenuation: Scale, Attenuation" on page 158

If a passive probe is connected, the probe attenuation is read out and shown in the "Setup" tab:
- "Probe unit, Auto attenuation" on page 159

If you need to change the unit or attenuation, change the "Mode" to "Manual" and enter the correct values.

Mode.. 162
Vertical unit, Attenuation, Gain...163

Mode

Defines how the attenuation of a passive probe is set.

"Auto" The instrument uses the values that are read out from the probe.

"Manual" You can define the attenuation unit and value.
See: "Vertical unit, Attenuation, Gain" on page 163

Remote command:
PROBe<m>:SETup:ATTenuation:MODE on page 1094
TRPRobe:SETup:ATTenuation:MODE on page 1094 (trigger input)

Vertical unit, Attenuation, Gain

If a predefined probe is connected and selected, the attenuation or gain values are shown.

For unknown probes and passive probes in manual mode, you can set user-defined values for unit, gain and attenuation.

If AC coupling is set, the attenuation of passive probes has no effect, and voltage is applied to the instrument with factor 1:1. Observe the voltage limits, otherwise you can damage the instrument.

Remote command:

`PROBe<m>:SETup:ATTenuation:UNIT` on page 1094
`PROBe<m>:SETup:ATTenuation:MANual` on page 1094
`PROBe<m>:SETup:GAIN:MANual` on page 1095
`TRPRobe:SETup:ATTenuation:UNIT` on page 1094 (external trigger input)
`TRPRobe:SETup:ATTenuation:MANual` on page 1094 (external trigger input)
`TRPRobe:SETup:GAIN:MANual` on page 1095 (external trigger input)

4.5.4 Setup for Active Voltage Probes

Active voltage probes with Rohde & Schwarz probe interface have an integrated data memory that contains identification data and individual probe correction parameters. The R&S RTE can detect these probes and read out the data. Furthermore, these probes have a micro button and a ProbeMeter.

Active voltage probes that are offered by Rohde & Schwarz but not equipped with a Rohde & Schwarz probe interface are known to the R&S RTE as predefined probes, see Chapter 4.5.6, "Setup for Predefined Probes", on page 171.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 157
- "Detect AutoZero, Use AutoZero" on page 157
- "External attenuation: Scale, Attenuation" on page 158

Special features of the Rohde & Schwarz probe interface are described in these sections:

- "Probe unit, Auto attenuation" on page 159
- "Micro button action" on page 159
- "ProbeMeter" on page 160

The specific settings of Rohde & Schwarz active probes are described in the following chapters:

- R&S RT-ZS Single-Ended Probes....................164
- R&S RT-ZD Differential Probes....................165
- R&S RT-ZPR Power Rail Probes....................166
- R&S RT-ZHD High-Voltage Differential Probes....................167

4.5.4.1 R&S RT-ZS Single-Ended Probes

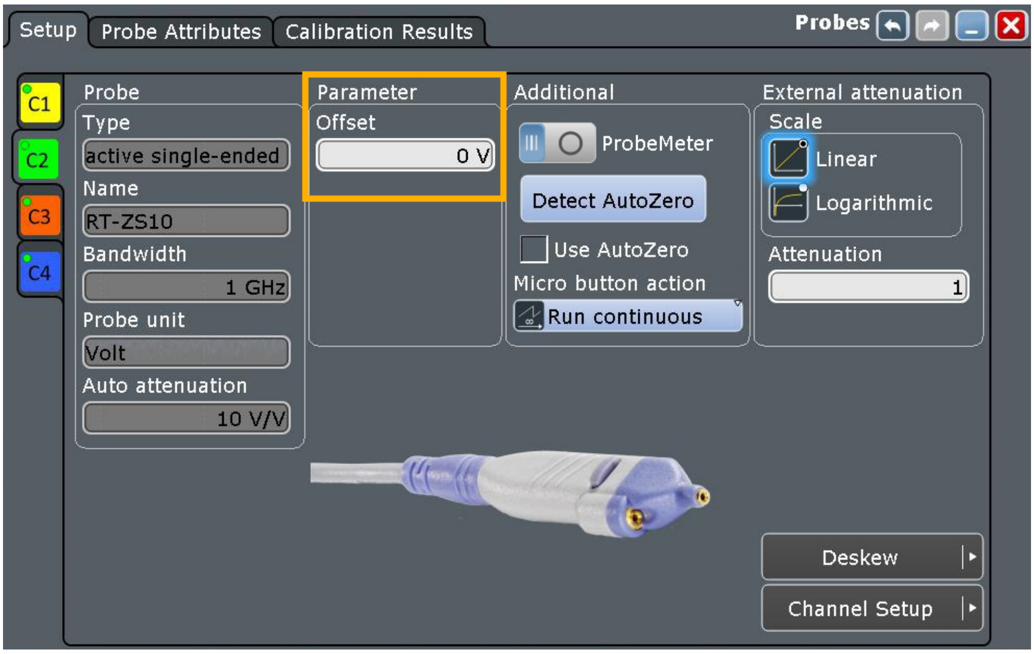


Figure 4-9: Probe setup for active single-ended probe R&S RT-ZS10

The only setting for R&S RT-ZS probes is the channel offset. See "Offset" on page 149.

4.5.4.2 R&S RT-ZD Differential Probes

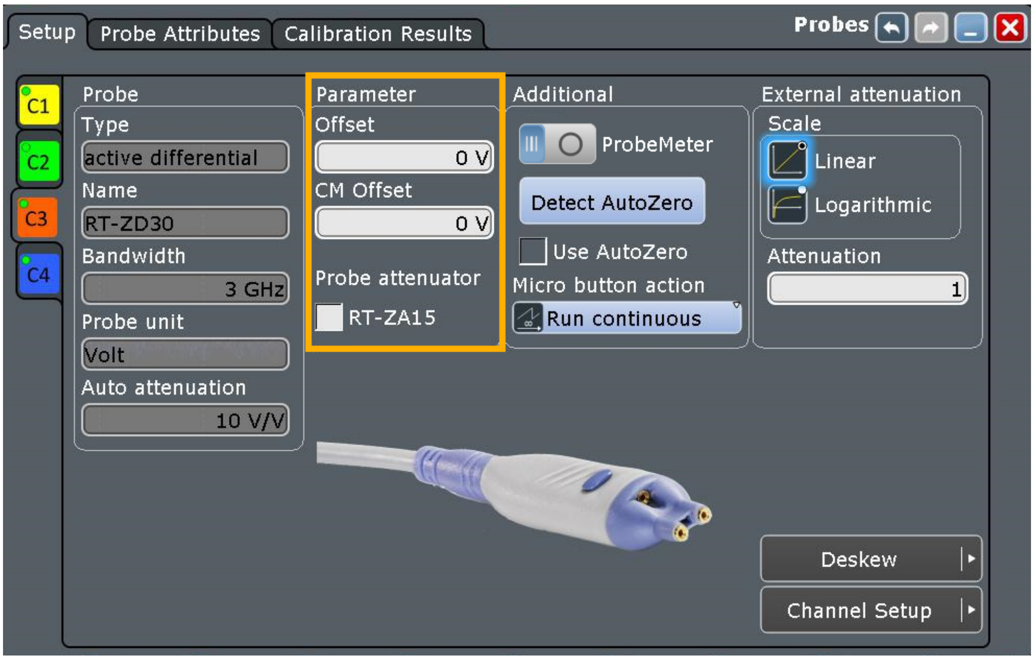


Figure 4-10: Probe setup for active differential probe R&S RT-ZD30

The offset is the differential offset. See "Offset" on page 149.

Specific settings for R&S RT-ZD probes are the following:

CM offset
Sets the common-mode offset to compensate for a common DC voltage applied to both input sockets (referenced to the ground socket). Offset compensation is particularly helpful for measurements on differential signals with high common mode levels, for example, current measurements using a shunt resistor. You can measure the common mode input voltage using the R&S ProbeMeter.

The setting is available for Rohde & Schwarz differential probes, and for modular probes in DM or CM mode (see "DM Offset, CM Offset, P Offset, N Offset" on page 170).

Remote command:
`PROBe<m>:SETup:CMOFfset` on page 1095
`TRPRobe:SETup:CMOFfset` on page 1095 (external trigger input)

Probe attenuator RT-ZA15
If you use the external attenuator R&S RT-ZA15 together with one of the differential active probes R&S RT-ZD, enable RT-ZA15 to include the external attenuation in the measurements.

Remote command:
`PROBe<m>:SETup:ZAXV` on page 1096
`TRPRobe:SETup:ZAXV` on page 1096

4.5.4.3 R&S RT-ZPR Power Rail Probes

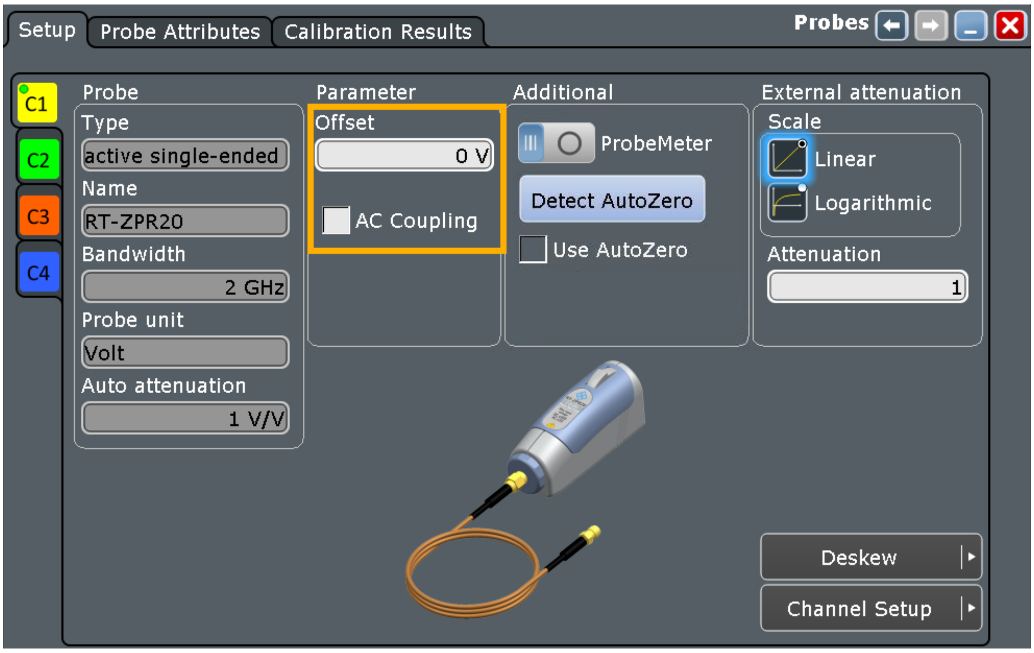


Figure 4-11: Probe setup for power rail probe R&S RT-ZPR

The offset is the channel offset. See "Offset" on page 149.

If the ProbeMeter is active, an additional function is provided in the "ProbeMeter" result box:

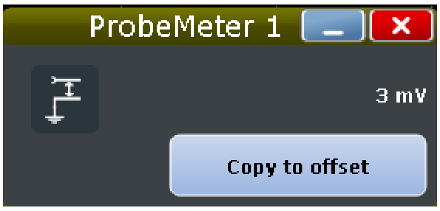


Figure 4-12: ProbeMeter result box for

Specific settings for R&S RT-ZPR probes are the following:

AC Coupling
Enables AC coupling in the R&S RT-ZPR power rail probes, which removes DC and very low-frequency components. The R&S RT-ZPR probe requires 50 Ω input termination, for which the channel AC coupling is not available. The probe setting allows AC coupling also at 50 Ω inputs.

Remote command:
`PROBe<m>:SETup:ACCoupling` on page 1096
`TRPRobe:SETup:ACCoupling` on page 1096

Copy to offset
Sets the measured ProbeMeter value as offset. Thus, the value is considered in measurements.

If the probe is connected to the external trigger input, the function is not available.

Remote command:
`PROBe<m>:SETup:ADVanced:PMToffset` on page 1096

4.5.4.4 R&S RT-ZHD High-Voltage Differential Probes

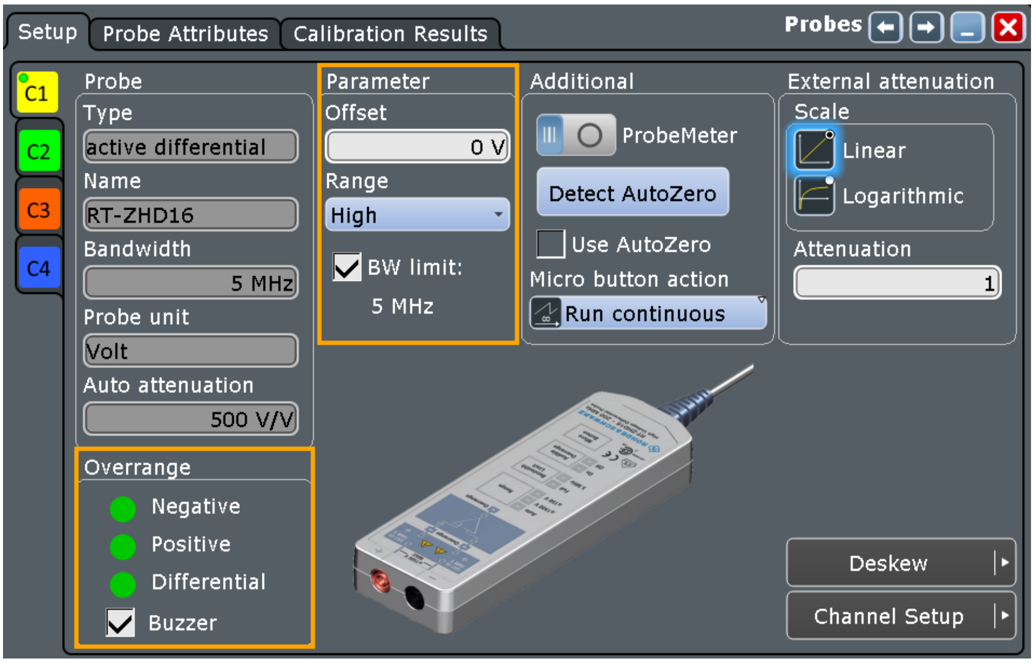


Figure 4-13: Probe setup for R&S RT-ZHD probes

The offset is the differential offset. See "Offset" on page 149.

Specific settings for R&S RT-ZHD probes are the following:

Range
Sets the voltage range of a R&S RT-ZHD probe. You can set the range on the probe control box or at the oscilloscope.

"Auto" The voltage range is set only at the oscilloscope with "Vertical scale".

"Low" Sets the lower voltage range of the connected probe. The selected value is shown in "Auto Attenuation".

"High" Sets the higher voltage range of the connected probe. The selected value is shown in "Auto Attenuation".

Remote command:
`PROBe<m>:SETup:ADVanced:RANGe` on page 1097

BW limit
Activates the lowpass filter in the probe control box and displays the used limit. You can also set the filter directly on the probe control box.

Remote command:
`PROBe<m>:SETup:ADVanced:FILTer` on page 1097
`TRPRobe:SETup:ADVanced:FILTer` on page 1097

Buzzer
Activates the acoustic overrange warning in the probe control box. You can also activate the sound directly on the probe control box.

Remote command:
`PROBe<m>:SETup:ADVanced:AUDioverload` on page 1097
`TRPRobe:SETup:ADVanced:AUDioverload` on page 1097

Negative, Positive, Differential Overrange
The color turns red if the voltage exceeds the probe range. The indicators are also available on the probe control box.

4.5.5 Modular Probes

The probes of the R&S RT-ZM family are modular probes. They have a probe head and a probe amplifier connected by a cable, and various probe tip modules and tip cables for different applications. R&S RT-ZM probes are equipped with Rohde & Schwarz probe interface, and provide special features: ProbeMeter, micro button, and a wide offset compensation range.

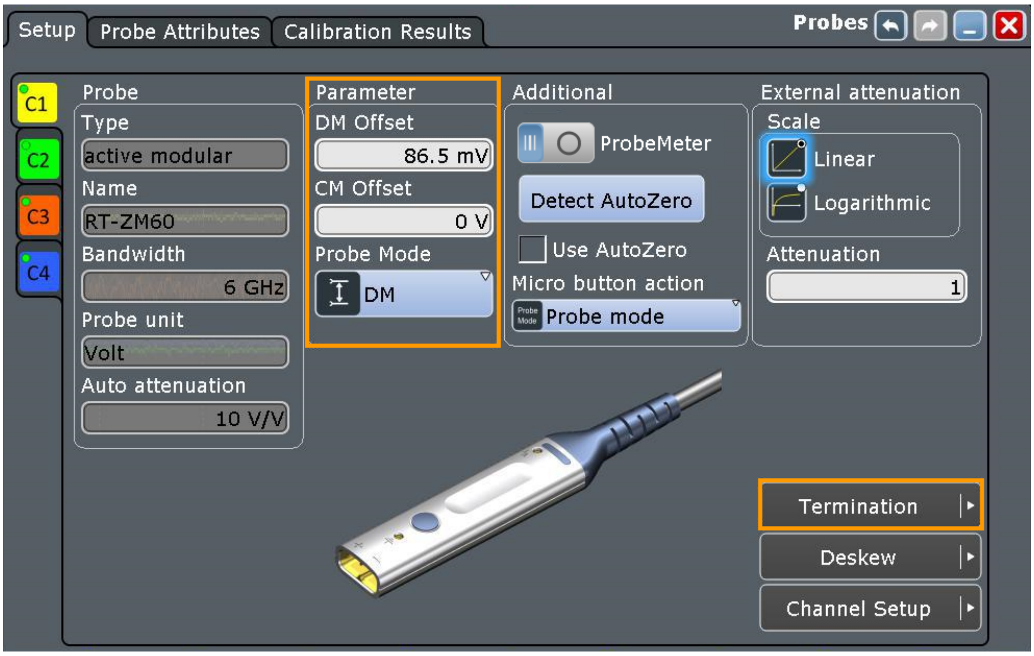
If you connect the R&S RT-ZM probe to the external trigger input, you can adjust only the trigger level and the probe mode. The instrument sets the trigger level to the probe offset value (DM, CM, P or N, depending on the probe mode).

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 157
- "Detect AutoZero, Use AutoZero" on page 157
- "External attenuation: Scale, Attenuation" on page 158

Special features of the Rohde & Schwarz probe interface are described in these sections:

- "Probe unit, Auto attenuation" on page 159
- "Micro button action" on page 159
- "ProbeMeter" on page 160



- Setup Parameters of Modular Probes....................169
- Termination Voltage (Only with R&S RT-ZMA40 SMA Module)....................170

4.5.5.1 Setup Parameters of Modular Probes

The basic setup parameters of all modular probes are the measurement mode and the offset settings.

Access: "Vertical" menu > "Probe Setup"

Probe Mode

Sets the measurement mode of modular probes.

The modular probes of the R&S RT-ZM family have a multi-mode function. You can switch between single-ended, differential and common mode measurements without reconnecting or resoldering the probe. You can set the probe mode in the dialog box, and you can assign the probe mode setting to the micro button.

If you use the R&S RT-ZMA30 browser module, only DM measurements are possible because this module has no ground connector.

The measurement modes are:

"DM" Differential mode input voltage (V_{dm}), the voltage between the positive and negative input terminal.

$$V_{dm} = V_p - V_n$$

"CM" Common mode input voltage (V_{cm}), the mean voltage between the positive and negative input terminal vs. ground.

$$V_{cm} = \frac{V_p + V_n}{2}$$

"P" Positive single-ended input voltage (V_p). The voltage between the positive input terminal and ground.

"N" Negative single-ended input voltage (V_N). The voltage between the negative input terminal and ground.

Remote command:
PROBe<m>:SETup:PRMode on page 1098
TRPRobe:SETup:PRMode on page 1098

DM Offset, CM Offset, P Offset, N Offset
Compensate offset voltages. Available offsets depend on the selected probe mode.

The offset of the selected probe mode is used as channel offset and considered automatically for correction. For example, in CM mode, the common mode offset is used as channel offset. See also: "Offset" on page 149.

"DM Offset" Compensates a DC voltage applied between the positive (V_p) and the negative (V_n) input terminal at the probe tip.

"CM Offset" Compensates a DC voltage applied to both input terminals referenced to ground. See also: "CM offset" on page 165.

"P Offset" Compensates a DC voltage applied to the positive input terminal (V_p) referenced to ground.

"N Offset" Compensates a DC voltage applied to the negative input terminal (V_n) referenced to ground.

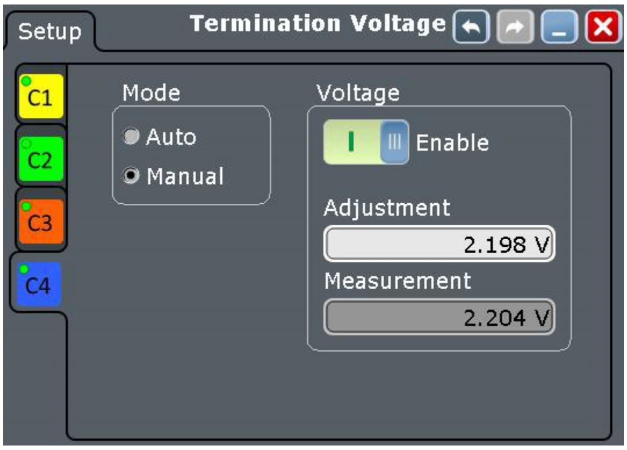
Remote command:
PROBe<m>:SETup:DMOFfset on page 1099
PROBe<m>:SETup:CMOFfset on page 1095
PROBe<m>:SETup:NOFFset on page 1099
PROBe<m>:SETup:POFFset on page 1100

4.5.5.2 Termination Voltage (Only with R&S RT-ZMA40 SMA Module)

Termination voltage is relevant if you use the R&S RT-ZMA40 SMA module. The SMA module applies a termination voltage (±4 V) to the DUT to enable measurements against a common mode DC voltage instead of ground. This measurement is required for many digital signal standards.

The termination voltage can be controlled by the oscilloscope. Therefore, connect the V_T terminal of the R&S RT-ZM probe amplifier to the V_T terminal of the R&S RT-ZMA40 SMA module using the red DC lead (see R&S RT-ZMUser Manual). The required termination voltage is measured and adjusted automatically, but can also be set manually.

Access: "Vertical" menu > "Probe Setup" > "Termination"



Mode
In "Auto" mode, the instrument uses the measured common mode voltage for termination.

In "Manual" mode, you can enter the voltage to be used for termination. Use the manual mode if you know the common mode voltage of the DUT.

Remote command:
`PROBe<m>:SETup:TERM:MODE` on page 1100

EnableTermination Voltage
Activates control of the termination voltage.

Remote command:
`PROBe<m>:SETup:TERM:STATe` on page 1100

Adjustment
Sets the voltage to be used for termination to DC voltage.

Remote command:
`PROBe<m>:SETup:TERM:ADJust` on page 1101

Measurement
Shows the measured common mode voltage.

Remote command:
`PROBe<m>:SETup:TERM:MEASure?` on page 1101

4.5.6 Setup for Predefined Probes

Probes that cannot be detected, but their characteristics are known to the R&S RTE are called "Predefined probes".

The following shared probe settings are available:
- "Type, Name, Bandwidth" on page 157
- "Detect AutoZero, Use AutoZero" on page 157
- "External attenuation: Scale, Attenuation" on page 158

The probe attenuation of the selected probe is also shown in the "Setup" tab:

- "Probe unit, Auto attenuation" on page 159

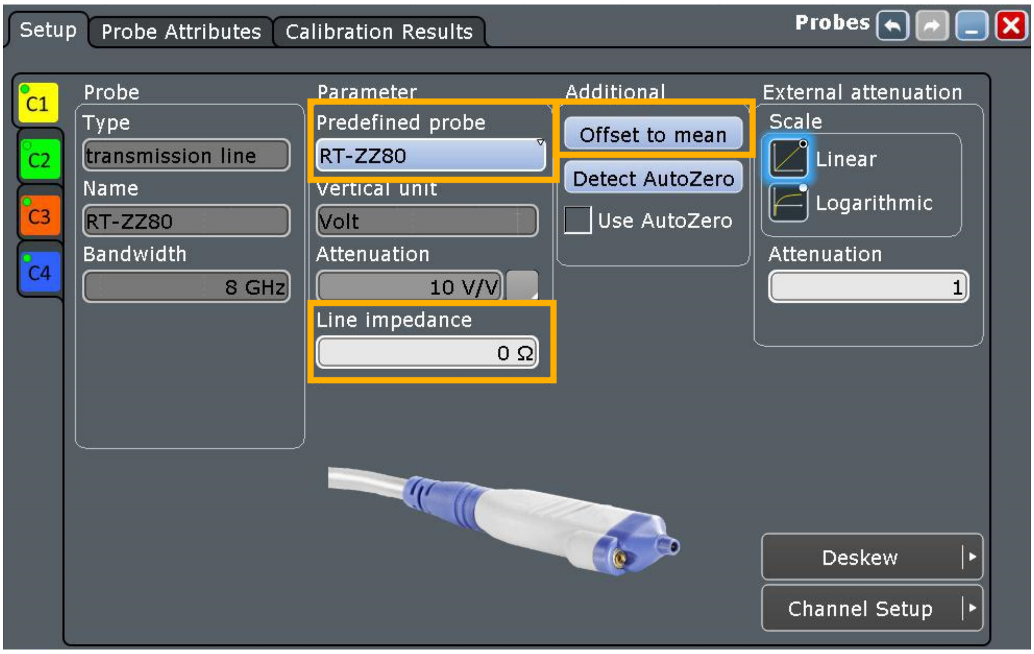


Figure 4-14: Probe setup for transmission line probe R&S RT-ZZ80

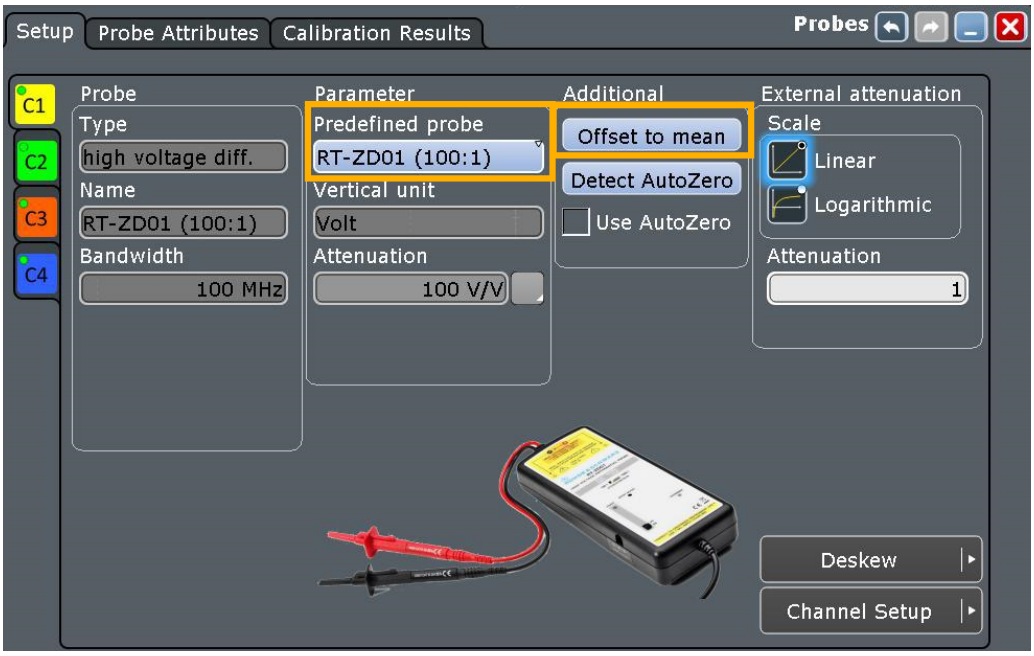


Figure 4-15: Probe setup for R&S RT-ZD01

Specific settings for predefined probes are the following:

Predefined probe....173
Offset to mean....173
Line impedance....173

Predefined probe
List of probes that are known to the instrument. The instrument lists only probes that match the instrument coupling and, if connected, also the used adapter.

Select the used probe on the list. The corresponding "Vertical unit" and the "Attenuation" or "Gain" are shown.

All other unrecognized probes that are not listed, are unknown probes. For these probes, set "Predefined probe" to "None". See Chapter 4.5.8, "Setup for Unknown Probes", on page 176.

Remote command:
`PROBe<m>:SETup:ATTenuation:DEFProbe` on page 1101
`TRPRobe:SETup:ATTenuation:DEFProbe` on page 1101 (trigger input)

Offset to mean
Performs an automatic compensation for a DC component of the input signal using the result of a background mean measurement. The result is shown in "Offset". The function is probe-independent and supports quick and convenient measurements of input signals with different DC offsets. It detects offset values even when the signal is out of the current measurement range. It also sets the zero level to the determined DC offset in the middle of the screen and thus prevents clipping of the waveform.

Remote command:
`PROBe<m>:SETup:OFFSet:TOMean` on page 1102

Line impedance
If the transmission line probe R&S RT-ZZ80 is selected, enter the impedance of the measured line.

The actual attenuation of the transmission line probe depends on the impedance of the line Z_0:

$Attenuation = 10 + Z_0 / 100$

The instrument uses the actual attenuation to determine the measurement values.

4.5.7 Setup for Current Probes

The setup and adjustment of current probes depends on the output connector of the probe: BNC or Rohde & Schwarz probe box.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 157
- "Detect AutoZero, Use AutoZero" on page 157
- "External attenuation: Scale, Attenuation" on page 158

Current probes R&S RT-ZCxx

The current probes **R&S RT-ZCxx** have BNC connectors. They are known to the R&S RTE as predefined probes, see Chapter 4.5.6, "Setup for Predefined Probes", on page 171. Demagnetizing and zero adjustment is done on the probe, see the probe's User Manual for details. Make sure to demagnetize and adjust the probe before taking measurements.

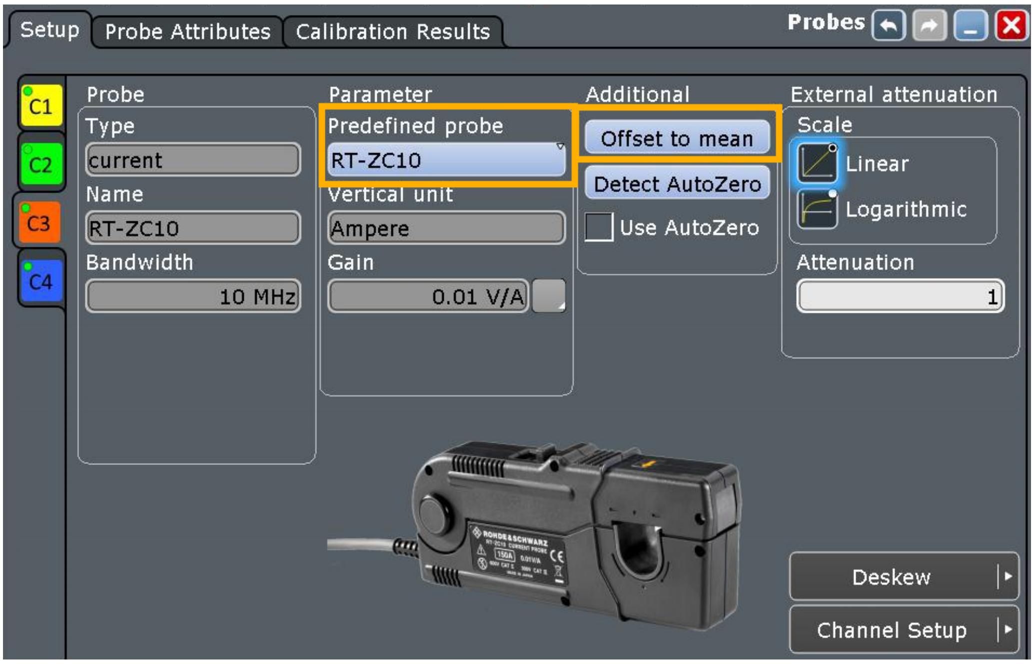


Figure 4-16: Probe setup for current probes R&S RT-ZC10

Current probes R&S RT-ZCxxB

Current probes **R&S RT-ZCxxB** have a Rohde & Schwarz probe interface; they are powered and remotely controlled by the oscilloscope.

When the probe is connected, demagnetization is performed automatically.

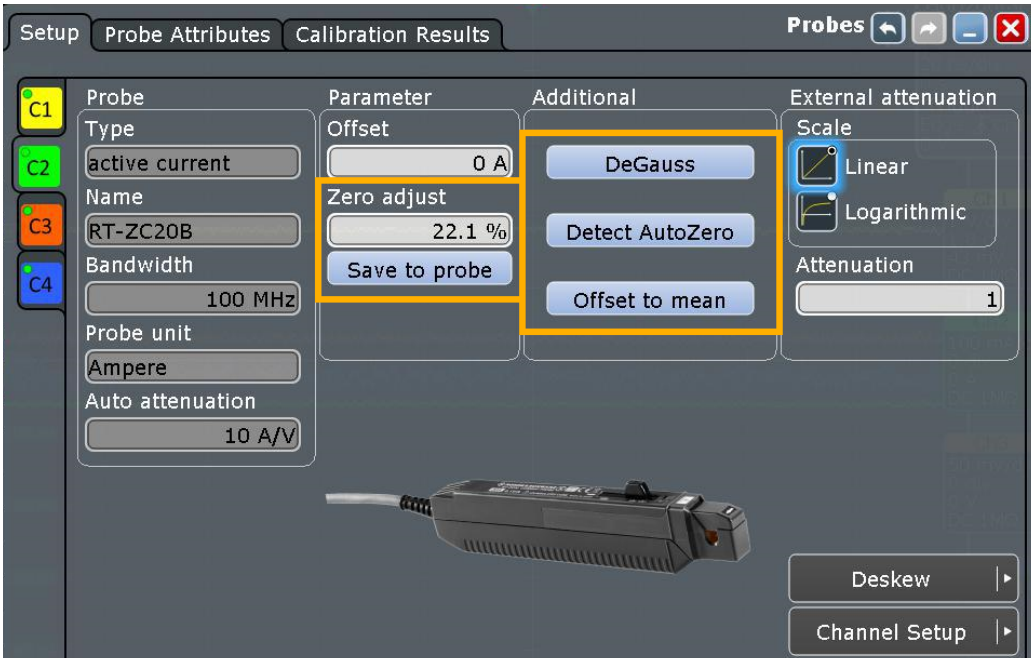


Figure 4-17: Probe setup for current probes R&S RT-ZC20B

For all current probes, attenuation or gain is shown in the "Setup" tab, and you can set the offset to mean:

- "Probe unit, Auto attenuation" on page 159
- "Offset to mean" on page 173

Current probes R&S RT-ZCxxB are adjusted by the following functions:

DeGauss....175
Detect AutoZero....175
Zero adjust....176
Save to probe....176

DeGauss

Demagnetizes the core if it has been magnetized by switching the power on and off, or by an excessive input. Always carry out demagnetizing before measurement.

The demagnetizing process takes about one second. During demagnetizing, a demagnetizing waveform is displayed.

Demagnetizing is done automatically when R&S RT-ZCxxB is connected to the oscilloscope, or when "Detect AutoZero" is performed.

Remote command:

`PROBe<m>:SETup:DEGauss` on page 1103

`TRPRobe:SETup:DEGauss` on page 1103

Detect AutoZero

If a current probe is connected, the function demagnetizes the probe's sensor head and sets the waveform to zero position to correct the error offset. Thus, it compensates for the remanence and offset caused by temperature drift.

For R&S RT-ZCxxB probes, the determined "Zero adjust" value is displayed and can be saved in the probe head.

See also "Detect AutoZero, Use AutoZero" on page 157.

Remote command:
`PROBe<m>:SETup:OFFSet:AZERo` on page 1088

Zero adjust
Zero adjust corrects the effect of an offset caused by temperature drift, and compensates for the remanence. The setting is only available if DC coupling is set.

To set the waveform to zero level by the instrument, use "Detect AutoZero". The detected value is displayed.

Alternatively, you can adjust the value manually until the waveform is set to zero level. Make sure to demagnetize the probe before zero adjustment.

The value is given in percent of the maximum range, which is internally defined. The actual setup range depends on the temperature drift, the measured current and other variables, and it can change over time. If you measure high currents, the probe core magnetizes, which impairs the measurement results. Therefore, repeat "Detect AutoZero" before the measurement.

Remote command:
`PROBe<m>:SETup:OFFSet:ZADJust` on page 1103

Save to probe
Saves the "Zero adjust" value in the probe box. If you connect the probe to another channel or to another R&S RTx oscilloscope, the value is read out again, and you can use the probe without further adjustment.

Remote command:
`PROBe<m>:SETup:OFFSet:STPRobe` on page 1103

4.5.8 Setup for Unknown Probes

If the R&S RTE cannot detect the probe, and the probe is not a predefined one, you can set the probe parameters manually.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 157
- "Detect AutoZero, Use AutoZero" on page 157
- "External attenuation: Scale, Attenuation" on page 158

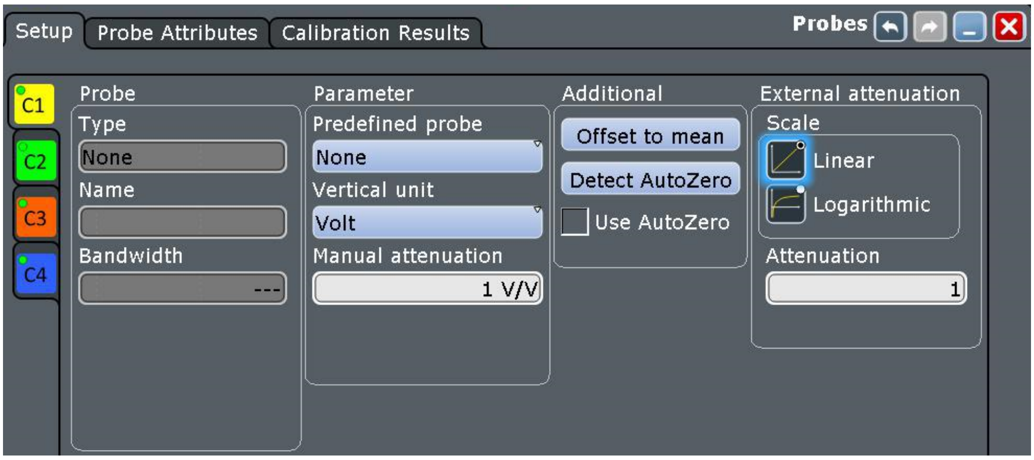


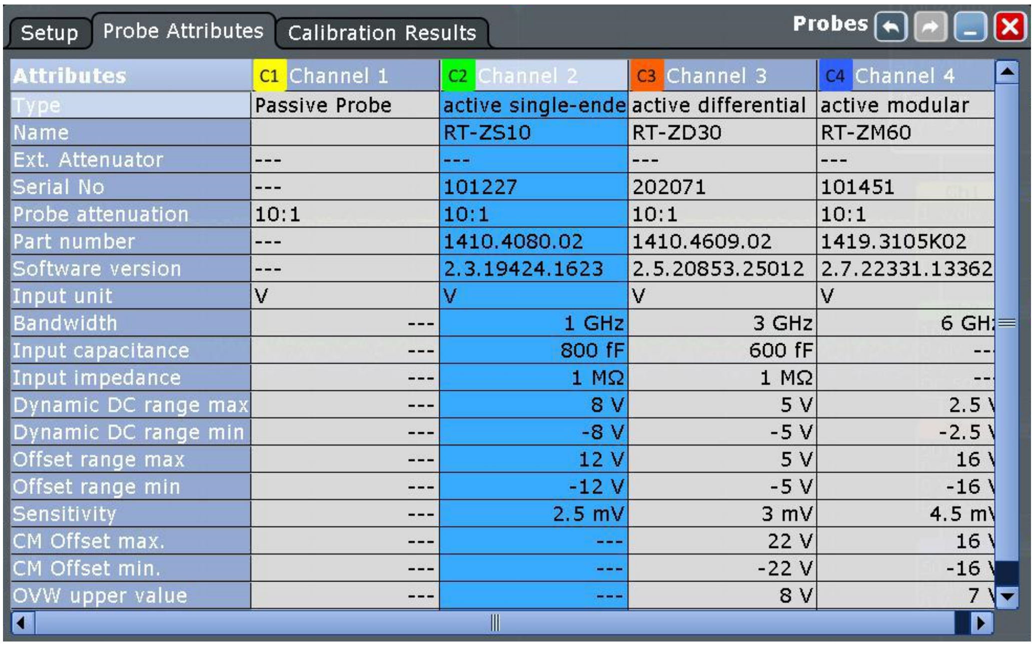
Figure 4-18: Probe setup for an unknown probe

Set the unit and the attenuation or gain of the probe: "Vertical unit, Attenuation, Gain" on page 163.

4.5.9 Probe Attributes

The "Probe Attributes" tab provides an overview of all R&S probes connected to an input channel.

For a specification of the probe parameters, refer to the data sheet.



Attributes	C1 Channel 1	C2 Channel 2	C3 Channel 3	C4 Channel 4
Type	Passive Probe	active single-ende	active differential	active modular
Name		RT-ZS10	RT-ZD30	RT-ZM60
Ext. Attenuator	---	---	---	---
Serial No	---	101227	202071	101451
Probe attenuation	10:1	10:1	10:1	10:1
Part number	---	1410.4080.02	1410.4609.02	1419.3105K02
Software version	---	2.3.19424.1623	2.5.20853.25012	2.7.22331.13362
Input unit	V	V	V	V
Bandwidth	---	1 GHz	3 GHz	6 GH
Input capacitance	---	800 fF	600 fF	--
Input impedance	---	1 MΩ	1 MΩ	--
Dynamic DC range max	---	8 V	5 V	2.5 V
Dynamic DC range min	---	-8 V	-5 V	-2.5 V
Offset range max	---	12 V	5 V	16 V
Offset range min	---	-12 V	-5 V	-16 V
Sensitivity	---	2.5 mV	3 mV	4.5 mV
CM Offset max.	---	---	22 V	16 V
CM Offset min.	---	---	-22 V	-16 V
OVW upper value	---	---	8 V	7 V

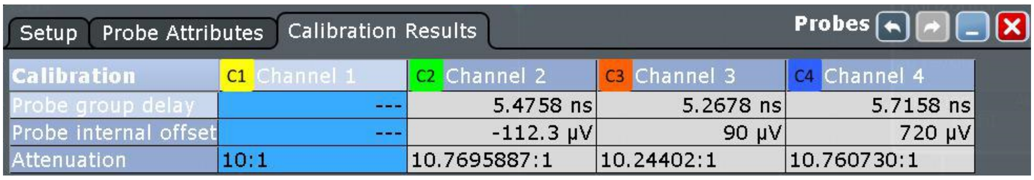
Remote commands:

- PROBe<m>:ID:SWVersion? on page 1104
- PROBe<m>:ID:PRDate? on page 1104

- PROBe<m>:ID:PARTnumber? on page 1104
- PROBe<m>:ID:SRNumber? on page 1105
- PROBe<m>:SETup:CAPacitance? on page 1105
- PROBe<m>:SETup:IMPedance? on page 1105
- TRPRobe:ID:SWVersion? on page 1104
- TRPRobe:ID:PRDate? on page 1104
- TRPRobe:ID:PARTnumber? on page 1104
- TRPRobe:ID:SRNumber? on page 1105
- TRPRobe:SETup:CAPacitance? on page 1105
- TRPRobe:SETup:IMPedance? on page 1105

4.5.10 Calibration Results

The "Calibration Results" tab provides the calibration data stored in the probe for all R&S probes connected to an input channel.



Calibration	C1 Channel 1	C2 Channel 2	C3 Channel 3	C4 Channel 4
Probe group delay	---	5.4758 ns	5.2678 ns	5.7158 ns
Probe internal offset	---	-112.3 µV	90 µV	720 µV
Attenuation	10:1	10.7695887:1	10.24402:1	10.760730:1

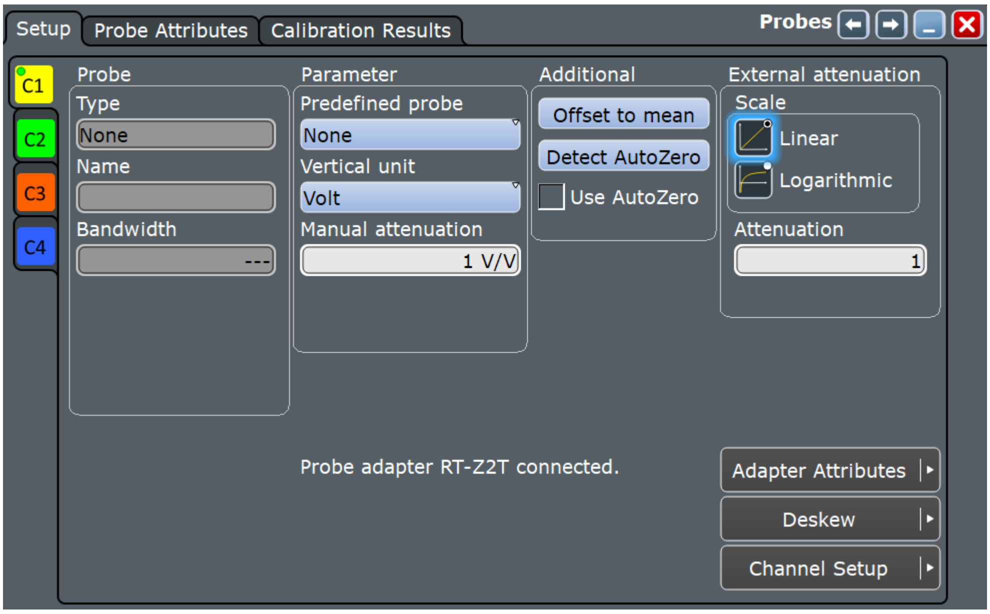
4.5.11 Probe Adapter R&S RT-Z2T

Using the R&S RT-Z2T probe interface adapter, you can connect selected Tektronix active probes with TekProbe BNC™ level II interface.

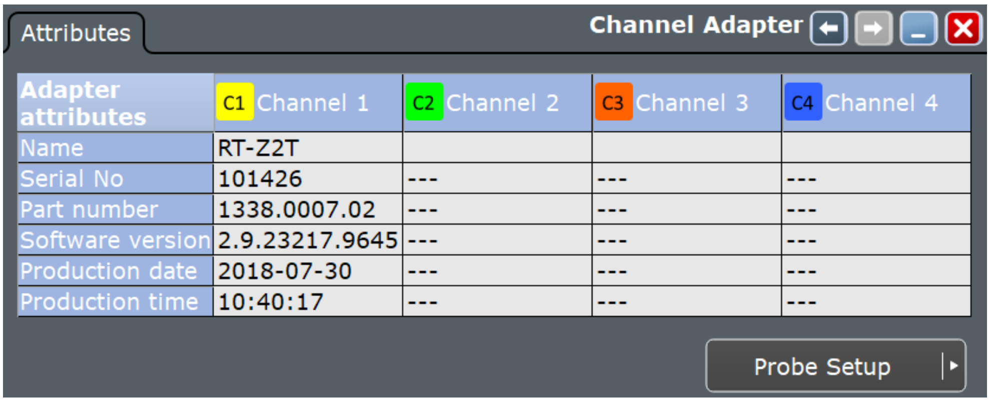
1. Connect the R&S RT-Z2T adapter to the channel input.
2. Connect the probe to the adapter.

   The instrument identifies the adapter. The R&S RTE lists the supported Tektronix probes as "Predefined probe".
3. Select "Vertical" menu > "Probe Setup".

   The dialog shows that the R&S RT-Z2T probe interface adapter is connected.



4. To see the information on the adapter, select "Adapter Attributes".



Adapter attributes	C1 Channel 1	C2 Channel 2	C3 Channel 3	C4 Channel 4
Name	RT-Z2T			
Serial No	101426	---	---	---
Part number	1338.0007.02	---	---	---
Software version	2.9.23217.9645	---	---	---
Production date	2018-07-30	---	---	---
Production time	10:40:17	---	---	---

5. Select the "Predefined probe" in the "Probe Setup".
   See also: Chapter 4.5.6, "Setup for Predefined Probes", on page 171.

Remote commands:
- `PROBe<m>:SETup:ADAPter?` on page 1106
- `PROBe<m>:SETup:ATTenuation:TDEFprobe` on page 1106

4.6 R&S RT-ZVC Probe

With the R&S RTE and option R&S RTE-B1E, you can use the R&S RT-ZVC multi-channel power probe. It has an integrated 2- or 4-channel amperemeter and 2- or 4-channel voltmeter. The probe provides parallel measurements of analog or digital, voltage/current signals with excellent 18-bit resolution.

For more information on the R&S RT-ZVC probe, see also its user manual.

Source Channels

You can simultaneously connect a R&S RT-ZVC and a R&S RT-ZL04 to the R&S RTE, but no parallel operation on screen is possible.

You can acquire and measure the R&S RT-ZVC or R&S RT-ZL04 together with the analog input channels. They are running on the same horizontal scale.

If an amperemeter or voltmeter channel is activated, it can be displayed on the screen and used as a source for:

- Cursor measurements
- Automatic measurements, and also histogram, limit checks, longterm, track, quick measurements
- Mask testing
- Basic and advanced maths
- Reference waveform
- Trigger type "Edge"
- Search: all search events
- XY diagrams
- Data export

All features of the base unit and the R&S RTE-K18 option are supported.

Data export

You can save the data of the amperemeter and voltmeter channels to an XML, CSV, or BIN file. One channel per file can be saved. Files in BIN format can be reloaded to the R&S RTE as reference waveforms.

See also:

- Chapter 11.2.7, "Saving and Loading Waveform Data", on page 457
- Chapter 11.2.2, "Waveforms - Export Settings", on page 445

Remote commands for export to file:

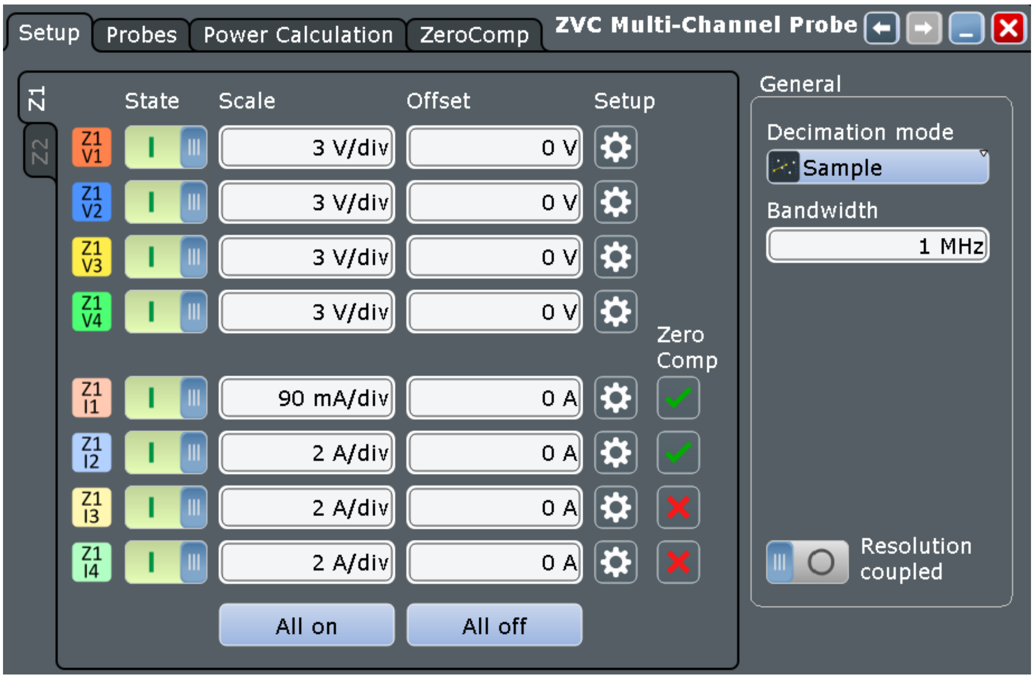
- `EXPort:WAVeform:SOURce` on page 1363
- `EXPort:WAVeform:NAME` on page 1364
- `EXPort:WAVeform:SAVE` on page 1365

4.6.1 R&S RT-ZVC Overview

4.6.1.1 Setup

Access: "Vertical" menu > "ZVC Multi-Channel Probe" > "Setup" tab

Make sure that the tab of the correct probe is selected on the left side.



State
Enables the corresponding channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
`ZVC:Z<m>:V<n>[:STATe]` on page 1116
`ZVC:Z<m>:I<n>[:STATe]` on page 1113

Scale
Defines the vertical scale for the channel in Volts per division. Increasing the scale compresses the display of the signal. Within a given operation range, modifying the scale is equivalent to scaling a display range.

For the voltmeter channels, the Scale, the Offset and the "Position" specify the operating range of the voltmeter.

Remote command:
`ZVC:Z<m>:V<n>:SCALe` on page 1115
`ZVC:Z<m>:I<n>:SCALe` on page 1111

Offset
The vertical center of the selected channel is shifted by the offset value and the signal is repositioned within the diagram area. Negative offset values move up the waveform, positive values move it down.

Within a given operation range, modifying the offset is equivalent to moving vertically the display range. The offset can only be modified such that the display range reaches at most the limits of the operation range.

For the voltmeter channels, the Scale, the Offset and the "Position" specify the operating range of the voltmeter.

Remote command:
`ZVC:Z<m>:V<n>:OFFSet` on page 1114
`ZVC:Z<m>:I<n>:OFFSet` on page 1109

Setup
Opens the voltage or current settings for the selected channel. See Chapter 4.6.2, "ZVC Voltage Settings", on page 185 and Chapter 4.6.3, "ZVC Current Settings", on page 187.

Zero Comp
Shows the status of the zero compensation of each current channel. The green checkmark indicates that the zero offset is compensated automatically. To adjust the settings, tap the icon. For details, see Chapter 4.6.1.4, "Zero Compensation", on page 184.

Remote command:
`ZVC:Z<m>:I<n>:ZERComp:STATe?` on page 1118

All on
Enables all available channels.

All off
Disables all available channels.

Decimation mode
Selects the decimation mode for all R&S RT-ZVC probes. Decimation reduces the data stream of the ADC to a stream of waveform points with lower sample rate and a less precise time resolution.

"Sample"	One of n samples in a sample interval of the ADC is recorded as waveform point, the other samples are discarded. The time between the two adjacent waveform points is exactly the resolution. Very short glitches might remain undiscovered by this method.
"Peak detect"	The minimum and the maximum of n samples in a sample interval are recorded as waveform points, the other samples are discarded.
"High res"	The average of n sample points is recorded as one waveform sample. Averaging reduces the noise, the result is a more precise waveform with higher vertical resolution. The high measurement resolution is suitable for high accuracy measurements of instantaneous values.

Remote command:
`ZVC:TYPE` on page 1108

Bandwidth
Sets the bandwidth limit of all R&S RT-ZVC probes. The bandwidth specifies the maximum frequency at which a purely sinusoidal signal is still transferred at 89 % (0.1 dB) of its amplitude.

The bandwidth of some current channels is restricted to 300 KHz due to their vertical settings.

Remote command:
`ZVC:BANDwidth` on page 1107

Resolution coupled
Sets the resolution of all R&S RT-ZVC channels.

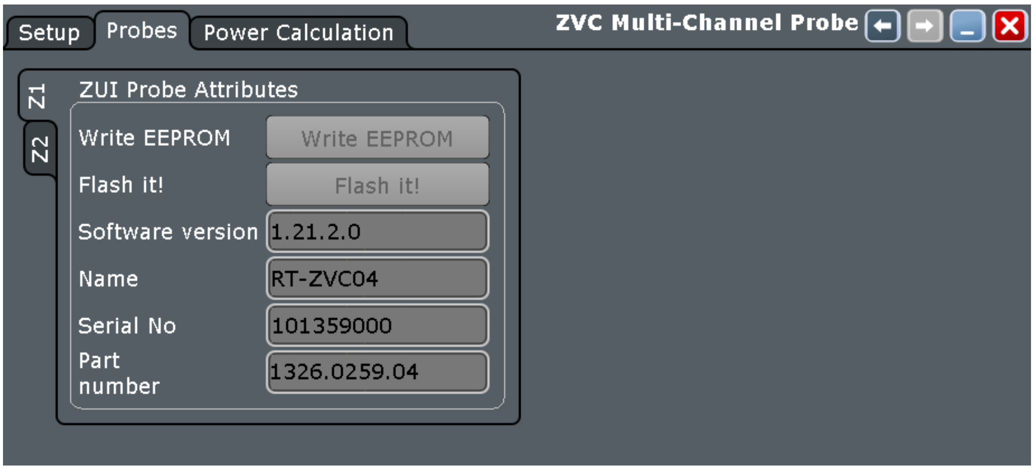
"On" The resolution of the analog channels is applied to R&S RT-ZVC channels. The signal is automatically interpolated or decimated to get the analog resolution.

"Off" The resolution of R&S RT-ZVC channels is set in a way so that the record length of the waveforms is minimum 1000 samples.

Remote command:
`ZVC:RESCoupled` on page 1109

4.6.1.2 Probes

Access: "Vertical" menu > "ZVC Multi-Channel Probe"> "Probes" tab



Software version
Displays the software version of the probe.

Remote command:
`ZVC:Z<m>:ID:SWVersion?` on page 1117

Name
Displays the name of the probe.

Remote command:
`ZVC:Z<m>:ID:NAME?` on page 1116

Serial no
Displays the serial number of the probe.

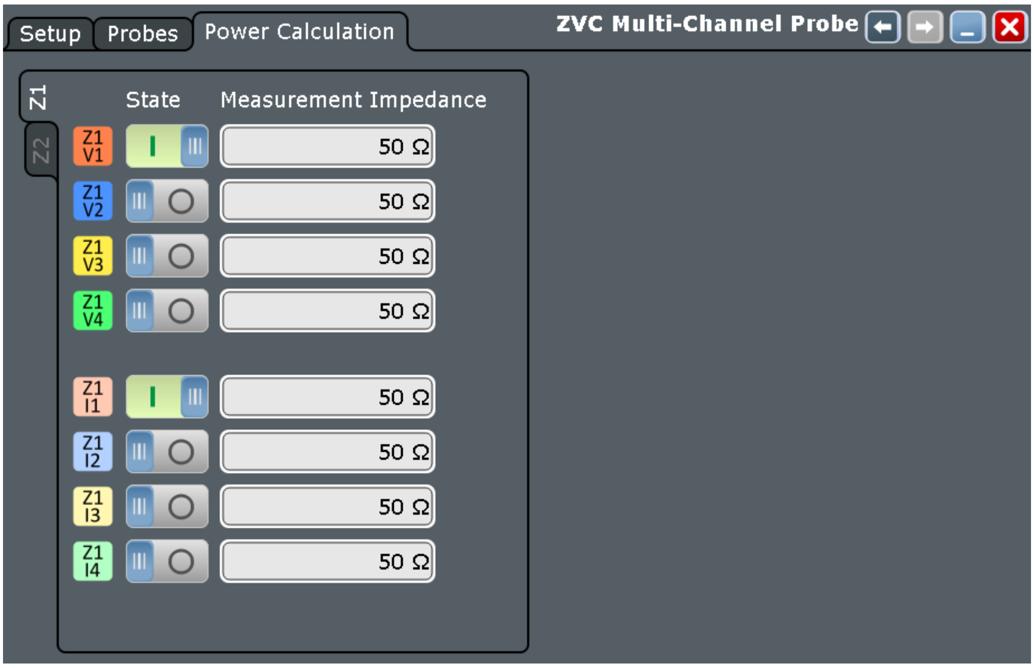
Remote command:
`ZVC:Z<m>:ID:SRNumber?` on page 1117

Part number
Displays the material number of the probe.

Remote command:
ZVC:Z<m>:ID:PARTnumber? on page 1116

4.6.1.3 Power Calculation

Access: "Vertical" menu > "ZVC Multi-Channel Probe" > "Power Calculation" tab



State
Enables the corresponding channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
ZVC:Z<m>:V<n>[:STATe] on page 1116
ZVC:Z<m>:I<n>[:STATe] on page 1113

Measurement Impedance
Sets the impedance of the probe channel for power calculations and measurements.

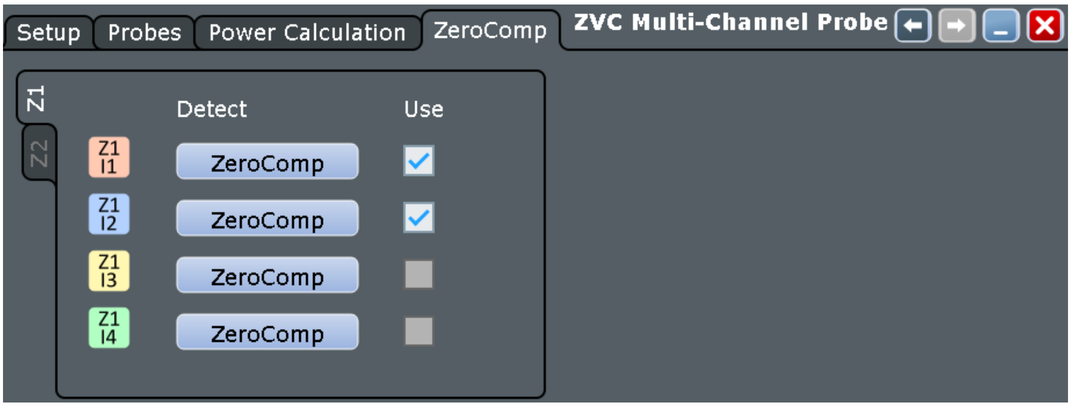
Remote command:
ZVC:Z<m>:V<n>:IMPedance on page 1114
ZVC:Z<m>:I<n>:IMPedance on page 1108

4.6.1.4 Zero Compensation

Access: "Vertical" menu > "ZVC Multi-Channel Probe" > "ZeroComp" tab

Zero compensation is used to avoid negative currents in measurement results, and to improve the measurement accuracy. The determined compensation is valid as long as the temperature is constant and the probe settings are unchanged.

1. Disconnect the DUT from the power supply.
2. Adjust the current settings, in particular, the shunt settings. See Chapter 4.6.3, "ZVC Current Settings", on page 187.
3. Connect one probe tip of the R&S RT-ZVC probe to the high voltage pin of the DUT. Disconnect the other probe tip from the DUT.
4. Open the "ZeroComp" settings: "Vertical" menu > "ZVC Multi-Channel Probe" > "ZeroComp" tab
5. Tap "ZeroComp" to determine the zero offset.
6. Enable "Use" to compensate for the zero offset.



ZeroComp
Measures the zero offset, the mean value on a currentless DUT. If temperature changes, or if you change the shunt or other probe settings, repeat the measurement.

Remote command:
`ZVC:Z<m>:I<n>:ZERComp:DETect` on page 1117

Use
If enabled, the measured "ZeroComp" offset value is compensated automatically.

Remote command:
`ZVC:Z<m>:I<n>:ZERComp:USE` on page 1117

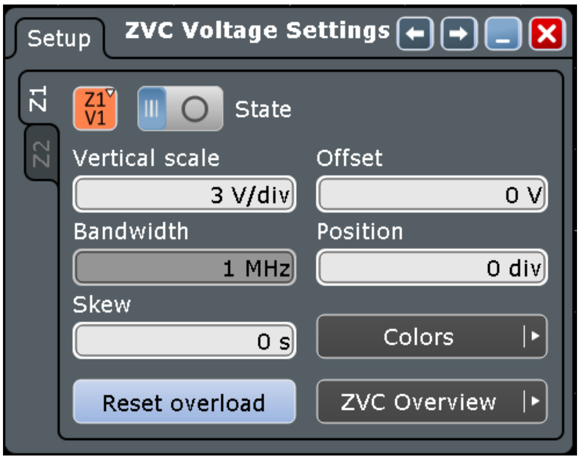
4.6.2 ZVC Voltage Settings

Access: "Vertical" menu > "ZVC Multi-Channel Probe" > "Setup" dialog > "Setup" icon of voltage channel

See also:

- Scale

- Offset



Channel
Selects the voltmeter channel to be configured.

State
Enables the corresponding voltage channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
`ZVC:Z<m>:V<n>[:STATe]` on page 1116

Bandwidth
Displays the bandwidth of the current channel. You can set the probe bandwidth in the "Setup" dialog.

Remote command:
`ZVC:Z<m>:V<n>:BANDwidth?` on page 1113

Position
Moves the selected signal up or down in the diagram. The visual effect is the same as for Offset. While the offset sets a voltage, position is a graphical setting given in divisions. Within a given operation range, modifying the position is equivalent to moving vertically the display range. The position can only be modified such that the display range reaches at most the limits of the operation range.

For the voltmeter channels, the Scale, the Offset and the "Position" specify the operating range of the voltmeter.

Remote command:
`ZVC:Z<m>:V<n>:POSition` on page 1115

Skew
Sets the skew, a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Remote command:
`ZVC:Z<m>:V<n>:SKEW` on page 1116

Reset Overload

Resets the overload indication at the probe.

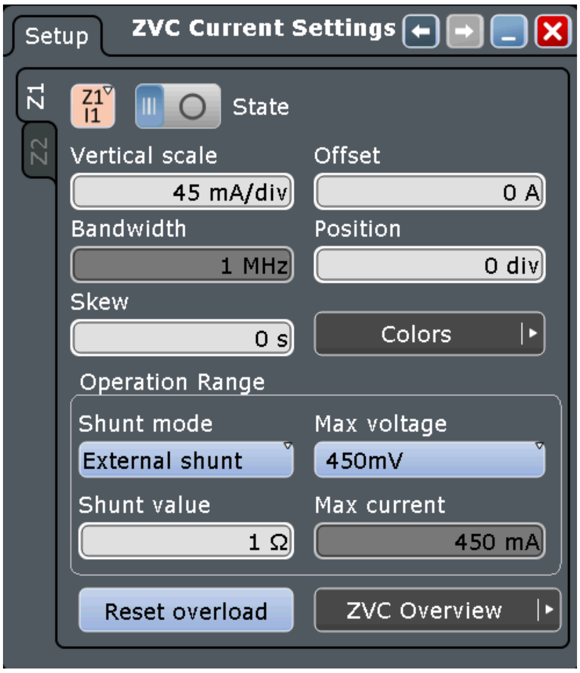
Remote command:
ZVC:Z<m>:V<n>:OVERload:RSTO on page 1114

4.6.3 ZVC Current Settings

Access: "Vertical" menu > "ZVC Multi-Channel Probe" > "Setup" dialog > "Setup" of current channel

See also:

- Scale
- Offset



Channel

Selects the amperemeter channel to be configured.

State

Enables the corresponding current channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
ZVC:Z<m>:I<n>[:STATe] on page 1113

Bandwidth

Displays the bandwidth of the current channel. You can set the probe bandwidth in the "Setup" dialog.

Remote command:
`ZVC:Z<m>:I<n>:BANDwidth?` on page 1108

Position
Moves the selected signal up or down in the diagram. The visual effect is the same as for Offset. While the offset sets a current, position is a graphical setting given in divisions. Within a given operation range, modifying the position is equivalent to moving vertically the display range. The position can only be modified such that the display range reaches at most the limits of the operation range.

Remote command:
`ZVC:Z<m>:I<n>:POSition` on page 1110

Skew
Sets the skew, a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Remote command:
`ZVC:Z<m>:I<n>:SKEW` on page 1113

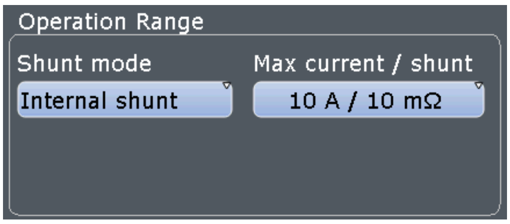
Shunt mode
Selects the internal or external shunt mode.

Regarding the shunt selection, i.e. the burden voltage level, there is a trade-off between the burden of the circuit under test and the SNR at the frontend input. The burden voltage is the DUT circuit loading caused by leads, connectors and the amperemeter circuit.

From the DUT perspective, the burden voltage has to be kept low not to distort the device operation. In contrast, from the probe's view the voltage has to be as large as possible to obtain a good SNR. For that reason, the external shunt can be applied to get the best compromise of both for a specific measurement range.

Remote command:
`ZVC:Z<m>:I<n>:SHUNt:MODE` on page 1111

Internal Shunt Mode
If Shunt mode is set to "Internal shunt", include the settings for the internal shunt mode.



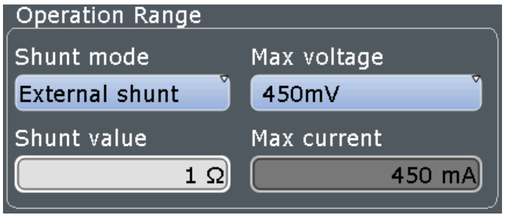
Max current / shunt ← Internal Shunt Mode
Selects the maximum current and the internal shunt value.

With the maximum current and the internal shunt selection, the operating range of the amperemeter is specified. At the same time, the burden voltage at the amperemeter input can be estimated. For values of the total round-trip resistance that can be seen at the test lead ends, consider the data sheet.

For using internal shunts, the circuit under test needs to be interrupted so that the current can flow through the probe.

Remote command:
ZVC:Z<m>:I<n>:SHUNt:MAXCurrent on page 1111

External shunt mode
If Shunt mode is set to "External shunt", include the settings for the external shunt mode.



Maximum voltage ← External shunt mode
Selects the maximum voltage at the external shunt.

Remote command:
ZVC:Z<m>:I<n>:SHUNt:MAXVoltage on page 1112

Shunt value ← External shunt mode
Sets the shunt value of the external shunt resistor.

Remote command:
ZVC:Z<m>:I<n>:SHUNt:EVALue on page 1111

Max current ← External shunt mode
Displays the maximum current.

Remote command:
ZVC:Z<m>:I<n>:SHUNt:MXCValue? on page 1112

Reset Overload ← External shunt mode
Resets the overload indication at the probe.

Remote command:
ZVC:Z<m>:I<n>:OVERload:RSTO on page 1109

4.7 Differential Signals

With R&S RTE, you can easily analyze differential signals using single-ended probes, or even cables. The instrument processes the input of single-ended probes on analog channels and creates the differential and common mode waveforms. Similar to Rohde & Schwarz modular probes, you can display the waveforms of differential voltage, common mode voltage, positive single-ended voltage and negative single-ended voltage. Triggering on differential signals is also possible.

4.7.1 Settings of Differential Signals

Access: "Vertical" menu > "Differential Signals"

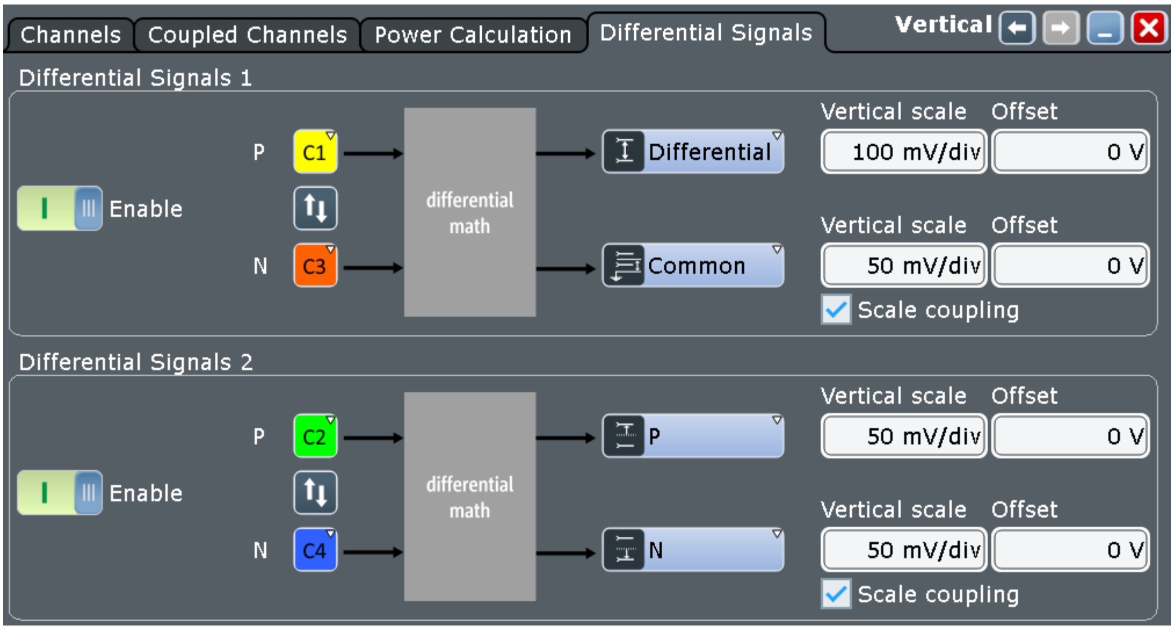


Figure 4-19: Setup of differential signals

Enable

Activates the differential signal.

Remote command:

DIFFerential<m>:STATe on page 1120

P, N, Switch

Select the analog input channel of the positive and negative signal.

For differential signal 1, analog Ch1 and Ch3 are used. Differential signal 2 uses Ch2 and Ch4. You can switch the channels with the arrow button.

Remote command:

DIFFerential<m>:PSIGnal[:SELect] on page 1121

DIFFerential<m>:NSIGnal[:SELect] on page 1121

Output: P, N, Differential, Common

Select the waveforms that are displayed as result of differential processing.

For each differential signal, 2 output waveforms are displayed.

In the diagram, you see the output waveforms, and the signal icons of the input channels are grayed out. The input channels are only visible if "P" or "N" are selected as output.

"P", "N" Displays the waveform of the positive or negative input signal. The input channel is shown directly, no additional waveform is created.

"Differential" Displays the differential signal as a new waveform.

"Common" Displays the common mode signal as a new waveform.

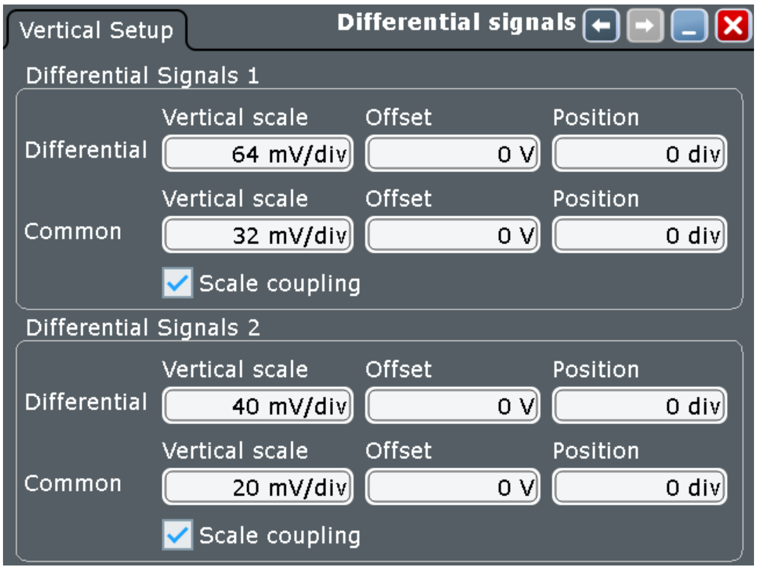
Remote command:
`DIFFerential<m>:AOUTput` on page 1121
`DIFFerential<m>:BOUTput` on page 1121

Vertical scale, Offset
Vertical scale and offset of differential and common mode waveforms can be set directly in the differential setup, or in the "Vertical Setup" of differential signals. See: "Vertical scale" on page 191 and "Offset" on page 192.

Vertical settings of P and N output are the vertical settings of the input channels.

4.7.2 Vertical Setup of Differential Signals

Access: "Vertical" menu > "Differential Signals" > "Vertical Setup"



Vertical scale
Sets the vertical scale of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Remote command:
`DIFFerential<m>:COMMon:SCALe` on page 1121
`DIFFerential<m>:DIFFerential:SCALe` on page 1121

Scale coupling
If enabled, the vertical scales of P, N, differential and common mode waveforms are coupled. The scales are related as follows:

$VertScale_P = VertScale_N = VertScale_{CM} = VertScale_{Diff}/2$

You can disable the scale coupling and set an individual scale for each waveform.

Remote command:
`DIFFerential<m>:COUPling` on page 1122

Offset
Sets the offset of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Remote command:
`DIFFerential<m>:COMMon:OFFSet` on page 1122
`DIFFerential<m>:DIFFerential:OFFSet` on page 1122

Position
Sets the vertical position of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Remote command:
`DIFFerential<m>:COMMon:POSition` on page 1122
`DIFFerential<m>:DIFFerential:POSition` on page 1122

4.7.3 Analysis of Differential Signals

Before analyzing the output waveforms of differential signals, make sure that the P and N signals are correctly aligned. If necessary, use "Horizontal" > "Skew" to deskew the input waveforms.

For all output waveforms of differential signals (differential, common mode, P, N), the usual analysis methods are available:

- Trigger on differential, common mode, P, and N waveforms.
  If the differential signal is enabled ("Enable" = on), all possible output waveforms are available as trigger source, even if they are not selected and not displayed.
- Zoom
- Cursor measurements
- Automatic measurements
- Histogram
- Mathematics. In formulas, use DIFF1, DIFF2, COMMON1, COMMON2.
- FFT, spectrogram
- Mask test
- Export of waveform data. You can export the differential and common waveforms if they are selected as output and displayed. Raw data is not available.
  P and N waveforms can be exported if they are selected as output and displayed (source = input channel).

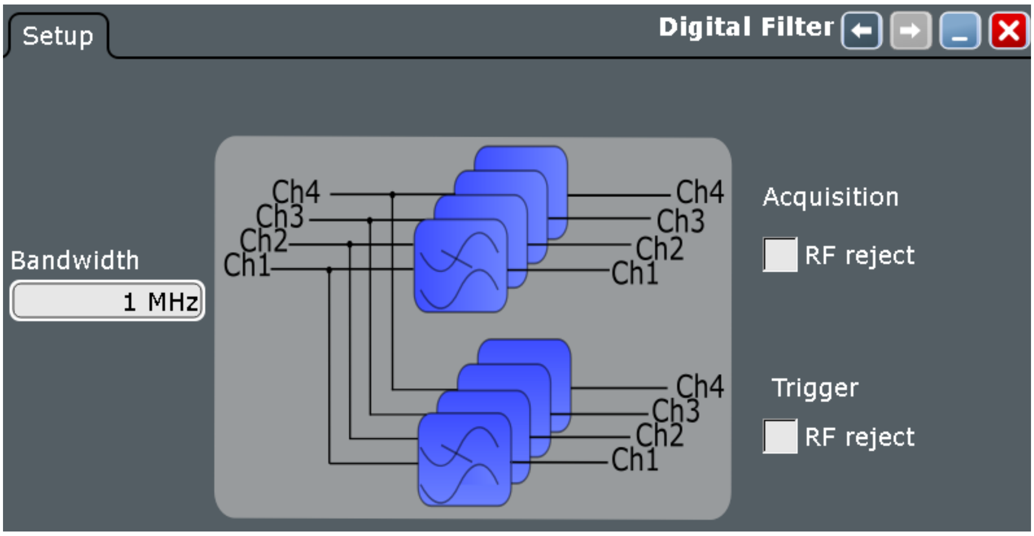
4.8 Digital Filter Setup

After processing by the A/D converter, the channel and trigger signals are digitized signals. These digitized signals can be filtered to reject high frequency - also known as

Digital Signal Processing (DSP). You can filter the acquisition channels as well as the trigger channel signal. For example, RF reject for the trigger signal ensures that triggering will not be caused by unexpected glitches.

If High definition mode is active, digital filter settings are enabled automatically. You can change the high definition bandwidth in the Digital Filter Setup, which is applied to the channels.

Access: "Vertical" menu > "Digital Filter"



Bandwidth
Sets the limit frequency. This limit is applied to the trigger channel and to the acquisition channels that are enabled for filtering.

Remote command:
`TRIGger<m>:RFReject` on page 1123

Acquisition RF reject
Enables the DSP filter for the input channels. Frequencies higher than the "Bandwidth" are rejected, lower frequencies pass the filter.

Remote command:
`CHANnel<m>:DIGFilter:STATe` on page 1123

Trigger RF reject
Enables the DSP filter for the trigger channel. Frequencies higher than the "Bandwidth" are rejected, lower frequencies pass the filter.

Remote command:
`TRIGger<m>:RFSReject` on page 1124

4.9 Horizontal Accuracy

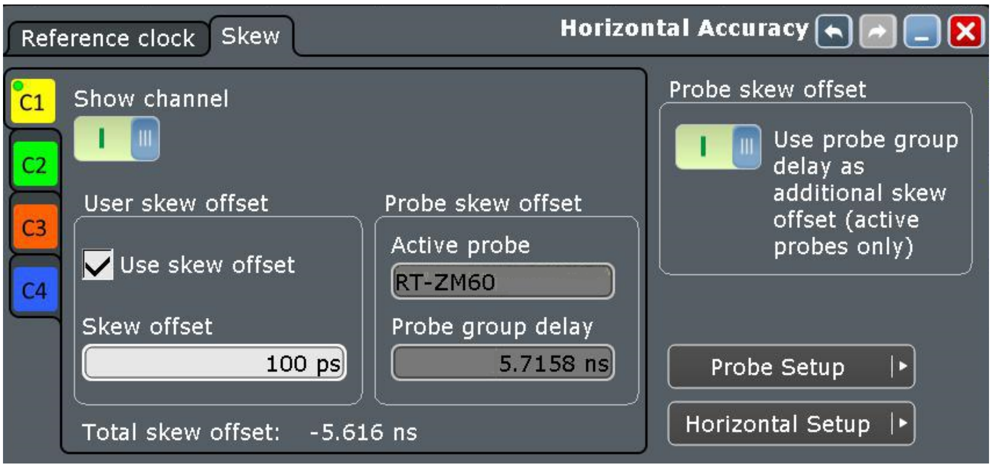
Access: "Horizontal" menu > "Skew".

The "Horizontal Accuracy" dialog box contains standard and optional settings to improve measurement and analysis accuracy and to reduce jitter effects.

4.9.1 Skew

Access: "Horizontal" menu > "Skew".

Skew compensates signal propagation differences between channels caused by the different length of cables, probes, and other sources. Correct skew values are important for accurate triggering and timing relations between channels.



Make sure that the correct channel tab is selected.

Enable / Show channel
Switches the channel signal on or off. The signal icon appears on the signal bar. The waveform of the last acquisition is displayed in the diagram.

Remote command:
CHANnel<m>:STATe on page 1079

Use skew offset
If enabled, the "Skew offset" value is used for compensation.

Remote command:
CHANnel<m>:SKEW:MANual on page 1124

Skew offset
Sets a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically. It affects only the selected input channel.

The delay can be shorter than the sample interval. Thus, the skew improves also horizontal and trigger accuracy.

Remote command:
CHANnel<m>:SKEW:TIME on page 1124

Probe skew offset
Measures the skew of all connected active probes and includes it in the total skew offset.

"Use probe group delay ..."
If enabled, the skew of all connected active probes is measured, displayed, and used for deskewing. The setting affects all active channels.

"Active probe" Shows the type of the probe that is connected to the selected channel.

"Probe group delay" Shows the result of the probe skew measurement on the selected channel.

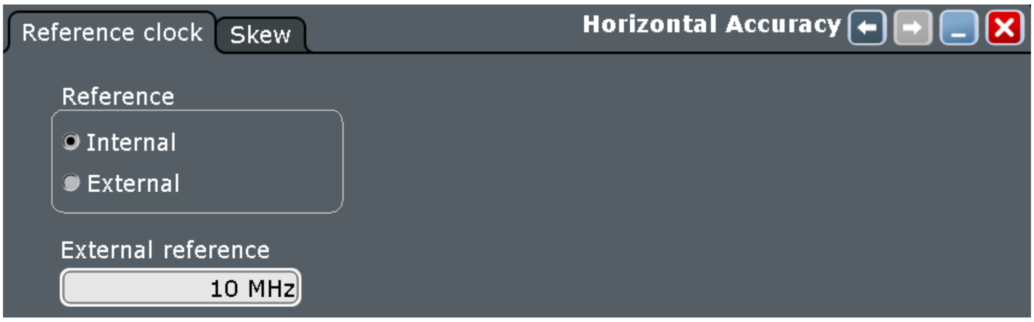
Remote command:
`PROBe<m>:SKEState` on page 1125

Total skew offset
Shows the effective skew offset, the sum of the measured "Probe group delay" and the "Skew offset". If "Use skew offset" is disabled, the skew offset is ignored.

4.9.2 Reference Clock

Access: "Horizontal" menu > "Skew" > "Reference clock" tab.

Input and output reference signals are connected to the [REF IN/OUT] connector on the rear panel of R&S RTE. You can select an internal or external reference clock on the "Reference clock" tab.



Reference
Sets the reference clock that is to be used.

"Internal" Uses the internal reference signal.

"External" Uses an external reference signal.

Remote command:
`SENSe[:ROSCillator]:SOURce` on page 1126

External reference
Sets the frequency of an external reference input signal: 10 MHz.

Remote command:
`SENSe[:ROSCillator]:EXTernal:FREQuency` on page 1126

4.10 Setting Up the Waveform

This chapter contains the fundamental procedures for setting up the acquisition and adjusting the channel waveforms.

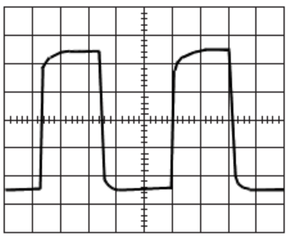
4.10.1 Adjusting Passive Probes

When using a passive probe, you have to compensate it when you connect it to the instrument the first time. Compensation matches the probe cable capacitance to the oscilloscope input capacitance to assure good amplitude accuracy from DC to upper bandwidth limit frequencies. A poorly compensated probe reduces the performance of the probe-oscilloscope system and introduces measurement errors resulting in distorted waveforms and inaccurate results.

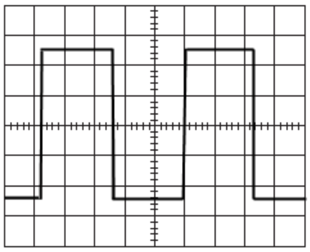
Two connector pins are located on the front panel. The right pin is on ground level. The left pin supplies a square wave signal with 1 kHz for low frequency probe compensation.

1. Connect the BNC connector of the probe to input [CH1].
2. Connect the probe's ground connector to the right compensation pin, and the tip with the left pin.
3. Press [AUTOSET].

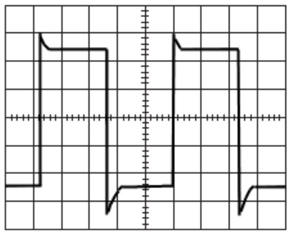
   A square wave appears on the display.
4. Adjust the compensation trimmer of the probe to optimum square wave response. For details, refer to the documentation of your probe.



undercompensated



optimum



overcompensated

4.10.2 Setting Up the Signal Input with Autoset

Autoset is the solution for the major part of routine test-setup. It is also a good start if you need to use more complex trigger settings. Autoset finds appropriate horizontal and vertical scales, vertical offset, and trigger conditions to present a stable waveform.

1. Connect the probe to the input connector [CH ×].

   The instrument recognizes the probe and turns on the channel.

2. Press the [AUTOSET] button on the left of the display.

4.10.3 Adjusting the Signal Input Manually

1. Connect the probe to the input connector [CH ×].
   The instrument recognizes the probe and turns on the channel.
2. On the "Horizontal" menu, tap "Setup".
3. Set the "Time scale".
4. If you want to analyze the signal some time before or after the trigger, use the "Position" and "Reference point" to adjust the visible section of the waveform.
5. Select to set either the resolution or the record length. Enter the required value.
6. Press the channel button corresponding to the input channel. It is illuminated with the color of the channel waveform.
7. In the "Channels" tab, select the "Coupling".
8. Adjust the vertical "Scale", and the vertical "Position".
9. Tap "Acquisition" to proceed with the acquisition setup.

4.10.4 Setting the Acquisition

Prerequisites:
- Probes are connected.
- Vertical and horizontal settings are adjusted.

The settings are described in Chapter 4.2.2, "Acquisition", on page 143.

1. On the "Horizontal" menu, tap "Acquisition".
2. To configure the channel-specific acquisition settings, select the "Channel" subtab.
3. Select the "Mode" - for example, Peak detect or High res.
4. Select the "Wfm Arithmetic" - for example, Average or Envelope.
   The instrument precludes incompatible combinations, like "Peak detect" with "Average".
5. If "Average" is selected for a waveform, enter the "Average count", that is the number of waveforms used for average calculation.

4.10.5 Starting and Stopping Acquisition

You can control the acquisition in two ways:
- Running continuous acquisition until you stop it.

- Running one acquisition or a given number of acquisitions.
  If "Envelope" or "Average" is selected in the "Acquisition" tab, one acquisition means a cycle containing as many acquired waveforms as required to satisfy the reset conditions.

Prerequisites:
- Probes are connected.
- Vertical and horizontal settings are adjusted.
- Triggering is set.
- Channels to be acquired are turned on.

To start and stop continuous acquisition

1. Check if the trigger mode is set to "Normal". The trigger mode is shown in the trigger label in the upper right edge of the screen.
   If not, press the trigger [MODE] key on the front panel to toggle the setting.
2. Press the [RUN CONT] key to start acquisition.
   The acquisition starts if a trigger occurs.
3. To stop , press the [RUN CONT] key again.
   The acquisition stops immediately.

To acquire a limited number of acquisitions

1. Press the TRIGGER key and tap the "Control" tab.
2. In the "Control" area, select the "Normal" trigger mode.
3. Enter the number of acquisitions in the "Average count" field.
4. Press the [RUN N× SINGLE] key on the front panel.
   You can stop the running acquisition before it is finished by pressing the key again.

4.10.6 Using the Roll Mode

The roll mode can be used if the acquisition process is slow - that is if the time scale is large. In roll mode, the instrument shows the waveform immediately and saves waiting for the waveform display. The roll mode can be activated by the instrument if several conditions are fulfilled.

To set the roll mode manually

1. Make sure that all requirements for the roll mode are fulfilled: see "Roll mode" on page 142.
2. Press the [HORIZONTAL] key.
3. In the "Roll mode" section of the "Setup" tab, set "Mode" to "Auto".

4. In the "Min roll mode gain" field, enter the acquisition time at which the instrument starts the roll mode.

4.10.7 Using Fast Segmentation

Fast Segmentation reduces the dead time between two waveform acquisition cycles.

The settings are described in Chapter 4.2.3, "Fast Segmentation", on page 145.

1. On the "Horizontal" menu, tap "Fast Segmentation".
2. Tap "Enable" to activate the Fast Segmentation mode.
3. If you want to sample the maximum number of acquisitions in a series, select "Acquire maximum".
   If you want to capture a defined number of acquisitions, disable "Acquire maximum" and enter the "Required" number of acquisitions.
4. Set the "Replay time", the display time of each acquisition.

4.10.8 Using Digital Filters

Before using digital filters, you determine if you want to filter input channels only or if the trigger signal is filtered too. The filter settings depend on this decision.

For details on filter settings and dependencies, see Chapter 4.8, "Digital Filter Setup", on page 192.

1. On the "Vertical" menu, tap "Digital Filter Setup".
2. To filter the input channels, enable "Acquisition RF reject".
3. To filter the input channels, enable "Trigger RF reject".
4. Set the frequency limit for the filter: "Bandwidth".
   This limit is applied to the trigger channel and to the acquisition channels enabled for filtering.

5 Triggers

5.1 Basics of Triggering

Triggering means to capture the interesting part of the relevant waveforms. Choosing the right trigger type and configuring all trigger settings correctly allows you to detect various incidents in analog, digital, and protocol signals.

Trigger

A trigger occurs if the complete set of trigger conditions is fulfilled. The trigger is the determining point in the waveform record. The instrument acquires continuously and keeps the sample points to fill the pre-trigger part of the waveform record. When the trigger occurs, the instrument continues acquisition until the post-trigger part of the waveform record is filled. Then it stops acquiring and waits for the next trigger. When a trigger is recognized, the instrument does not accept another trigger until the acquisition is complete and the holdoff time has expired.

Trigger setup

A simple trigger setup includes:

- Source of the trigger signal, its coupling and filtering
- Trigger type selection and setup
- Horizontal position of the trigger: see: Chapter 4.1.3.2, "Horizontal Position", on page 134
- Trigger mode

The R&S RTE provides various trigger types for troubleshooting and signal analysis, for example, edge trigger, glitch trigger, interval trigger, pattern trigger, and much more.

For complex tasks like verifying and debugging designs, advanced trigger settings are available:

- Hysteresis, that is the rejection of noise to avoid unwanted trigger events caused by noise
- Holdoff to define exactly which trigger event causes the trigger
- Qualification to consider the states of digital signals on other input channels and their logical combination
- Trigger sequences to combine two trigger type conditions

Action on trigger

A trigger can initiate one or more actions, for example, saving a screenshot or saving waveform data. All available actions can be initiated at the same time.

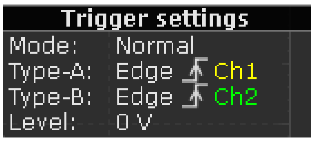
Trigger sequence

A complex trigger sequence joins two or more separate trigger conditions with an optional delay time and an optional reset time or reset condition. Similar setups are also known as multi-step trigger or A/B trigger.

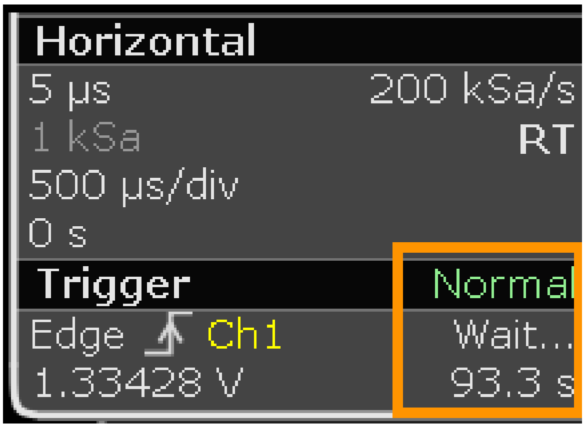
Trigger information

Information on the most important trigger settings is shown in the trigger label on top of the signal bar. If you double-tap the trigger label, the "Trigger" dialog box opens. The label shows:

- Trigger mode
- Trigger type, edge/polarity and trigger source, for A- and B-trigger
- Trigger level



If the trigger mode is normal, and no trigger has been found for longer than one second, the trigger state is shown in the signal bar. The state indicates the waiting for trigger time. For long timebases, it shows the remaining pretrigger time, and after the trigger the time until the acquisition is completed.



5.2 Setting Up the Trigger

This chapter provides step-by-step procedures for the important stages of trigger setup.

5.2.1 Configuring a Simple Trigger

Prerequisites:

- Horizontal and vertical settings are set appropriately to the signals.

- The acquisition is running, the [RUN CONT] key lights green.

For details on settings, see Chapter 5.3, "Trigger Types", on page 203.

Proceed as follows:

1. Press the [TRIGGER] key on the front panel.
   The "Trigger" dialog box opens with the "Setup" tab.
2. At the left hand-side, select the "A" vertical sub tab.
3. Tap the "Source" button and select the trigger source.
4. Tap the "Type" button and select the trigger type.
5. Under "Trigger type dependent settings", configure the settings for the selected trigger type.
   See: Chapter 5.3, "Trigger Types", on page 203
6. To set the trigger level automatically, tap "Find level".
7. Set the normal trigger mode. Do one of the following:
   - Press the [MODE] key on the front panel until "Normal" is shown in the trigger label.
   - Tap the "Normal" trigger mode option in the "Ctrl/Action" tab.

5.2.2 Positioning the Trigger

By positioning the trigger on the time axis, you define which part of the waveform is displayed: mainly the pre-trigger part, or the post-trigger part, or the part around the trigger point.

For details on position settings, see Chapter 4.2.2, "Acquisition", on page 143.

1. Press the [HORIZONTAL] key.
   Alternatively, tap the "Horizontal" menu and then "Setup".
2. Set the "Reference point" and the "Position".
   If you want to set the trigger position outside the waveform display, make sure that "Restrict horizontal position to acquisition range" is disabled.

5.2.3 Using Holdoff

For details on holdoff settings, see Chapter 5.4, "Holdoff", on page 226.

1. Press the [TRIGGER] key and select the "Holdoff" tab.
   Alternatively, tap the "Trigger" menu and then "Holdoff".
2. Select the "Holdoff mode".

3. Enter the "Holdoff settings" belonging to the selected mode.

5.2.4 Setting Up an A → B → R Trigger Sequence

The complete configuration of a complex "A → B → R" trigger sequence consists of:

- A-trigger condition
- B-trigger condition in the same way as for the A-trigger, and optional delay time between the two triggers
- Optional reset by timeout and/or R-trigger

For details on sequence settings, see Chapter 5.7, "Sequence", on page 232.

1. Press the [TRIGGER] key and select the "Setup" tab.
2. Select the type of the "Sequence": "A → B → R".
3. Tap the "A" subtab and configure the first condition.
   See: Chapter 5.2.1, "Configuring a Simple Trigger", on page 201.
4. Select the "B" subtab and configure the B-trigger condition.
5. Optionally, set the "Delay A -> B" that the instrument waits after an A-trigger until it recognizes B-triggers.
6. Set the "B event count". The last B-trigger causes the trigger.
7. You can also define a reset condition. The sequence restarts with the A-trigger if no B-trigger occurs and the reset condition is fulfilled.
   a) Select the "R" subtab.
   b) To specify a reset by timeout, enable "Reset timeout", and enter the time in "Timeout".
   c) To specify a reset trigger type condition, enable "Reset event" and configure the reset trigger type.
      The trigger types and settings depend on the A and B trigger settings. The instrument provides only possible, reasonable combinations.

5.3 Trigger Types

The setup of the trigger type is the most important part of the trigger definition. It determines the method to identify specific signal phenomena. Almost all trigger types are available for all conditions in a trigger sequence, that is, you can combine different types in the sequence. The instrument checks the trigger settings for compatibility and feasibility, and disables settings that do not fit the previous settings in the sequence.

Make sure that the correct trigger tab is selected on the left before you enter the settings.

The settings in the "Setup" tab are:

- Basic Trigger Settings....204
- Edge....206
- Glitch....206
- Width....208
- Runt....209
- Window....210
- Timeout....212
- Interval....213
- Slew Rate....214
- Data2Clock....216
- State....217
- Pattern....218
- Serial Pattern....219
- TV/Video Trigger....221
- Line Trigger....225
- Triggering on Serial Buses....225
- Triggering on Parallel Buses and Digital Channels....226

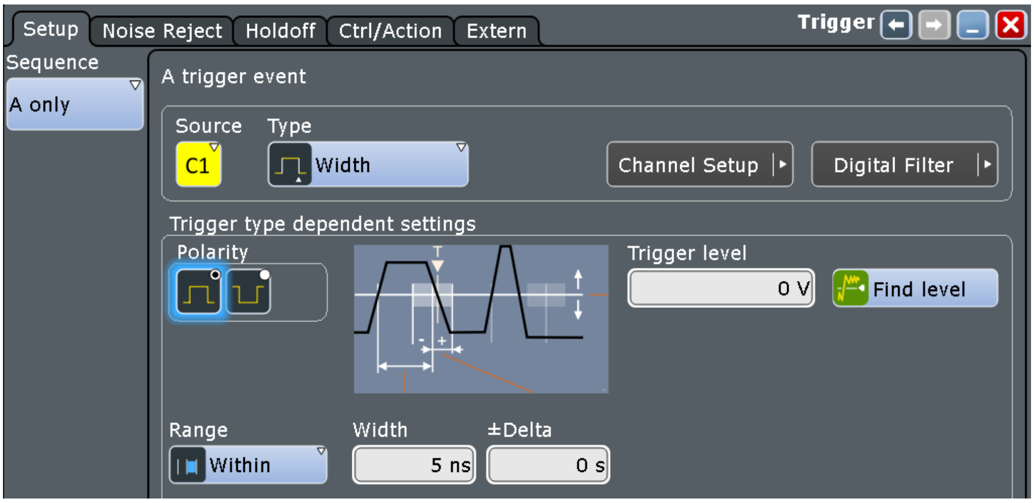
5.3.1 Basic Trigger Settings

Access: [TRIGGER] > "Setup" tab

The basic trigger settings are the trigger source and the trigger type, including the trigger level. These settings are specific for each condition in a trigger sequence. For the trigger source, the current ground/coupling settings are displayed, filtering is also possible.

Depending on the trigger type, additional settings are available. These settings are located under "Trigger type dependent settings".

- Set the trigger level automatically: "Find level".



C1

Source

Selects the source of the trigger signal for the current trigger condition. The trigger source works even if it is not displayed in a diagram. It must be synchronized to the signal to be displayed and analyzed.

The trigger source can be:

- Channel 1...4: An analog input channel
- Extern: External analog signal connected to the external trigger input
  The external trigger source is supported for the "A only" sequence. It is not available if a longer trigger sequence is selected, or if qualification is enabled.
- Line: The instrument generates the trigger from the AC power input and synchronizes the signal to the AC power frequency. Use this source if you want to analyze signals related to the power line frequency, such as lighting equipment and power supply devices.
- Serial bus, D0...D15, Logic, Parallel bus 1...4:
  If options with trigger functionality are installed, the variety of trigger sources of the A-setup is enhanced with specific trigger sources.

Available sources depend on the trigger sequence setting. If "A only" is selected, all inputs (analog input channels, serial and parallel buses, digital channels) can be used as trigger source. If any other trigger sequence is selected, only channel inputs Ch1...4 can be set as trigger source, and all other input sources are disabled. See also: Chapter 5.7, "Sequence", on page 232

Remote command:
`TRIGger<m>:SOURce[:SELect]` on page 1128

Type

Selects the trigger type specific for each condition in a trigger sequence. The current trigger type is shown on the button.

The following trigger types are available:

- Edge, see page 206
- Glitch, see page 206
- Width, see page 208
- Runt, see page 209
- Window, see page 210
- Timeout, see page 212
- Interval, see page 213
- Slew Rate, see page 214
- Data2Clock, see page 216
- State, see page 217
- Pattern, see page 218
- Serial Pattern, see page 219
- TV/Video Trigger, see page 221
- Line Trigger, see page 225

Restrictions:

- If the external trigger input is used as trigger source, the analog edge trigger is the only available trigger type.

Remote command:
`TRIGger<m>:TYPE` on page 1129

Find level
Sets the trigger level automatically to *0.5 * (MaxPeak – MinPeak)*. The function is not available for an external trigger source and the TV trigger.

Remote command:
`TRIGger<m>:FINDlevel` on page 1130

5.3.2 Edge

Access: [TRIGGER] > "Setup" tab > "Type = Edge"

The edge trigger is the most common trigger type. It is well-known from analog oscilloscopes; and you can use it for analog and digital signals.

The trigger condition is fulfilled when the signal from the trigger source passes the specified threshold voltage in the specified direction (slope).

If the trigger source is a channel signal, the edge trigger uses the digitized trigger signal. This signal can be qualified and filtered with the DSP filter. If the trigger source is the external trigger input, the coupling and filter for this signal is set directly in the trigger setup.

Slope
Sets the edge type for the trigger condition.

"Positive"	Selects the rising edge, that is a positive voltage change.
"Negative"	Selects the falling edge, that is a negative voltage change.
"Both"	Selects the rising as well as the falling edge. This option is not available if the trigger source is the external trigger input.

Remote command:
`TRIGger<m>:EDGE:SLOPe` on page 1131
`TRIGger<m>:ANEDge:SLOPe` on page 1133
`TRIGger<m>:SLEW:SLOPe` on page 1144

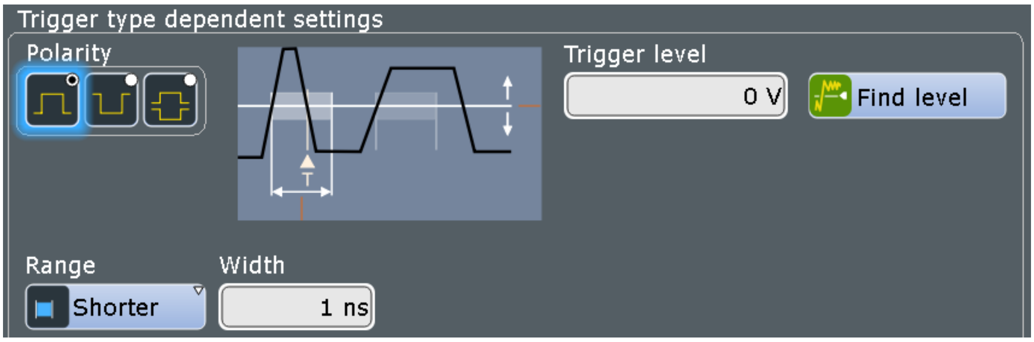
Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display).

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1130

5.3.3 Glitch

Access: [TRIGGER] > "Setup" tab > "Type = Glitch"

The glitch trigger detects pulses shorter or longer than a specified time. It identifies deviation from the nominal data rate and helps to analyze causes of even rare glitches and their effects on other signals.



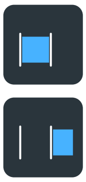
Polarity
Indicates the polarity of a pulse, that is the direction of the first pulse slope.

"Positive" Selects positive going pulses.

"Negative" Selects negative going pulses.

"Either" Selects both positive and negative going pulses.

Remote command:
`TRIGger<m>:GLITch:RANGe` on page 1134
`TRIGger<m>:RUNT:POLarity` on page 1137



Range
Selects which glitches are identified: shorter or longer than the specified "Width".

Remote command:
`TRIGger<m>:GLITch:RANGe` on page 1134

Width
Sets the length of a glitch. The instrument triggers on pulses shorter or longer than this value. The minimum width is 100 ps.

You need to know the expected pulse widths of the circuit to set the glitch width correctly.

Remote command:
`TRIGger<m>:GLITch:WIDTh` on page 1134

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1130

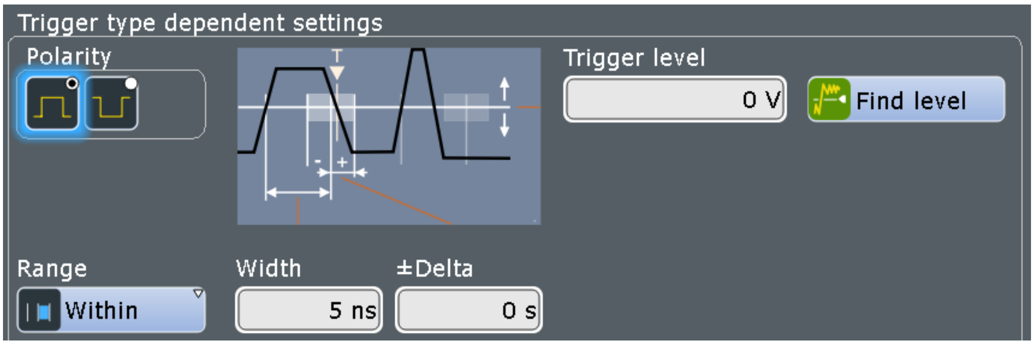
5.3.4 Width

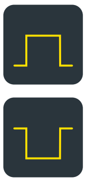
Access: [TRIGGER] > "Setup" tab > "Type = Width"

The width trigger compares the pulse width (duration of a pulse) with a given time limit. It detects pulses with an exact pulse width, pulses shorter or longer than a given time, and pulses inside or outside the allowable time range. The pulse width is measured at the trigger level.

Using the width trigger, you can define the pulse width more precisely than with the glitch trigger. However, using the range settings "Shorter" and "Longer", you can also trigger on glitches.

The width trigger can only analyze **either** positive **or** negative polarity, but searching for a width is also possible for both polarities at the same time ("Either").

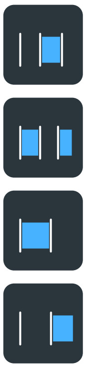




Polarity
Indicates the polarity of a pulse, that is the direction of the first pulse slope.

"Positive"	Triggers on positive going pulses.
"Negative"	Triggers on negative going pulses.

Remote command:
TRIGger<m>:WIDTh:POLarity on page 1135



Range
Selects how the range of a pulse width is defined:

"Within"	Triggers on pulses inside a given range. The range of the pulse width is defined by "±Delta" related to "Width".
"Outside"	Triggers on pulses outside a given range. The range definition is the same as for "Within" range.
"Shorter"	Triggers on pulses shorter than the given "Width".
"Longer"	Triggers on pulses longer than the given "Width".

Remote command:
TRIGger<m>:WIDTh:RANGe on page 1135

Width
For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits ±Delta.

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

Remote command:
`TRIGger<m>:WIDTh:WIDTh` on page 1136

±Delta
Defines a range around the given width value.

The combination "Range" = Within and "±Delta" = 0 triggers on pulses with a pulse width that equals "Width".

The combination "Range" = Outside and "±Delta" = 0 means to trigger on pulse widths ≠ "Width".

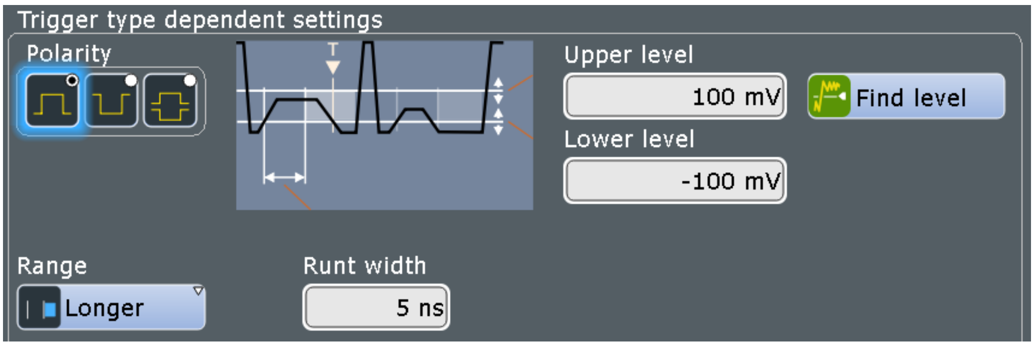
Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1130

5.3.5 Runt

Access: [TRIGGER] > "Setup" tab > "Type = Runt"

A runt is a pulse lower than normal in amplitude. The amplitude crosses the first threshold twice in succession without crossing the second one. In addition to the threshold amplitudes, you can define a time limit for the runt in the same way as for width triggers. For example, this trigger can detect logic, digital, and analog signals remaining below a specified threshold amplitude because I/O ports are in undefined state.




Polarity
Indicates the polarity of a pulse, that is the direction of the first pulse slope.

"Positive"	Selects positive going pulses.
"Negative"	Selects negative going pulses.
"Either"	Selects both positive and negative going pulses.

Remote command:
TRIGger<m>:GLITch:RANGe on page 1134
TRIGger<m>:RUNT:POLarity on page 1137

Upper level
Sets the upper voltage threshold.

Remote command:
TRIGger<m>:LEVel<n>:RUNT:UPPer on page 1137

Lower level
Sets the lower voltage threshold.

Remote command:
TRIGger<m>:LEVel<n>:RUNT:LOWer on page 1137

Range
Selects how the time limit of the runt pulse is defined:

"Any runt"	Triggers on all runts fulfilling the level condition, without time limitation.
"Longer"	Triggers on runts longer than the given "Runt width".
"Shorter"	Triggers on runts shorter than the given "Runt width".
"Within"	Triggers if the runt length is inside a given time range. The range is defined by "Runt width" and "±Delta".
"Outside"	Triggers if the runt length is outside a given time range. The range definition is the same as for "Within" range.

Remote command:
TRIGger<m>:RUNT:RANGe on page 1137

Runt width
For the ranges "Shorter" and "Longer", the runt width defines the maximum and minimum pulse width, respectively.

For the ranges "Within" and "Outside", the runt width defines the center of a range which is defined by "±Delta".

Remote command:
TRIGger<m>:RUNT:WIDTh on page 1138

±Delta
Defines a range around the given runt width.

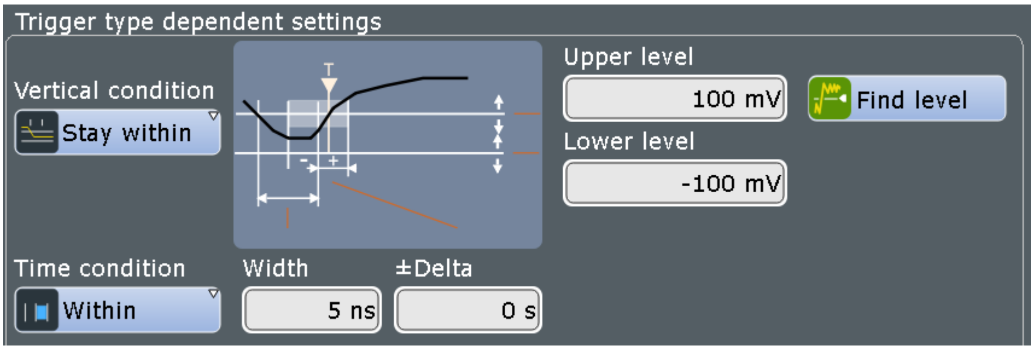
Remote command:
TRIGger<m>:RUNT:DELTa on page 1138

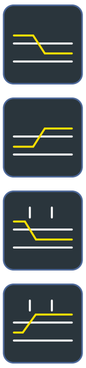
5.3.6 Window

Access: [TRIGGER] > "Setup" tab > "Type = Window"

The window trigger checks the signal run in relation to a "window". The window is formed by the upper and lower voltage levels. The trigger condition is fulfilled, if the waveform enters or leaves the window, or if the waveform stays inside or outside for a time longer or shorter than specified.

With the window trigger, you can display longer transient effects.





Vertical condition
Selects how the signal run is compared with the window:

"Enter"	Triggers when the signal crosses the upper or lower level and thus enters the window made up of these two levels.
"Exit"	Triggers when the signal leaves the window.
"Stay within"	Triggers if the signal stays between the upper and lower level for a specified time. The time is defined in various ways by the Time condition.
"Stay outside"	Triggers if the signal stays above the upper level or below the lower level for a specified time. The time is also defined by the "Time condition".

Remote command:
`TRIGger<m>:WINDow:RANGe` on page 1139

Upper level
Sets the upper voltage limit for the window.

Remote command:
`TRIGger<m>:LEVel<n>:WINDow:UPPer` on page 1139

Lower level
Sets the lower voltage limit for the window.

Remote command:
`TRIGger<m>:LEVel<n>:WINDow:LOWer` on page 1139

Time condition
Selects how the time limit of the window is defined. Time conditioning is available for the vertical conditions "Stay within" and "Stay outside".

"Within"	Triggers if the signal stays inside or outside the vertical window limits at least for the time *Width* - *Delta* and for *Width* + *Delta* at the most.
"Outside"	"Outside" is the opposite definition of "Within". The instrument triggers if the signal stays inside or outside the vertical window limits for a time shorter than *Width* - *Delta* or longer than *Width* + *Delta*.

"Shorter"	Triggers if the signal crosses vertical limits before the specified "Width" time is reached.
"Longer"	Triggers if the signal crosses vertical limits after the specified "Width" time is reached.

Remote command:
`TRIGger<m>:WINDow:TIME` on page 1140

Width
For the ranges "Within" and "Outside", the width defines the center of a time range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", it defines the maximum and minimum time lapse, respectively.

Remote command:
`TRIGger<m>:WINDow:WIDTh` on page 1140

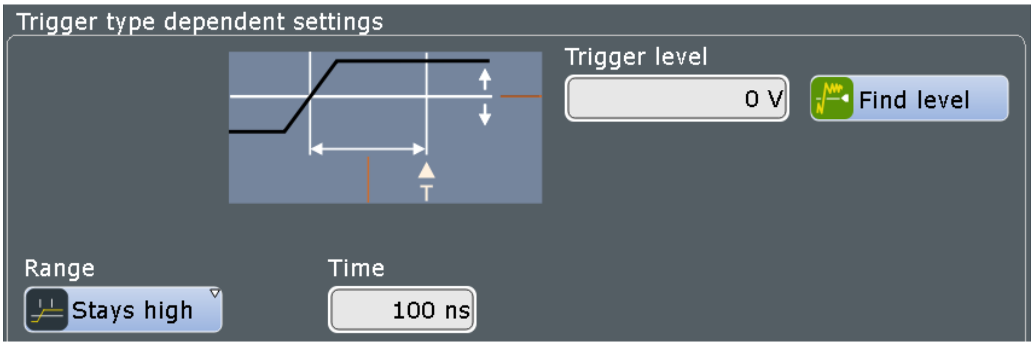
±Delta
Defines a range around the "Width" value.

Remote command:
`TRIGger<m>:WINDow:DELTa` on page 1141

5.3.7 Timeout

Access: [TRIGGER] > "Setup" tab > "Type = Timeout"

The timeout trigger checks if the signal stays above or below the threshold voltage for a specified time lapse. In other words, the trigger occurs if the trigger source does not have the expected transition within the specified time.



Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1130

Range
Selects the relation of the signal level to the trigger level:

"Stays high"	The signal level stays above the trigger level.
"Stays low"	The signal level stays below the trigger level.
"High or low"	The signal level stays above or below the trigger level.

Remote command:
`TRIGger<m>:TIMeout:RANGe` on page 1141

Time
Defines the time limit for the timeout at which the instrument triggers.

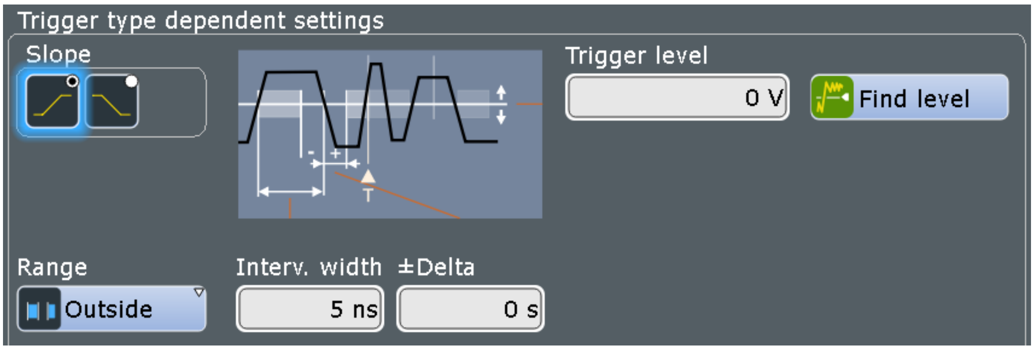
Remote command:
`TRIGger<m>:TIMeout:TIME` on page 1142

5.3.8 Interval

Access: [TRIGGER] > "Setup" tab > "Type = Interval"

The interval trigger analyzes the time between two pulses.

The interval trigger can analyze either rising or falling edges, but searching for an interval is also possible for both edges at the same time ("Either").

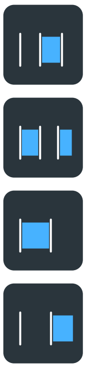


Slope
Sets the edge for the trigger. You can analyze the interval between positive edges or between negative edges.

Remote command:
`TRIGger<m>:INTerval:SLOPe` on page 1142

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1130



Range
Selects how the range of an interval is defined:

"Within"	Triggers on pulse intervals inside a given range. The range is defined by "Interv. width" and "±Delta".
"Outside"	Triggers on intervals outside a given range. The range definition is the same as for "Within" range.
"Shorter"	Triggers on intervals shorter than the given "Interv. width".
"Longer"	Triggers on intervals longer than the given "Interv. width".

Remote command:
`TRIGger<m>:INTerval:RANGe` on page 1142

Interv. width
Defines the time between two pulses.

Remote command:
`TRIGger<m>:INTerval:WIDTh` on page 1143

±Delta
Defines a range around the "Interval width" value.

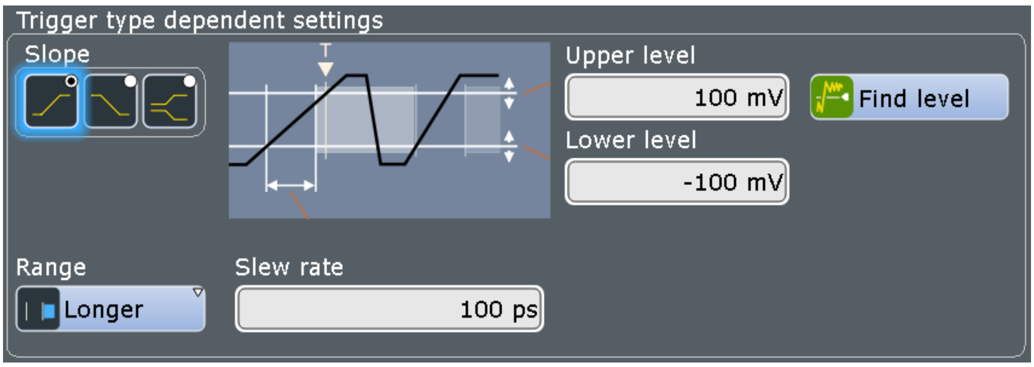
Remote command:
`TRIGger<m>:INTerval:DELTa` on page 1143

5.3.9 Slew Rate

Access: [TRIGGER] > "Setup" tab > "Type = Slew rate"

The slew rate trigger is also known as transition trigger. It triggers if the transition time from the lower to higher voltage level (or vice versa) is shorter or longer as defined, or outside or inside a specified time range.

The slew rate trigger finds slew rates faster than expected or permissible to avoid over-shooting and other interfering effects. It also detects slow edges violating the timing in pulse series.



Slope
Sets the edge type for the trigger condition.

"Positive"	Selects the rising edge, that is a positive voltage change.

"Negative" Selects the falling edge, that is a negative voltage change.

"Both" Selects the rising as well as the falling edge. This option is not available if the trigger source is the external trigger input.

Remote command:
`TRIGger<m>:EDGE:SLOPe` on page 1131
`TRIGger<m>:ANEDge:SLOPe` on page 1133
`TRIGger<m>:SLEW:SLOPe` on page 1144

Upper level

Sets the upper voltage threshold. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope.

Remote command:
`TRIGger<m>:LEVel<n>:SLEW:UPPer` on page 1144

Lower level

Sets the lower voltage threshold. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope.

Remote command:
`TRIGger<m>:LEVel<n>:SLEW:LOWer` on page 1144

Range

Selects how the time limit for the slew rate is defined. The time measurement starts when the signal crosses the first trigger level - the upper or lower level depending on the selected slope. The measurement stops when the signal crosses the second level.

"Within" Triggers on slew rates inside a given time range. The range is defined by "Slew rate" and "±Delta".

"Outside" Triggers on slew rates outside a given time range. The range definition is the same as for "Within" range.

"Shorter" Triggers on slew rates shorter than the given "Slew rate" limit.

"Longer" Triggers on slew rates longer than the given "Slew rate" limit.

Remote command:
`TRIGger<m>:SLEW:RANGe` on page 1145

Slew rate

For the ranges "Within" and "Outside", the slew rate defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the slew rate defines the maximum and minimum slew rate limits, respectively.

Remote command:
`TRIGger<m>:SLEW:RATE` on page 1145

±Delta

Defines a time range around the given slew rate.

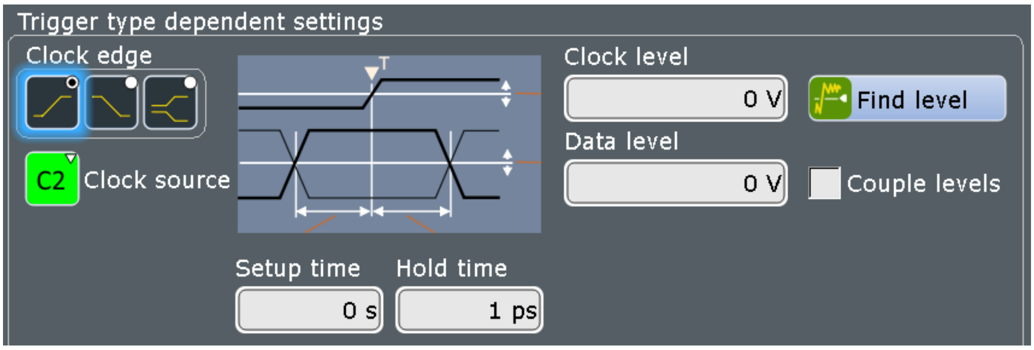
Remote command:
`TRIGger<m>:SLEW:DELTa` on page 1146

5.3.10 Data2Clock

Access: [TRIGGER] > "Setup" tab > "Type = Data2Clock"

With the Data2Clock trigger - also known as setup/hold trigger - you can analyze the relative timing between two signals: a data signal and the synchronous clock signal. Many systems require, that the data signal must be steady for some time before and after the clock edge, for example, the data transmission on parallel interfaces. With this trigger type, you can also test the time correlation of sideband and in-band signals.

The trigger occurs if the data signal crosses the data level during the setup and hold time. The reference point for the time measurement is defined by clock level and clock edge.



Clock source
Selects the input channel of the clock signal.

Remote command:
TRIGger<m>:DATatoclock:CSOurce[:VALue] on page 1146
TRIGger<m>:SPATtern:CSOurce[:VALue] on page 1153

Clock edge
Sets the edge of the clock signal to define the time reference point for the setup and hold time:

"Positive"	Rising edge, a positive voltage change.
"Negative"	Falling edge, a negative voltage change.
"Both"	Both the rising and the falling edge.

Remote command:
TRIGger<m>:DATatoclock:CSOurce:EDGE on page 1146

Clock level
Sets the voltage level for the clock signal. Both "Clock level" and "Clock edge" define the starting point for calculation of the setup and hold time.

Remote command:
TRIGger<m>:DATatoclock:CSOurce:LEVel on page 1147

Data level
Sets the voltage level for the data signal. At this level, the setup and hold time is measured.

Remote command:
TRIGger<m>:LEVel<n>[:VALue] on page 1130

Couple levels (Trigger level and hysteresis coupling)
Sets the trigger levels and hysteresis values for all channels to the values of the currently selected trigger source. The function affects only the levels defined for the selected condition. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.

Remote command:
TRIGger<m>:SCOupling on page 1147

Setup time
Sets the minimum time **before** the clock edge while the data signal must stay steady above or below the data level.

The setup time can be negative. In this case, the hold time is always positive. If you set a negative setup time, the hold time is adjusted by the instrument.

Remote command:
TRIGger<m>:DATatoclock:STIMe on page 1148

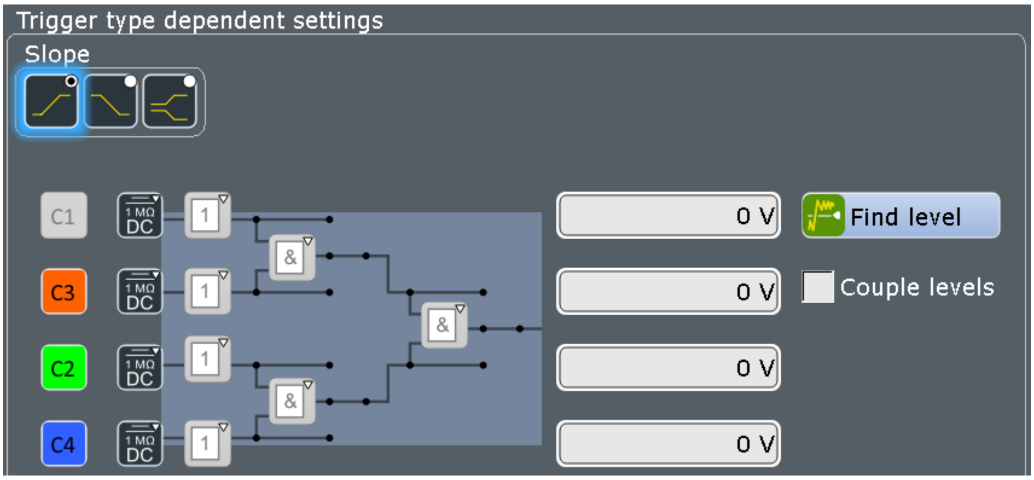
Hold time
Sets the minimum time **after** the clock edge while the data signal must stay steady above or below the data level.

The hold time can be negative. In this case, the setup time is always positive. If you set a negative hold time, the setup time is adjusted by the instrument.

Remote command:
TRIGger<m>:DATatoclock:HTIMe on page 1147

5.3.11 State

The state trigger is a qualified edge trigger. It combines the edge trigger settings with trigger qualification.



The individual settings are:

- "Slope" on page 206
- "Pattern" on page 218
- "Trigger Levels" on page 219
- "Find level" on page 206
- "Couple levels (Trigger level and hysteresis coupling)" on page 217

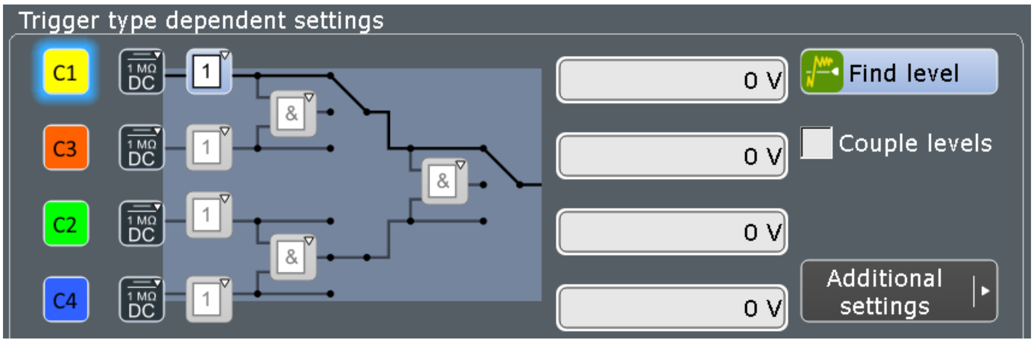
5.3.12 Pattern

Access: [TRIGGER] > "Setup" tab > "Type = Pattern"

The pattern trigger is a logic trigger. It provides logical combination of the input channels and supports you in verifying the operation of digital logic. In addition to the pattern and the trigger levels, you can define a timing condition.

Pattern

The pattern contains the channel selection, and the logical operations structure of hardware-based Boolean logic.



"Channel" — Select the channels to be considered. For qualification, you can select all channel signals except for the trigger source. In pattern trigger setup, the trigger source channel is selected by default, and you can select all other channel signals.

"Coupling" — The current coupling or ground connection is shown for each channel and can be changed directly in the pattern, if necessary.

"Boolean operator" — Defines the logical operation on the digital signal resulting from the comparison with the trigger level.

- "Direct": leaves the input value unchanged
- "NOT": inverts the input value

"Logical operator" — Defines the logic combination of two sources. The sources are channel 1/3 and channel 2/4 on the first step, and in the second step the logical combination resulting from the first step.

- "AND": logical AND, conjunctive combination
- "NAND": logical NOT AND
- "OR": logical OR, disjunctive combination
- "NOR": logical NOT OR

Remote command:
TRIGger<m>:QUALify<n>:A:LOGic on page 1149
TRIGger<m>:QUALify<n>:A[:ENABle] on page 1148
TRIGger<m>:QUALify<n>:AB:LOGic on page 1150
TRIGger<m>:QUALify<n>:ABCD:LOGic on page 1150
TRIGger<m>:QUALify<n>:B:LOGic on page 1149
TRIGger<m>:QUALify<n>:B[:ENABle] on page 1148
TRIGger<m>:QUALify<n>:C:LOGic on page 1149
TRIGger<m>:QUALify<n>:C[:ENABle] on page 1149
TRIGger<m>:QUALify<n>:CD:LOGic on page 1150
TRIGger<m>:QUALify<n>:D:LOGic on page 1149
TRIGger<m>:QUALify<n>:D[:ENABle] on page 1149

Trigger Levels
Define the trigger level for each input channel. For state and pattern trigger, the trigger level is a decision threshold: If the signal value is higher than the trigger level, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the trigger level.

You can set the trigger levels for all channels to the same value, see "Couple levels (Trigger level and hysteresis coupling)" on page 217.

Additional settings: Timing
"Timing" adds time limitation to the pattern condition.

Tap "Additional settings" to open the timing setup.

"Off"	No time limitation. The trigger occurs if the pattern condition is fulfilled.
"Timeout"	Defines how long the result of the pattern condition must be true or false.
"Width"	Defines a time range for keeping up the true result of the pattern condition. The range is defined in the same way as for width and interval triggers, see "Range" on page 208.

Remote command:
TRIGger<m>:PATTern:MODE on page 1151
TRIGger<m>:PATTern:TIMeout:MODE on page 1151
TRIGger<m>:PATTern:TIMeout[:TIME] on page 1152
TRIGger<m>:PATTern:WIDTh:DELTa on page 1153
TRIGger<m>:PATTern:WIDTh:RANGe on page 1152
TRIGger<m>:PATTern:WIDTh[:WIDTh] on page 1152

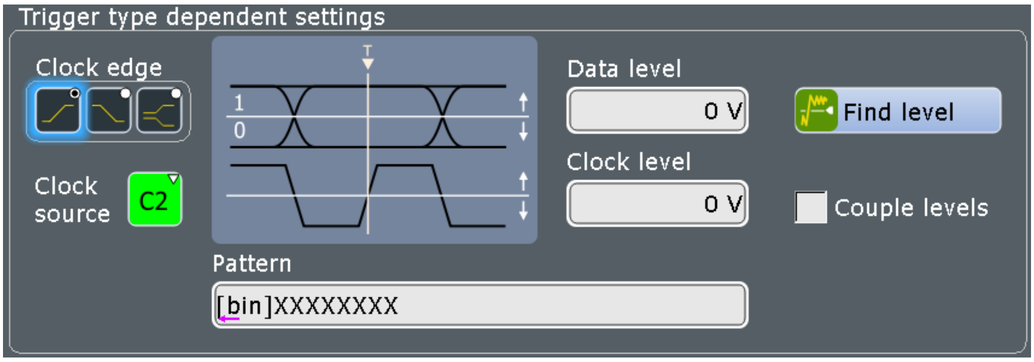
5.3.13 Serial Pattern

Access: [TRIGGER] > "Setup" tab > "Type = Serial Pattern"

The serial pattern is used to trigger on signals with serial data patterns in relation to a clock signal - for example, on bus signals like the I²C bus.

The instrument expects the bits coming in LSB first order. A triggered waveform in the diagram shows the LSB on the left and the MSB on the right side.

For convenient and comprehensive triggering on specific serial data, options for serial protocol analysis are provided.



Clock source
Selects the input channel of the clock signal.

Remote command:
TRIGger<m>:DATatoclock:CSOurce[:VALue] on page 1146
TRIGger<m>:SPATtern:CSOurce[:VALue] on page 1153

Clock edge
Together with the clock level, the clock edge sets the point in time when the state of the data signal is checked:

"Positive"	Rising edge, a positive voltage change.
"Negative"	Falling edge, a negative voltage change.
"Both"	Both the rising and the falling edge.

Remote command:
TRIGger<m>:SPATtern:CSOurce:EDGE on page 1154

Clock level
Sets the voltage level for the clock signal.

Remote command:
TRIGger<m>:SPATtern:CSOurce:LEVel on page 1154

Data level
Sets the voltage level for the data signal.

If the signal value is higher than the data level, the state is 1. Below the level, the signal state is 0.

Remote command:
TRIGger<m>:LEVel<n>[:VALue] on page 1130

Couple levels (Trigger level and hysteresis coupling)
Sets the trigger levels and hysteresis values for all channels to the values of the currently selected trigger source. The function affects only the levels defined for the selected condition. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.

Remote command:
`TRIGger<m>:SCOupling` on page 1147

Pattern
The pattern contains the bits of the serial data to be found in the data stream. The maximum length of the pattern is 128 bit. Touch and hold the "Pattern" field to open the "Bit Pattern Editor" where you can enter the pattern in various formats.

See also: Chapter 12.1.5, "Bit Pattern Editor", on page 481.

In binary format, an *X* indicates that the logical level for the bit is not relevant (do not care).

Remote command:
`TRIGger<m>:SPATtern:PATTern` on page 1154

5.3.14 TV/Video Trigger

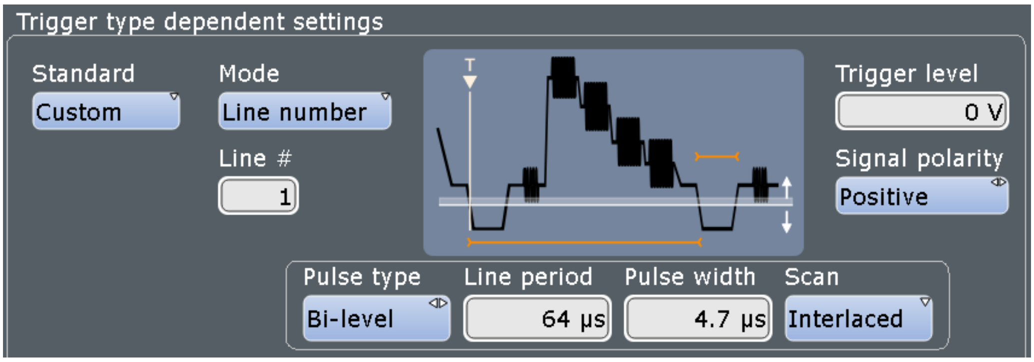
Access: [TRIGGER] > "Setup" tab > "Type = TV"

The TV or video trigger is used to analyze analog baseband video signals. You can trigger on baseband video signals from standard definition and high definition standards, and also on user defined signals.

The instrument triggers on the line start - the horizontal sync pulse. You can trigger on all lines, or specify a line number. You can also trigger on the field or frame start.

Also, a delay can be set: Set the "Holdoff events" in the "Holdoff" tab to the number of fields to be skipped. See also: Chapter 5.4, "Holdoff", on page 226.

Make sure that the trigger level crosses the synchronizing pulses of the video signal, see "Trigger level" on page 223.



Most video signals have an output impedance of 75 Ω. The channel inputs of the R&S RTE have an input impedance of 50 Ω or 1 MΩ. Make sure to provide the adequate matching to ensure amplitude fidelity. A simple 75 Ω feed-through termination combined with 1 MΩ oscilloscope inputs is suitable for most applications.

Once the trigger is set correctly, you can use cursor and automatic measurements to perform amplitude and timing measurements.

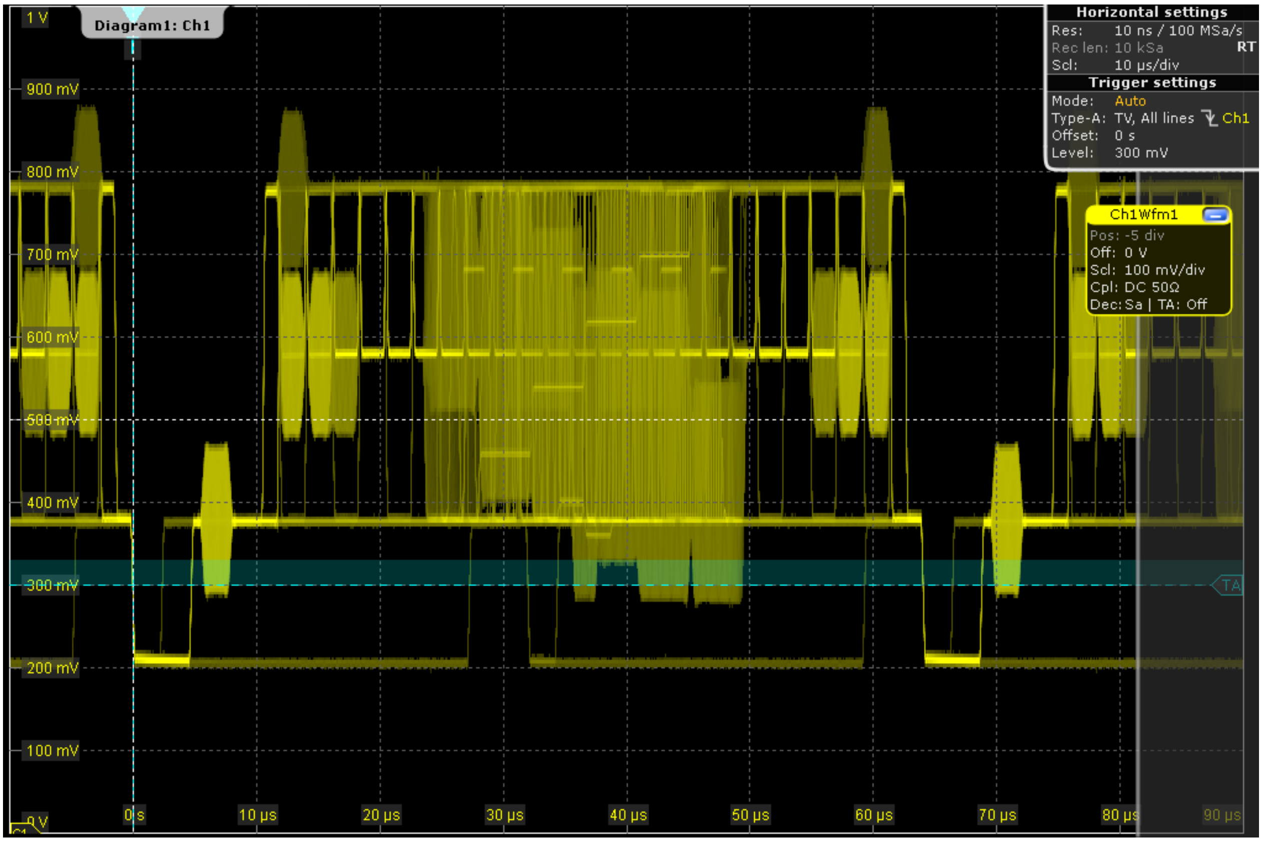


Figure 5-1: Trigger on all lines of a PAL signal with positive signal polarity, trigger level = 300 mV

Standard

Selects the TV standard or "Custom" for user-defined signals.

HDTV standards are indicated by the number of active lines, the scanning system (p for progressive scanning, i for interlaced scanning) and the frame rate. For interlaced scanning, the field rate is used instead of the frame rate. 1080p/24sF is an HDTV standard using progressive segmented frame scanning.

"Custom" can be used for signals of other video systems, for example, medical displays, video monitors, and security cameras. To trigger on these signals, you have to define the pulse type and length of the sync pulse, the scanning system and the line period.

Remote command:
`TRIGger<m>:TV:STANdard` on page 1155

Mode
Selects the lines or fields on which the instrument triggers. Available modes depend on the scanning system of the selected standard.

"All fields"	Triggers on the first video line of the frame (progressive scanning) or field (interlaced scanning), for example, to find amplitude differences between the fields.
"Odd fields / Even fields"	Triggers on the first video line of the odd or even field. These modes are available for interlaced scanning (PAL, PAL-M, SECAM, NTSC, 1080i) and progressive segmented frame scanning (1080p/24sF). They can be used, for example, to analyze the components of a video signal.
"All lines"	Triggers on the line start of all video lines, for example, to find maximum video levels.
"Line number"	Triggers on a specified line. Enter the line number in "Line #".

Remote command:
`TRIGger<m>:TV:MODE` on page 1156

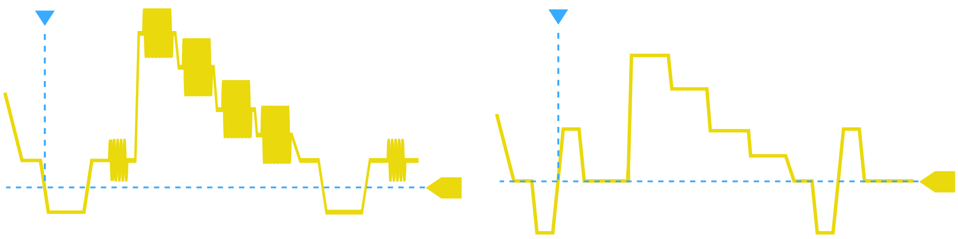
Line #
Sets the number of the line to be triggered on if "Mode" is set to "Line number". Usually the lines of the frame are counted, beginning from the frame start.

For NTSC signals, the lines are counted per field, not per frame. Therefore, you have to set the "Field" (odd or even), and the line number in the field.

Remote command:
`TRIGger<m>:TV:LINE` on page 1157
`TRIGger<m>:TV:LFIeld` on page 1158

Trigger level
Sets the trigger level as threshold for the sync pulse. Make sure that the trigger level crosses the synchronizing pulses of the video signal.

The hysteresis is set according to the settings in the "Noise Reject" tab.



Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1130

Signal polarity
Sets the polarity of the signal. Note that the sync pulse has the opposite polarity, for example, a positive signal has a negative sync pulse.

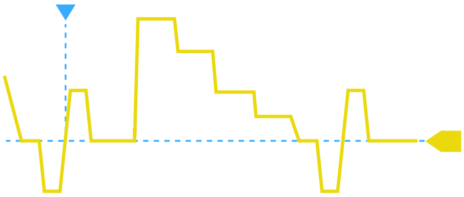


Figure 5-2: Signal with positive polarity and tri-level sync pulse

Remote command:
TRIGger<m>:TV:POLarity on page 1156

Pulse type
Sets the type of the sync pulse, either bi-level sync pulse (used in SDTV signals), or tri-level sync pulse (used in HDTV signals).

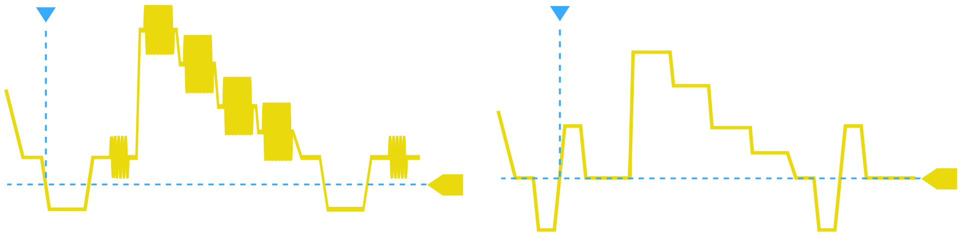
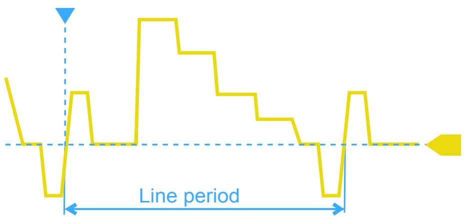


Figure 5-3: Bi-level (left) and tri-level (right) sync pulses

This setting is available for user-defined video signals if "Standard" is set to "Custom".

Remote command:
TRIGger<m>:TV:CUSTom:STYPe on page 1159

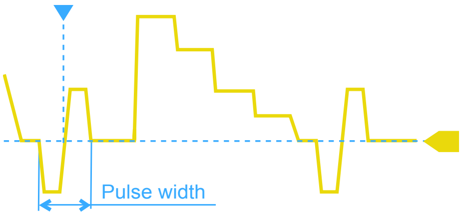
Line period
Sets the duration of a single video line, the time between two successive sync pulses.



This setting is available for user-defined video signals if "Standard" is set to "Custom".

Remote command:
TRIGger<m>:TV:CUSTom:LDURation on page 1158

Pulse width
Sets the width of the sync pulse.



This setting is available for user-defined video signals if "Standard" is set to "Custom".

Remote command:
`TRIGger<m>:TV:CUSTom:SDURation` on page 1159

Scan
Sets the scanning system.

This setting is available for user-defined video signals if "Standard" is set to "Custom".

"Interlaced"	Interlace scanning uses two fields to create a frame. One field contains all the odd lines (odd, first, or upper filed), the other contains all the even lines of the image (even, second, or lower field). First the lines of the odd filed are processed, then the lines of the even field.
"Progressive"	Progressive scanning is a method to capture, transmit and display all lines of a frame in sequence.
"Segmented"	Progressive segmented frame uses progressive scanning to capture the frame, and interlaced scanning for transmission and display.

Remote command:
`TRIGger<m>:TV:CUSTom:SCANmode` on page 1158

5.3.15 Line Trigger

Access: [TRIGGER] > "Setup" tab > "Source" = "Line"

The line trigger triggers on the AC power input and synchronizes the signal to the AC power frequency. It is not a trigger type but rather a special trigger source. Use the line source if you want to analyze signals related to the power line frequency, such as lighting equipment and power supply devices.

Slope
Selects the rising or falling edges of the AC power input for the trigger condition.

Remote command:
`TRIGger<m>:POWerline:SLOPe` on page 1159

5.3.16 Triggering on Serial Buses

Protocol analysis including configuration, triggering, and decoding is described in Chapter 12, "Protocol Analysis", on page 473.

For information on triggering on serial buses, see the "Trigger" chapter of the relevant protocol.

5.3.17 Triggering on Parallel Buses and Digital Channels

Triggering on digital signals requires the Mixed Signal Option. The option is described in Chapter 13, "Mixed Signal Option (MSO, R&S RTE-B1)", on page 867.

For information on triggering, see Chapter 13.3.1, "Trigger Settings for Digital Signals and Parallel Buses", on page 879.

5.4 Holdoff

Access: "Trigger" menu > "Holdoff"

Holdoff conditions define a waiting time after the current trigger until the next trigger can be recognized.

Holdoff mode
Selects the method to define the holdoff condition.

The trigger holdoff defines when the next trigger after the current will be recognized. Thus, it affects the next trigger to occur after the current one. Holdoff helps to obtain stable triggering when the oscilloscope is triggering on undesired events.

Holdoff settings are not available if the trigger source is an external trigger input or serial bus. For the TV trigger, only the "Events" mode is useful.

Example:

You want to analyze the first pulse in a burst of several pulses. At first, you select a sufficiently slow time base to display the entire burst. Then, you set the holdoff time a little longer than the length of the burst. Now, each trigger corresponds to the first pulse in successive bursts, and you can change the time base to display the waveform in more detail.

The following methods are available:

"Time"	Defines the holdoff directly as a time period. The next trigger occurs only after the "Holdoff time" has passed.
"Events"	Defines the holdoff as a number of trigger events. The next trigger only occurs when this number of events is reached. The number of triggers to be skipped is defined in "Holdoff events".
"Random"	Defines the holdoff as a random time limited by "Random minimum time" and "Random maximum time". For each acquisition cycle, the instrument selects a new random holdoff time from the specified range. Random holdoff prevents synchronization to discover effects invisible with synchronized triggering, for example, the features of a pulse train.

"Auto" The holdoff time is calculated automatically based on the current horizontal scale.
"Auto time scaling" defines the factor the horizontal scale is multipied with.
"Auto time" shows the resulting holdoff time: *Auto time* = *Auto time scaling* * *Horizontal scale*.

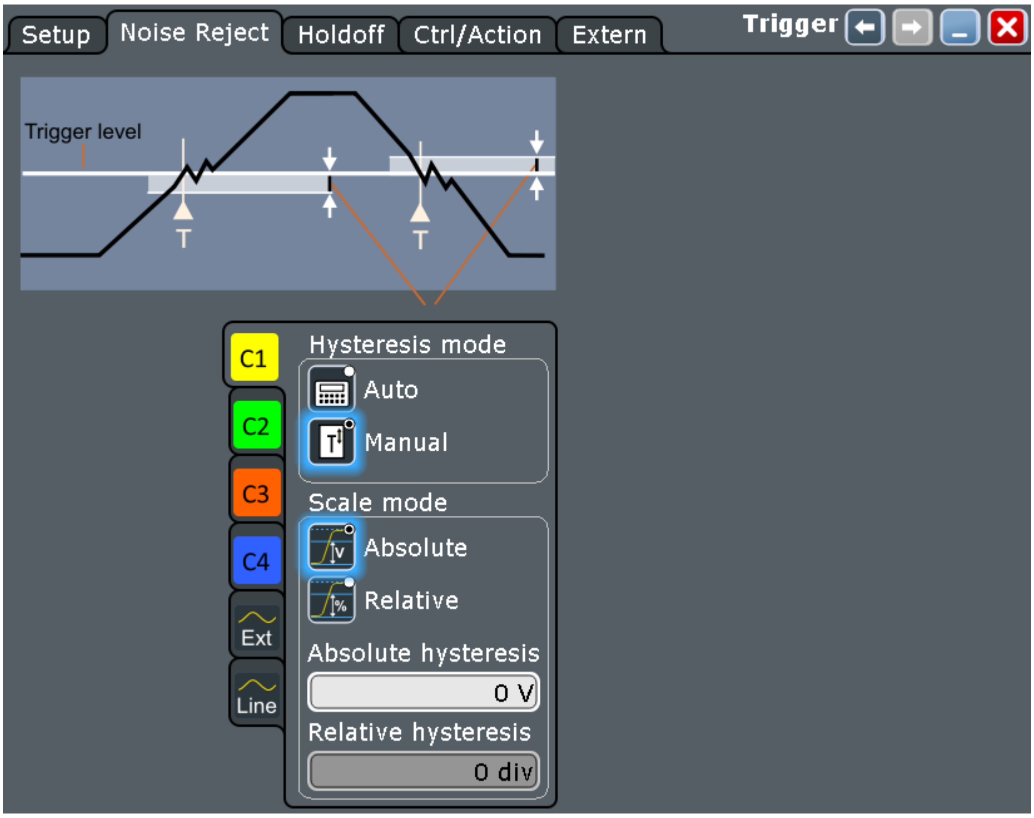
"Off" No holdoff

Remote command:
`TRIGger<m>:HOLDoff:MODE` on page 1160
`TRIGger<m>:HOLDoff:TIME` on page 1161
`TRIGger<m>:HOLDoff:EVENts` on page 1161
`TRIGger<m>:HOLDoff:MAX` on page 1162
`TRIGger<m>:HOLDoff:MIN` on page 1161
`TRIGger<m>:HOLDoff:AUTotime?` on page 1162
`TRIGger<m>:HOLDoff:SCALing` on page 1163

5.5 Noise Reject

The rejection of noise by setting a hysteresis avoids unwanted trigger events caused by noise oscillation around the trigger level.

You can select the hysteresis mode and value for each channel separately, or couple the trigger levels and set the same hysteresis for channels. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.



Hysteresis mode

Selects how the hysteresis is set.

"Auto" This is the recommended mode. The hysteresis is set by the instrument to reject the internal noise of the instrument.

"Manual" The hysteresis is defined directly in absolute or relative values.

Remote command:
TRIGger<m>:LEVel<n>:NOISe[:STATe] on page 1163

Scale mode

Selects whether the hysteresis is defined in absolute or relative values. The setting is available only in manual hysteresis mode.

Remote command:
TRIGger<m>:LEVel<n>:NOISe:MODE on page 1164

Absolute hysteresis

Defines a range in absolute values around the trigger level. If the signal jitters inside this range and crosses the trigger level thereby, no trigger event occurs.

Remote command:
TRIGger<m>:LEVel<n>:NOISe:ABSolute on page 1165

Relative hysteresis

Defines a range in divisions around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Remote command:
TRIGger<m>:LEVel<n>:NOISe:PERDivision on page 1165
TRIGger<m>:LEVel<n>:NOISe:RELative on page 1165

Noise reject (external trigger)
Enables the noise reject for the external trigger input.

Remote command:
TRIGger<m>:ANEDge:NREJect on page 1166

5.6 Control / Action

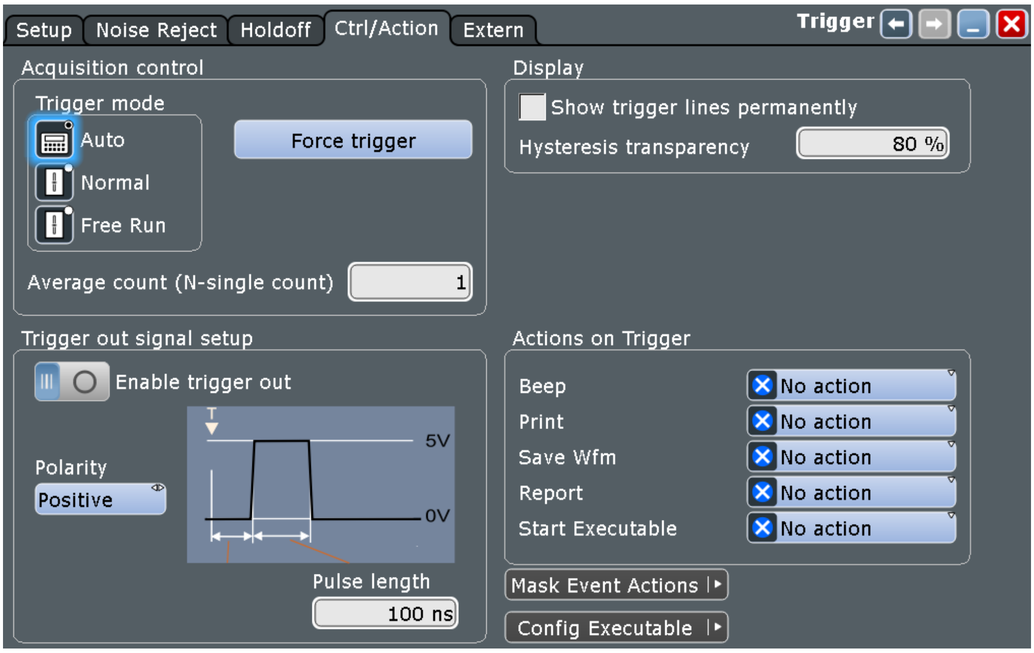
Access: [TRIGGER] > "Ctrl/Action" tab

The settings and functions of trigger control define when the instrument triggers. They affect all trigger types and all triggers in a trigger sequence.

In addition to the settings in the dialog box, you need the [RUN CONT] and [RUN N× SINGLE] keys on the front panel to start and stop the acquisition and thus the triggering.

The action settings define what happens when a trigger occurs. All available actions can be initiated at the same time.

The R&S RTE can provide an external trigger signal to synchronize the measurements of other instruments. The trigger out signal is also adjusted and enabled in the "Control" tab.



Trigger mode
Sets the trigger mode which determines the behavior of the instrument if no trigger occurs. The current setting is shown on the trigger label on top of the signal bar.

To toggle quickly between "Auto" and "Normal" mode, use the [MODE] key on the front panel (in "Trigger" section).

"Auto" The instrument triggers repeatedly after a time interval if the trigger conditions are not fulfilled. If a real trigger occurs, it takes precedence. This mode helps to see the waveform even before the trigger conditions are set correctly. The waveform on the screen is not synchronized, and successive waveforms are not triggered at the same point of the waveform. The time interval depends on the time base settings.

"Normal" The instrument acquires a waveform only if a trigger occurs, that is, if all trigger conditions are fulfilled. If no trigger occurs, no waveform is acquired and the last acquired waveform is displayed. If no waveform was captured before, none is displayed.
When no trigger has been found for longer than one second, a message box appears that shows the time elapsed since the last trigger.

"Free Run" The instrument starts acquisition immediately and triggers after a short time interval independent of the time base settings and faster than in "Auto" mode. Real triggers are ignored. Use this mode if the "Auto" mode is too slow.

Remote command:
`TRIGger<m>:MODE` on page 1169

Average count (N-single count)

Access:
- TRIGGER > "Control" tab > "Average count (N-single count)"
- [ACQUISITION] > "Average count (N-single count)"
- [HORIZONTAL] > "Fast Segmentation" tab > disable "Acquire maximum" > "Required"
- [MATH] > "Setup" tab > "Mode" is not "Off" > "Average count"

The acquisition and average count has several effects:
- It sets the number of waveforms acquired with [RUN N× SINGLE]
- It defines the number of waveforms used to calculate the average waveform. Thus, the instrument acquires sufficient waveforms to calculate the correct average if "Average" is enabled for waveform arithmetic. The higher the value is, the better the noise is reduced.
- It sets the number of acquisitions to be acquired in a fast segmentation acquisition series. Thus, you can acquire exactly one fast segmentation acquisition series with [RUN N× SINGLE].
  If fast segmentation is enabled and configured to acquire the maximum number of acquisitions, the acquisition count is set to that maximum number and cannot be changed. See also: "Number of acquisitions" on page 146.
- It is the "Finished" criteria for the state of a mask test.

Remote command:
`ACQuire:COUNt` on page 1078

Force Trigger
If the acquisition is running in normal mode and no valid trigger occurs, forcing the trigger provokes an immediate single acquisition. Thus you can confirm that a signal is available and use the waveform display to determine how to trigger on it.

If you need this function frequently, you can add the "Force Trigger" icon to the toolbar.

Remote command:
`TRIGger<m>:FORCe` on page 1169

[RUN CONT]/[RUN N× SINGLE]
Front panel keys to start and stop a continuous acquisition or a defined number of acquisition cycles, respectively. The number of acquisitions is set with "Average count".

Remote command:
`RUN` on page 1070
`SINGle` on page 1070
`STOP` on page 1070

Trigger out signal setup
Defines the pulse that is provided to the [TRIGGER OUTPUT] connector on the rear panel.

A trigger out pulse can be provided either when a trigger occurs, or when a mask test violation occurs, or when a limit check violation in a measurement occurs.

"Enable trigger out"	Generates the trigger out signal on trigger event. The setting is not available if: ● A mask test is running with "Trigger Out Pulse" set to "On violation". ● A measurement running with limit check enabled and "Trigger Out Pulse" set to "On violation".
"Polarity"	Sets the polarity of the trigger out pulse, that is the direction of the first pulse edge.
"Pulse length"	Sets the length of the trigger out pulse.
"Delay"	Sets the delay of the first pulse edge to the trigger point. The setting is only available if "Enable trigger out" is active.

Remote command:
`TRIGger<m>:OUT:STATe` on page 1170
`TRIGger<m>:OUT:POLarity` on page 1170
`TRIGger<m>:OUT:PLENgth` on page 1170
`TRIGger<m>:OUT:DELay` on page 1171

Show trigger lines permanently
Displays the trigger levels and the hysteresis in the diagrams until you disable this option.

Remote command:
`DISPlay:TRIGger:LINes` on page 1128

Hysteresis transparency

Defines the transparency of the hysteresis area above or below the trigger level. The hysteresis is only visible if "Show trigger lines permanently" is enabled.

Actions on trigger

The trigger can initiate several actions, each time a trigger occurs. To activate an action, set it to "On trigger". The following actions are available:

"Beep"	Generates a beep sound.
"Print"	Saves a screenshot according to settings in "File" menu > "Print Setup".
"Save Wfm"	Saves the waveform data to a file according to settings in [FILE] > "Waveform / Results" > "Waveforms".
"Report"	Creates and saves a report using the settings defined in "File" menu > "Report Setup".
"Start Executable"	Starts an external application. Tap "Config Executable" to set the application path and parameters. See: Chapter 3.5, "External Application", on page 120.

Remote command:

`TRIGger<m>:EVENt:BEEP` on page 1171

`TRIGger<m>:EVENt:PRINt` on page 1171

`TRIGger<m>:EVENt:WFMSave` on page 1172

`TRIGger<m>:EVENt:RUNexec` on page 1172

5.7 Sequence

A trigger sequence consists of at least one trigger condition and additional conditions defining when the trigger occurs.

A only

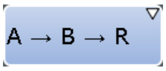
The simple sequence "A only" only contains the A-trigger condition.

A → B

The trigger sequence "A → B" consists of two subsequent conditions: A-trigger and B-trigger, an optional delay and an optional setting to count B-triggers. A- and B-triggers are configured in the same way.

After the A-trigger conditions have been met and an optional delay has passed, the B-trigger with independent conditions is enabled. The instrument waits until one or a specified number of B-trigger conditions occur. The latest B-trigger causes the trigger event, and then the sequence starts again. The B-trigger can only cause the trigger event if it occurs after the A-trigger and after the delay time.

The "A → B" trigger sequence requires that input channels CH1 to CH4 are set as trigger sources for all conditions. All other input sources are disabled.



A → B → R

The trigger sequence "A → B → R" consists of two subsequent conditions: A-trigger and B-trigger with optional B-trigger delay and count. In addition, a reset condition R can be configured: timeout or R-trigger condition. A-, B-, and R-triggers are configured in the same way.

After the A-trigger conditions have been met, and an optional delay has passed, the B-trigger with independent conditions is enabled. The instrument waits until one or a specified number of B-trigger conditions occur. If the reset condition is not fulfilled, the latest B-trigger causes the trigger event, and then the sequence starts again. The B-trigger can only cause the trigger event if it occurs after the A-trigger and after the delay time.

If you expect, for example, an irregular B-trigger, you can configure a reset condition to restart the sequence. The reset condition can be a simple timeout, and/or a reset event that is defined in the same way as the A- and B-trigger conditions.

The "A → B → R" trigger sequence requires that input channels CH1...4 are set as trigger sources for all conditions. All other input sources are disabled. The "A → B → R" sequence is not available if one the following trigger types is set as A-trigger:

- Data2Clock
- TV

The instrument checks all trigger settings for compatibility and disables settings that do not fit the previous settings in the sequence.

See also: Chapter 5.2.4, "Setting Up an A → B → R Trigger Sequence", on page 203.

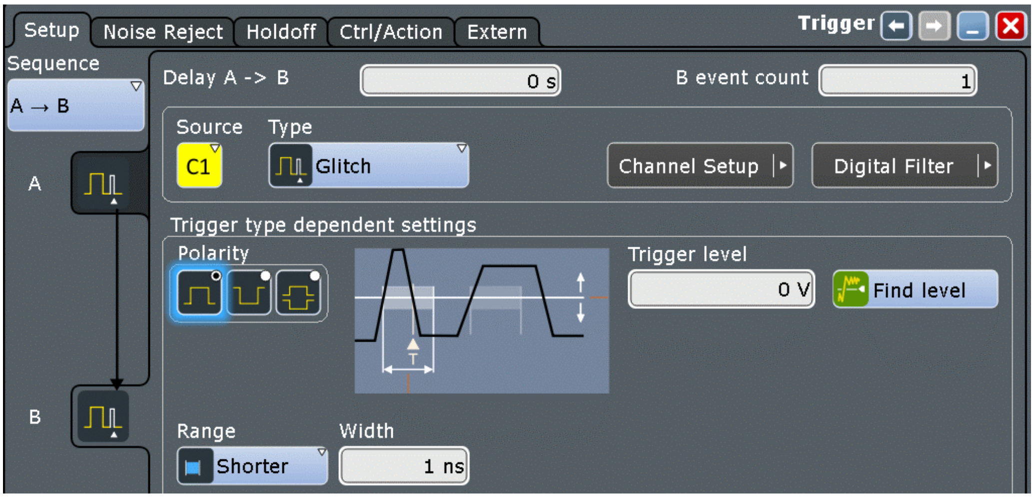
Remote command:

- `TRIGger<m>:SEQuence:MODE` on page 1166

5.7.1 B-Trigger Setup

Access: [TRIGGER] > "Setup" tab > "A → B → R" sequence or "A → B" sequence > "B" subtab

The B-trigger is the second condition of the trigger sequence. You can configure a delay between the A- and B-trigger, and define a number of fulfilled B-trigger conditions to be ignored. The B-trigger condition is configured in the same way as the A-trigger. The instrument disables settings that do not fit the previous settings in the sequence.



Couple trigger levels
Sets the trigger levels to the values of the current trigger condition. Each channel has its own trigger level.

Only available in "A → B → R" sequences.

Example:

If the "A" tab is selected in the "Setup" tab, and the trigger level for C1 is 70 mV, the coupling sets the trigger levels for C1 in the B- and R trigger conditions also to 70 mV. If the B-trigger and/or R-trigger uses another source as the A-trigger, the level remains unchanged.

Remote command:
`TRIGger<m>:ECOupling` on page 1167

Delay
Sets the time that the instrument waits after an A-trigger until it recognizes B-triggers.

Remote command:
`TRIGger<m>:SEQuence:DELay` on page 1167

B-event count
Sets the number of B-trigger conditions to be fulfilled after an A-trigger. The last B-trigger causes the trigger event.

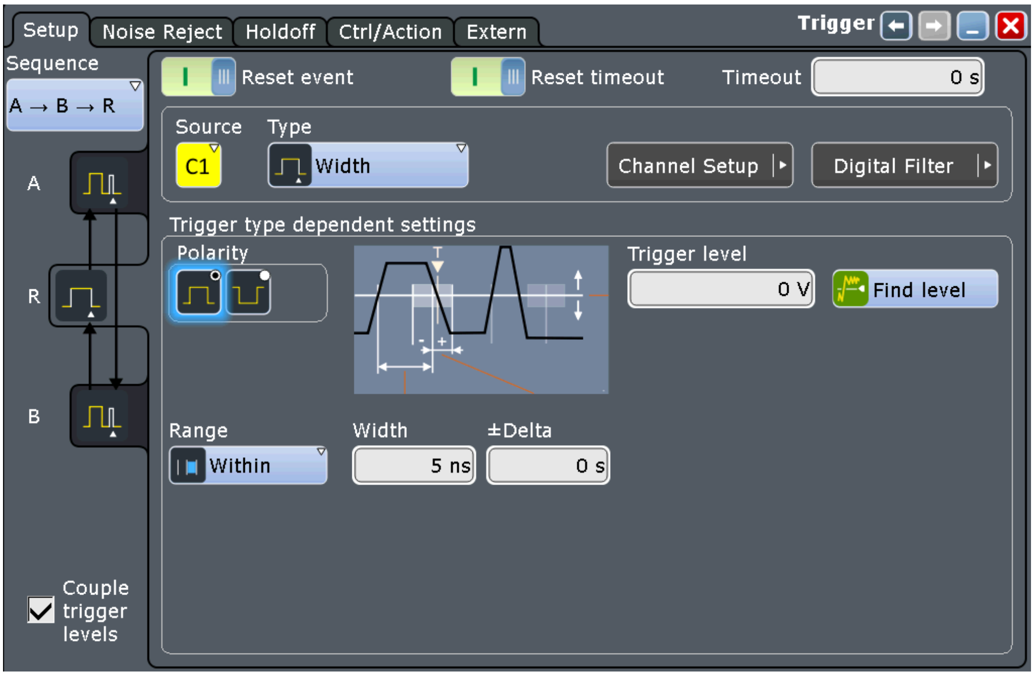
The waiting time for B-triggers can be restricted with a reset condition: timeout or reset event.

Remote command:
`TRIGger<m>:SEQuence:COUNt` on page 1167

5.7.2 R-Trigger Setup

Access: [TRIGGER] > "Setup" tab > "A → B → R" sequence > "R" subtab

The reset condition R can be a timeout or a trigger condition, or a combination of both.



Reset timeout / Timeout

If timeout is enabled, the instrument waits for the "Timeout" time for the specified number of B-triggers. If no trigger occurs during that time, the sequence is restarted with the A-trigger.

Remote command:

TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle] on page 1168

TRIGger<m>:SEQuence:RESet:TIMeout:TIME on page 1168

Reset event

If enabled, the trigger sequence is restarted by the R-trigger condition if the specified number of B-triggers does not occur before the trigger conditions are fulfilled. The R-trigger condition is configured in the same way as the A-trigger. The instrument disables settings that do not fit the previous settings in the sequence.

Remote command:

TRIGger<m>:SEQuence:RESet:EVENt on page 1168

5.8 External Trigger Input

Except for using analog or digital input channels as trigger source, you can also use external signals as trigger source. The external signal is connected to the external trigger input, which is equipped with the Rohde & Schwarz probe interface. Thus, you can use all Rohde & Schwarz probes to connect the external trigger signal. The only trigger type to trigger on external signals is the edge trigger.

1. Connect the external trigger signal to the external trigger input.

2. Set up the probe: "Trigger" menu > "Extern" tab.
   See Chapter 5.8.1, "Probe Setup: Extern Tab", on page 236.
3. Set up the trigger:
   a) Select the "Trigger" > "Setup" tab.
   b) Select the source: "Extern"
   c) Adjust the trigger settings.
      See Chapter 5.8.2, "External Trigger Setup", on page 237.

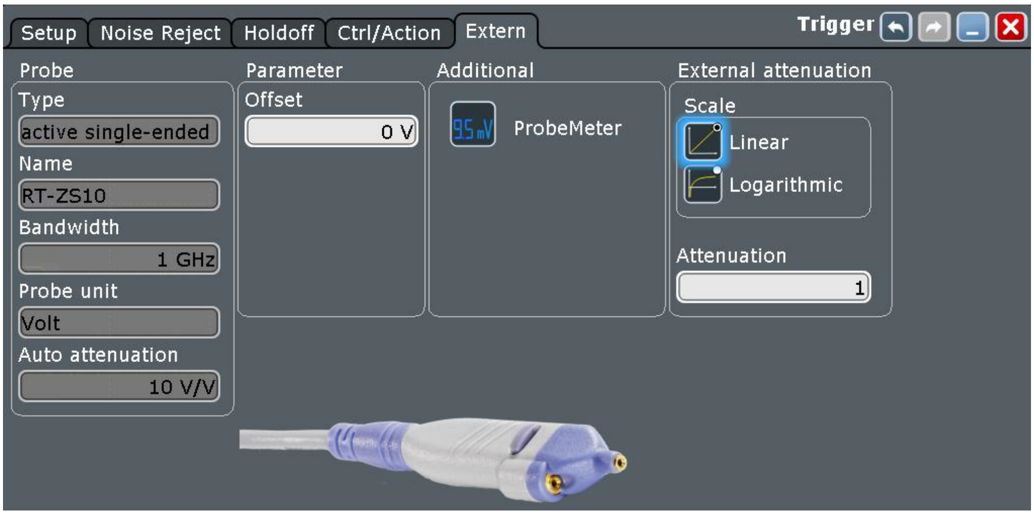
If the trigger source is a channel input, the trigger system uses the digitized signal. The trigger system of the instrument is a separate system, thus the signal processing by enhancement, decimation and arithmetic has no impact on the trigger signal. Most of the R&S RTE trigger types use the digitized trigger signal.

If the trigger source is the external trigger input, the trigger comparator uses the analog input signal. For the external trigger signal, only the edge trigger of the A-trigger is available. Trigger sequence is not supported.

5.8.1 Probe Setup: Extern Tab

The "Trigger" > "Extern" tab provides all settings that are relevant for the probe that is connected to the external trigger input. The functionality on the tab changes according to the type of the attached probe. They are the same as for probe setup of input channels: see Chapter 4.5, "Probes", on page 155.

If the external trigger input is overloaded, a message informs you.



Remote commands:

- TRIGger<m>:EXTern:OVERload on page 1172
- TRPRobe:ID:PARTnumber? on page 1104
- TRPRobe:SETup:STATe? on page 1087
- TRPRobe:SETup:TYPE? on page 1087
- TRPRobe:SETup:ATTenuation:DEFProbe on page 1101

- TRPRobe:SETup:ATTenuation:MANual on page 1094
- TRPRobe:SETup:ATTenuation:MODE on page 1094
- TRPRobe:SETup:ATTenuation:UNIT on page 1094
- TRPRobe:SETup:ATTenuation[:AUTO]? on page 1088
- TRPRobe:SETup:NAME? on page 1087
- TRPRobe:SETup:BANDwidth? on page 1087
- TRPRobe:SETup:CMOFfset on page 1095
- TRPRobe:SETup:GAIN:MANual on page 1095
- TRPRobe:SETup:ZAXV on page 1096

R&S ProbeMeter: remote commands:

- TRPRobe:SETup:DISPlaydiff on page 1091
- TRPRobe:PMETer:VISibility on page 1091
- TRPRobe:PMETer:RESults:COMMon? on page 1092
- TRPRobe:PMETer:RESults:DIFFerential? on page 1092
- TRPRobe:PMETer:RESults:NEGative? on page 1093
- TRPRobe:PMETer:RESults:POSitive? on page 1093
- TRPRobe:PMETer:RESults:SINGle? on page 1092

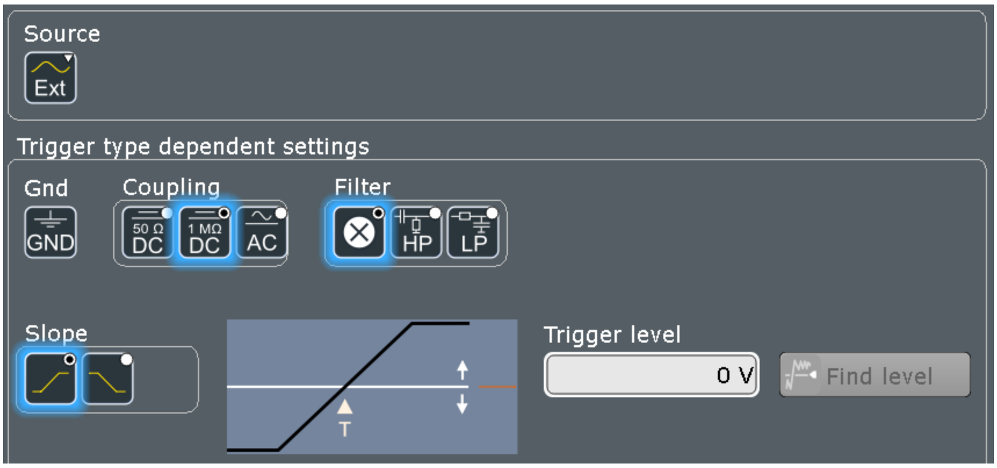
Probe attributes: remote commands

- TRPRobe:ID:PRDate? on page 1104
- TRPRobe:ID:SRNumber? on page 1105
- TRPRobe:ID:SWVersion? on page 1104
- TRPRobe:SETup:CAPacitance? on page 1105
- TRPRobe:SETup:IMPedance? on page 1105

5.8.2 External Trigger Setup

Access: [TRIGGER] > "Setup" tab > "Source = Extern"

External trigger signals, which are connected to the external trigger input, can be triggered with an edge trigger. The "Find level" function is not available for external trigger signals.



Ground

If the selected trigger source is the external trigger input, you can connect the trigger input to the ground.

Remote command:

TRIGger<m>:ANEDge:GND on page 1133

Coupling

You can set the coupling in the trigger configuration.

"DC 50 Ω"	Direct connection with 50 Ω termination, passes both DC and AC components of the trigger signal.
"DC 1 MΩ"	Direct connection with 1 MΩ termination, passes both DC and AC components of the trigger signal.
"AC"	Connection through capacitor, removes unwanted DC and very low-frequency components.

Remote command:

TRIGger<m>:ANEDge:COUPling on page 1131

Filter

If the selected trigger source is "Extern" (external trigger input), you can directly select a filter to reject high or low frequencies.

For all other trigger sources, you can add a digital filter using the Digital Filter Setup.

"Off"	The trigger signal is not filtered.
"Highpass"	Frequencies below the "Cut-off" frequency are rejected, higher frequencies pass the filter. You can adjust the "Cut-off" frequency, the default is 50 kHz.
"Lowpass"	Frequencies higher than the "Cut-off" frequency are rejected, lower frequencies pass the filter. You can adjust the "Cut-off" frequency, the default is 50 kHz.

Remote command:

TRIGger<m>:ANEDge:FILTer on page 1132

TRIGger<m>:ANEDge:CUToff:HIGHpass on page 1132

TRIGger<m>:ANEDge:CUToff:LOWPass on page 1133

Slope

Sets the edge type for the trigger condition.

"Positive"	Selects the rising edge, that is a positive voltage change.
"Negative"	Selects the falling edge, that is a negative voltage change.
"Both"	Selects the rising as well as the falling edge. This option is not available if the trigger source is the external trigger input.

Remote command:

`TRIGger<m>:EDGE:SLOPe` on page 1131

`TRIGger<m>:ANEDge:SLOPe` on page 1133

`TRIGger<m>:SLEW:SLOPe` on page 1144

Trigger level

Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:

`TRIGger<m>:LEVel<n>[:VALue]` on page 1130

5.9 Acquisition Info

Access: "Trigger" menu > "Acquisition Info"

Shows the current number of acquisitions that have been acquired. The count is shown for a running acquisition cycle and as well for the last stopped acquisition cycle.

Remote command:

`ACQuire:CURRent?` on page 1173

6 Waveform Analysis

This chapter describes general methods to check and analyze waveforms. These are:

- Zoom....................240
- Reference Waveforms....................252
- Mathematics....................257
- History....................273
- XY-Diagram....................280

6.1 Zoom

The zoom functions allow you to magnify a specific section of the diagram in order to view more details. You can define several zoom areas for the same diagram and even couple them, or you use the hardware zoom.

6.1.1 Methods of Zooming

The R&S RTE provides various ways of zooming: You define the section of a diagram that you want to magnify, and the zoomed view is shown in a separate zoom diagram. Additionally, you can magnify the diagram directly: The hardware zoom changes the horizontal and vertical scales of the diagram so that you see the selected section.

There are different ways to initiate and configure the zoom function:

- **Fingertip zoom**: magnifies the waveforms around your fingertip. When you drag your finger, the magnifier moves, too. You can convert the fingertip zoom into a standard zoom diagram.
- **Graphical method**: you draw, move and adjust the zoom area on the touchscreen – a very quick and simple method for standard zoom and hardware zoom.
- **Numeric method**: you enter x- and y-values in a dialog box or adjust them using navigation controls. These are precise ways which can be used to optimize a graphically defined zoom.
  With the numeric method there are two ways of defining the zoom area:
  – Specifying **start and stop values** for the x- and y-axes; the acquired data within those values is zoomed.

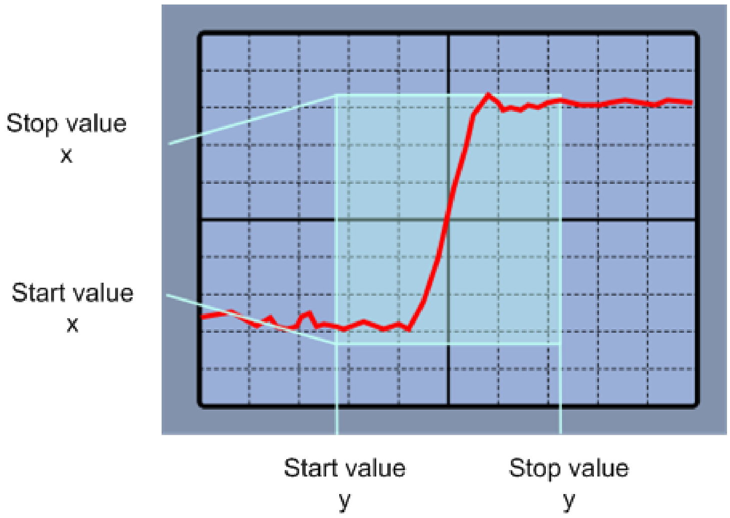


Figure 6-1: Numeric zoom using start and stop values

– Specifying the x- and y-**position** of the centerpoint of the area plus a **range** for the x- and y-axes; the area defined by that centerpoint and the ranges is zoomed.

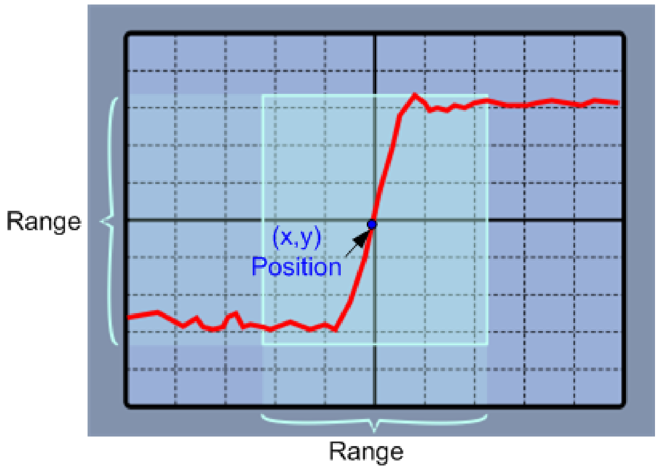


Figure 6-2: Numeric zoom using position and range

- **Coupled zoom** creates a copy of the selected zoom area. Coupled zoom areas always have the same size (size coupling). They can be positioned separately or together (position coupling).

Zoom areas can be used for gating, for example, to define a measurement gate. You can set the gate exactly to the limits of the zoom.

Evaluation gates - available histogram areas, masks, and measurement gates - can be displayed in zoom diagrams to simplify the graphical gate adjustment on the touch-screen. See: "Show evaluation gate(s) in zoom" on page 111.

6.1.2 Zoom Settings

The zoom area, i.e. the section to be enlarged, can be defined using two different methods:

- Using the zoom functions on the toolbar and draw the zoom area on the touch-screen
- Specifying numeric values:
  – start and stop values for the x- and y-axes
  – x and y position of one point in the diagram plus a range for the x- and y-axes

  See also: Chapter 6.1.1, "Methods of Zooming", on page 240.

- Zoom Functions on the Toolbar..242
- Start and Stop Settings..243
- Position and Range Settings..244

6.1.2.1 Zoom Functions on the Toolbar

The zoom icon on the toolbar shows the last selected zoom type. A short tap on the icon activates the selected zoom. If you touch the icon and drag your finger down, a menu opens where you can select another zoom type.

Standard zoom
Displays a magnified section of the diagram in an additional zoom diagram. It is a display zoom, instrument settings are not changed.

Touch and hold the zoom area to open the "Zoom" dialog box.

Remote command:
`LAYout:ZOOM:ADD` on page 1174

Hardware zoom
Changes the instrument settings - horizontal and vertical scales as well as trigger level and offset - to display a part of the diagram in greater detail.

Coupled zoom
Creates a coupled zoom area and its related zoom diagram. If you change the size of one zoom area, the size of all coupled zoom areas is changed as well.

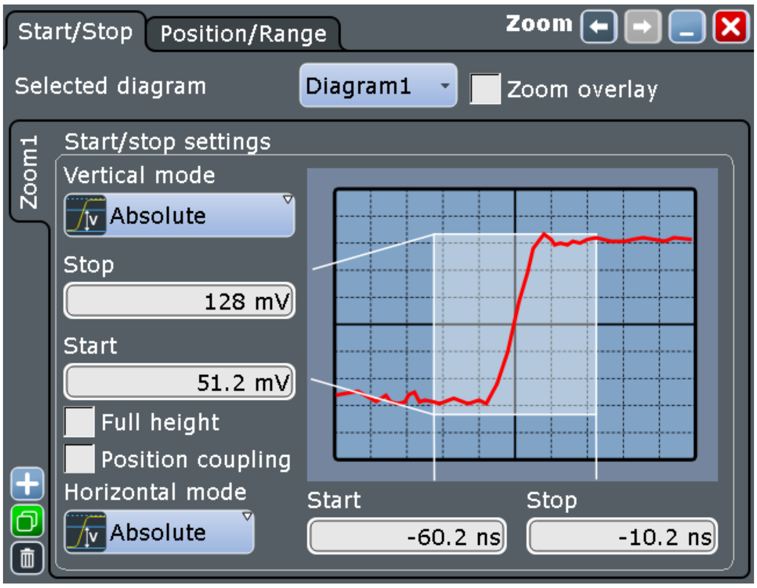
Remote command:
`LAYout:ZOOM:ADDCoupled` on page 1175

Fingertip zoom
Magnifies the waveforms around your fingertip.

Tap the icon and put your finger on the waveform. The touched part of the waveform is displayed in a magnifier. Drag your finger on the screen to move the magnifier. You can change the zoom factor using the Navigation knob.

6.1.2.2 Start and Stop Settings

The "Start/Stop" tab allows you to specify start and stop values for the x- and y-axes. The acquired data within these ranges is zoomed.



Selected diagram
Indicates which of the waveform diagrams is selected for zooming.

Zoom overlay
Shows all zooms of a diagram in one zoom window. The zoomed areas are overlaid for better comparison of the zoomed waveforms.

The setting affects all zoom diagrams.

Remote command:
LAYout:ZOOM:ONEDiagram on page 1175

Vertical
Defines whether absolute or relative values are used to specify the y-axis values.

Remote command:
LAYout:ZOOM:VERTical:MODE on page 1179
SEARch:RESDiagram:VERT:MODE on page 1349

Stop / Relative stop
Defines the upper limit of the zoom area on the y-axis.

Remote command:
LAYout:ZOOM:VERTical:RELative:STOP on page 1182
LAYout:ZOOM:VERTical:ABSolute:STOP on page 1181

Start / Relative start
Defines the lower limit of the zoom area on the y-axis.

Remote command:
LAYout:ZOOM:VERTical:RELative:STARt on page 1182
LAYout:ZOOM:VERTical:ABSolute:STARt on page 1180

Full height
Uses the full diagram height for the zoom area. Only horizontal zoom settings can be changed.

Position coupling
Enables or disables the position coupling of coupled zooms. If position coupling is enabled and you move one zoom area, the other coupled zoom areas are moved, too, and keep their distance.

Remote command:
LAYout:ZOOM:POSCoupling on page 1175

Horizontal
Defines whether absolute or relative values are used to specify the x-axis values.

Remote command:
LAYout:ZOOM:HORZ:MODE on page 1176
SEARch:RESDiagram:HORZ:MODE on page 1348

Start / Relative start
Defines the lower limit of the zoom area on the x-axis.

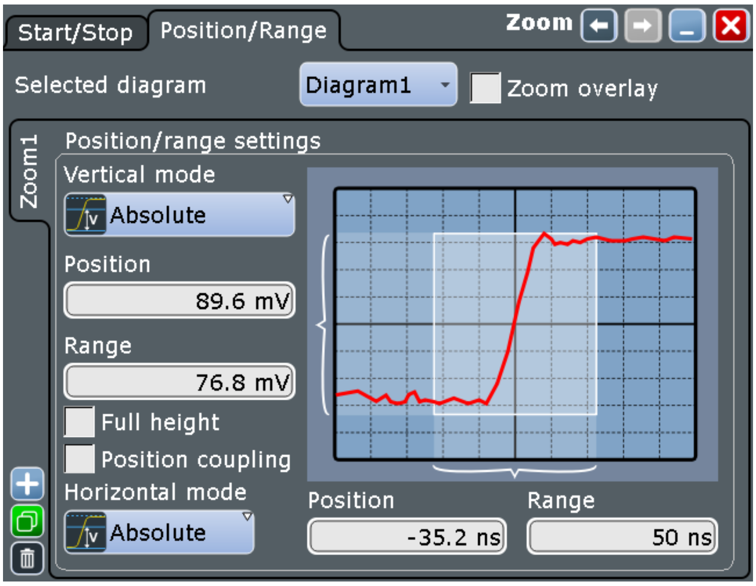
Remote command:
LAYout:ZOOM:HORZ:ABSolute:STARt on page 1177
LAYout:ZOOM:HORZ:RELative:STARt on page 1178

Stop / Relative stop
Defines the upper limit of the zoom area on the x-axis.

Remote command:
LAYout:ZOOM:HORZ:ABSolute:STOP on page 1177
LAYout:ZOOM:HORZ:RELative:STOP on page 1179

6.1.2.3 Position and Range Settings

In the "Position/Range" tab, you specify the x and y position of center point of the zoom area plus a range for the x- and y-axes; the area defined by that point and the ranges is zoomed.



Vertical

Defines whether absolute or relative values are used to specify the y-axis values.

Remote command:
`LAYout:ZOOM:VERTical:MODE` on page 1179
`SEARch:RESDiagram:VERT:MODE` on page 1349

Position / Relative position (vertical)

Defines the y-value of the centerpoint of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:ABSolute:POSition` on page 1179
`LAYout:ZOOM:VERTical:RELative:POSition` on page 1181
`SEARch:RESDiagram:VERT:ABSolute:POSition` on page 1349
`SEARch:RESDiagram:VERT:RELative:POSition` on page 1350

Range / Relative Range (vertical)

Defines the height of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:RELative:SPAN` on page 1181
`LAYout:ZOOM:VERTical:ABSolute:SPAN` on page 1180
`SEARch:RESDiagram:VERT:ABSolute:SPAN` on page 1349
`SEARch:RESDiagram:VERT:RELative:SPAN` on page 1350

Full height

Uses the full diagram height for the zoom area. Only horizontal zoom settings can be changed.

Position coupling

Enables or disables the position coupling of coupled zooms. If position coupling is enabled and you move one zoom area, the other coupled zoom areas are moved, too, and keep their distance.

Remote command:
LAYout:ZOOM:POSCoupling on page 1175

Horizontal
Defines whether absolute or relative values are used to specify the x-axis values.

Remote command:
LAYout:ZOOM:HORZ:MODE on page 1176
SEARch:RESDiagram:HORZ:MODE on page 1348

Position / Relative position (horizontal)
Defines the x-value of the centerpoint of the zoom area.

Remote command:
LAYout:ZOOM:HORZ:ABSolute:POSition on page 1176
LAYout:ZOOM:HORZ:RELative:POSition on page 1178
SEARch:RESDiagram:HORZ:ABSolute:POSition on page 1347
SEARch:RESDiagram:HORZ:RELative:POSition on page 1348

Range / Relative Range (horizontal)
Defines the width of the zoom area.

Remote command:
LAYout:ZOOM:HORZ:ABSolute:SPAN on page 1176
LAYout:ZOOM:HORZ:RELative:SPAN on page 1178
SEARch:RESDiagram:HORZ:ABSolute:SPAN on page 1347
SEARch:RESDiagram:HORZ:RELative:SPAN on page 1348

6.1.3 Zooming for Details

The usage of the various zoom methods is described in the following procedures:

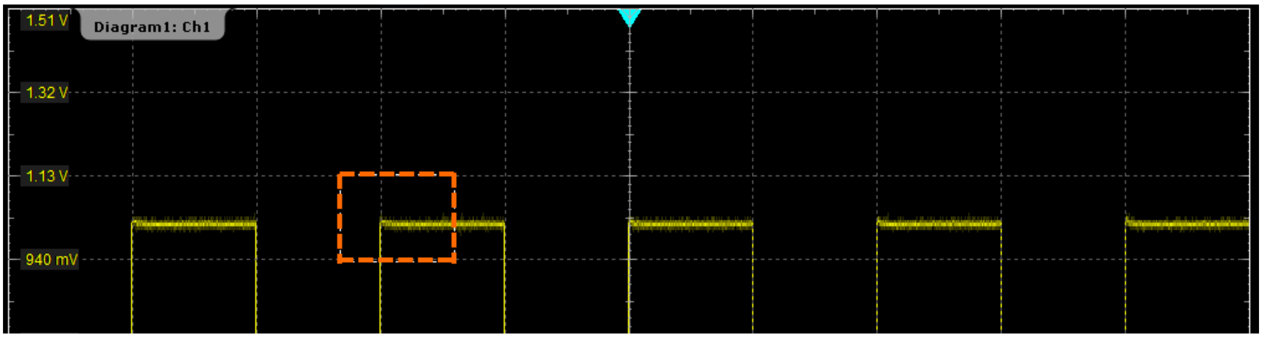
- Chapter 2.3.4.2, "Using the Fingertip Zoom", on page 45
- To define the zoom area graphically on the touchscreen
- To define the zoom area numerically using start-stop values
- To define the zoom area numerically using position and range values
- To define multiple zoom areas
- To define coupled zoom areas
- To close the zoom diagram
- To use the hardware zoom

To define the zoom area graphically on the touchscreen

For graphical zooming, you use your finger on the screen.

1. On the toolbar, tap the "Standard Zoom" icon.

2. Touch the position that you want to define as one corner of the zoom area. Then drag your finger to the opposite corner of the zoom area.
While you drag your finger on the touchscreen, a dotted rectangle indicates the current zoom area. When the rectangle covers the required zoom area, remove your finger.



The indicated area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle.

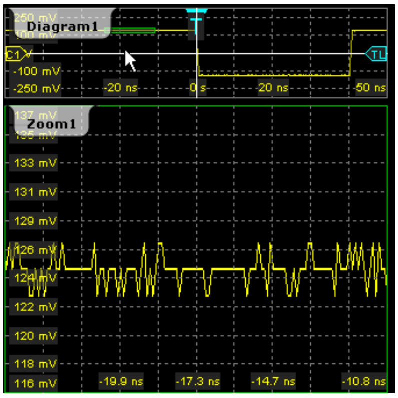


Figure 6-3: Zoom diagram and overview diagram

3. If the position of the zoom area is not correct, drag the rectangle in the overview to the correct position.
4. If the size of the zoom area is not yet ideal, tap the rectangle in the overview diagram.
Now, 4 red lines indicate the edges of the zoom area. A dashed red line indicates the selected edge, which you can adjust.

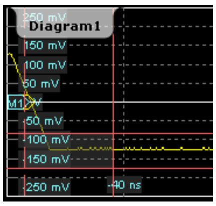


Figure 6-4: Zoom area indicated by edges

Note: Tapping the zoom area toggles between area and edge adjustment.

If the rectangle area is too small to select by tapping, press the CHECKMARK key in the navigation area to toggle between area adjustment and edge adjustment modes.

5. Touch the edge that you want to move, and drag it to the required position.

To optimize the zoom definition of an active zoom diagram, double-tap the zoom diagram. The "Zoom" dialog box for numeric definition is opened.

To adjust the zoom area using navigation controls

If you have created a zoom, and the size and position are not yet ideal, you can adjust them using the navigation knob and the navigation keys. You can adjust the size and position of the zoom area, or adjust the edges of the zoom area individually.

1. To adjust the size and position of the zoom area:
   a) Press the [☑] key until the zoom area is active (grey rectangle with white border).
   b) Turn the navigation knob to shift the zoom area. Press the knob twice to toggle between vertical and horizontal move.
   c) To adjust the size of the zoom area, press the navigation knob until "Span" is shown in the upper left corner.
   d) Turn the knob to increase or decrease the zoom area.

   Note: Pressing the navigation knob toggles between: horizontal position > horizontal span > vertical position > vertical span.

   Tip: If several zoom areas are visible, or cursors are active in addition, the [⇤ Field] and ⇥ Field keys toggle between the zoom areas and the cursor sets.
2. To adjust the edges of the zoom area:
   a) Press the [☑] key until 4 red lines indicate the edges of the zoom area. A dashed red line indicates the selected edge, which you can adjust.
   b) Press the navigation knob until the required edge is selected (dashed red line).
   c) Turn the navigation knob to move the active edge to the required position.
   d) Press the navigation knob again. Adjust the next edge.

   Tip: Pressing the [⇧] or [⇩] keys moves the selected edge to the next division line left (DOWN) or right (UP). In area adjustment mode, these keys move the zoom area one division to the left (DOWN) or right (UP).

To create a new zoom using the Zoom dialog box

1. There are two ways to create a new zoom:
   - If you want to create a new, unconfigured zoom, tap the "Add" icon.
   - If you want to create zoom based on an existing one, tap the "Copy" icon.

     
2. Enter a name for the zoom using the on-screen keyboard.

To define the zoom area numerically using start-stop values

1. On the "Display" menu, tap "Zoom".
2. Select the Start and Stop Settings tab.
3. Under "Vertical mode", select whether you want to define absolute or relative y-axis values. Relative values cause the zoom area to adapt to the input values dynamically.
4. Define the "Start" and "Stop" values that define the lower and upper borders (respectively) of the zoom area on the y-axis (see Figure 6-1).
5. Under "Horizontal mode", select whether you want to define absolute or relative x-axis values.
6. Define the "Start" and "Stop" values that define the lower and upper borders (respectively) of the zoom area on the x-axis.

   When you close the dialog box, the specified area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle (see Figure 6-3).

To define the zoom area numerically using position and range values

1. On the "Display" menu, tap "Zoom".
2. Select the Position and Range Settings tab.
3. Under "Vertical mode", select whether you want to define absolute or relative y-axis values. Relative values cause the zoom area to adapt to the input values dynamically.
4. Under "Position", define the y-value of the center point of the zoom area (see Figure 6-2).
5. Under "Range", define the height of the zoom area.
6. Under "Horizontal mode", select whether you want to define absolute or relative x-axis values.
7. Under "Position", define the x-value of the center point of the zoom area.
8. Under "Range", define the width of the zoom area.

   When you close the dialog box, the specified area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle.

To define multiple zoom areas

You can define more than one zoom area for the same diagram, for example to compare several peaks in a measurement. These zoom areas can be displayed in separate zoom diagrams, or together in one zoom diagram.

To define multiple zoom areas graphically, simply repeat the steps described in To define the zoom area graphically on the touchscreen - for each area. Numerically, proceed as follows:

1. On the "Display" menu, tap "Zoom".
2. Select the required tab according to the method you want to use to define the zoom area.
3. To copy the current zoom area definition, tap the "Copy" icon.
   Alternatively, tap the "Add" icon to add a new zoom area.
4. Enter a name for the new zoom diagram using the displayed on-screen keyboard.
5. Define the zoom area as described for the first zoom.

   An additional zoom diagram is displayed for the new zoom area, and another rectangle in the original diagram indicates the new zoom area. Each rectangle in the overview has the same color as the corresponding zoom diagram frame.
6. In the "Zoom" dialog box, enable "Zoom overlay".

   The zooms are shown in the same zoom diagram, as if the zoom areas are overlaid.

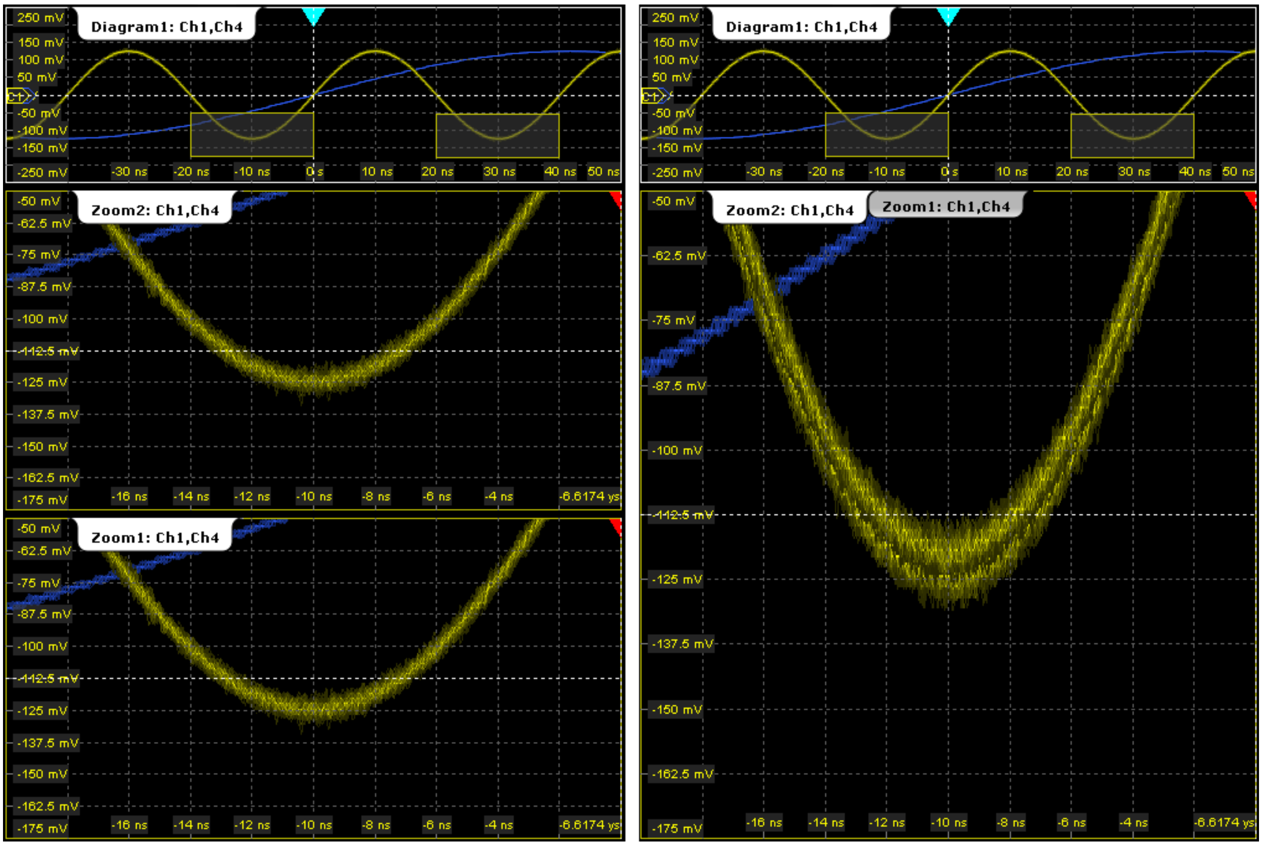


Figure 6-5: Multiple zoom diagrams. Left: separate zoom diagrams. Right: overlaid zoom

To define coupled zoom areas

You can define multiple zoom areas for one diagram that are coupled. If you change the size of one zoom area, the size of all coupled zoom areas is changed as well. Furthermore, you can couple also the position in order to move all coupled zooms at once. Coupling is useful, for example, if you want to compare recurring peaks in a signal.

1. On the toolbar, tap the "Coupled Zoom" icon.

2. In the diagram overview, tap an existing zoom area.

   The selected zoom area is duplicated.

3. Drag the duplicate zoom area to the required position.
4. To create further coupled zooms, repeat the steps above.

   Now, if you change the zoom area size of any of the coupled zoom areas in the "Zoom" dialog box, the settings are changed for all coupled zoom areas.

5. In the "Zoom" dialog box, select the diagram that contains the coupled zooms.
6. Select a zoom tab.
7. Enable "Position coupling".

   If you move one of the coupled zoom areas in the diagram, all other coupled zooms are moved as well, and their distance is kept unchanged.

To close the zoom diagram

1. Tap the "Delete" icon on the toolbar.
2. Tap the zoom diagram.

   The diagram in the overview diagram returns to the original display size.

To use the hardware zoom

In contrast to the normal zoom, the hardware zoom changes the instrument settings - horizontal and vertical scales, and also the trigger level and offset. Thus, the selected area is displayed in the diagram instead of the original waveform. No additional zoom diagram is opened.

1. On the toolbar, tap the "Hardware Zoom" icon.

2. Drag your finger on the touch screen to mark the zoom area.
   A dotted rectangle indicates the current zoom area. When the rectangle covers the required zoom area, remove your finger. The diagram changes and shows the magnified area.

   Tip: To return to the previous display, use the "Undo" icon.

Note: You can combine hardware zoom and normal zoom - first use the hardware zoom, then the zoom into the display. The reverse approach is also possible: Create a zoom diagram, and then apply the hardware zoom to the waveform diagram. Both the waveform and the zoom diagrams are changed.

6.2 Reference Waveforms

You can configure up to four reference waveforms to display stored waveforms. Any active signal or mathematical waveform can be stored as a reference waveform. It can then be loaded again later to restore the waveform on the screen.

6.2.1 Working with Reference Waveforms

Reference waveforms can be displayed in addition to the signal waveforms, saved to file, and loaded back for further analysis. Reference waveforms can be loaded only from BIN files.

Note: Saving and loading reference waveforms, and preset with active reference waveform delete the undo stack. After these actions, undo is not possible.

To update a reference waveform using the toolbar icon

If you often need to update a reference waveform, you can use the "Update Ref Wfm" toolbar icon.

1. Add the "Update Ref Wfm" icon to the toolbar, see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.
2. Touch the icon and open the icon menu.

3. Select the reference waveform to be used.
4. Tap the waveform to be used as reference waveform.

To display a reference waveform

1. In the "Math" menu, select "Reference Waveform > Setup", or press the [REF] key.
2. Select the tab for the reference waveform you want to display ("Ref1"-"Ref4").
3. Load a stored reference waveform as described in "To load a reference waveform" on page 253, or select a source to be displayed as a reference:

a) In the "Reference" tab, tap the "Selected source" icon and select a source from the selection list. The source can be any active signal, math, or other reference waveform.
b) Tap the "Update with" button to update the current reference waveform with the source data.

4. Tap the "Show reference waveform" icon so it is highlighted.

   The reference waveform is displayed on the screen.
   It has the same scaling as the origin.

To save a reference waveform

1. In the "Math" menu, select "Reference Waveform > Setup", or press the [REF] key.

   Tip: You can also save a waveform as a reference waveform in the "File" dialog box, see Chapter 11.2.7, "Saving and Loading Waveform Data", on page 457. Here, you can also save multiple waveforms in one file.

2. Select the tab for the reference waveform you want to store ("Ref1"-"Ref4").
3. Display and configure the reference waveform as described in "To display a reference waveform" on page 252.
4. Select the file format.

   Note: Reference waveforms can be loaded only from BIN files. XML and CSV formats are meant for further processing in other applications.

5. To save the waveform to the currently selected file, tap "Save". By default, the prefix for reference waveform files is "RefCurve".
   To save the waveform to another file, select "Save As".
   Enter a file name and select the directory. The file type is already defined according to the selection in the previous step. In order to load the reference waveform on the instrument again later, use the file type BIN.

   The source settings of the reference waveform and the current scaling settings are stored to the specified file.

To load a reference waveform

Note: Reference waveforms can be loaded only from BIN files.

1. In the "Math" menu, select "Reference Waveform > Setup", or press the [REF] key.
2. Select the tab for the reference waveform you want to load ("Ref1" - "Ref4").
3. To load the waveform from the specified file, tap "Load".
   To load the waveform from a different file, tap "Open". Select the file from the file selection dialog box. Only BIN files are displayed in the file list.

   The selected waveform is loaded as the specified reference waveform.
   If multiple waveforms are saved in the file, you are asked to assign each waveform to a reference waveform. All waveforms are loaded together.

4. If the reference waveform is not visible, tap the "Show reference waveform" icon.

To view a reference waveform

► You can view a reference waveform, using the "Graphical Recall" function.
For details, see: Chapter 11.1.3, "Graphical Recall Function", on page 435.

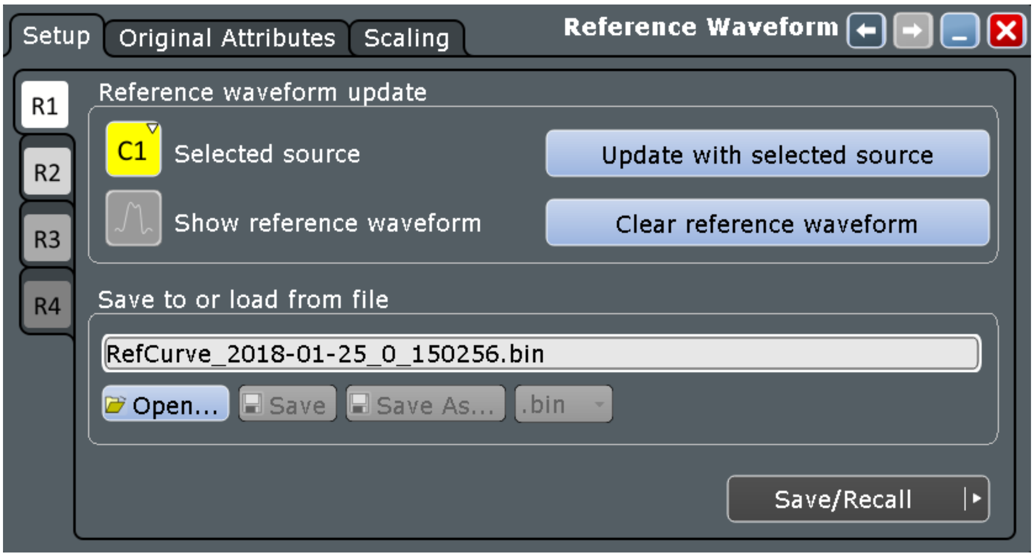
6.2.2 Settings for Reference Waveforms

To compare waveforms and analyze differences between waveforms, you can use up to four reference waveforms R1 to R4. Each reference waveform has its own memory on the instrument. You can also save an unlimited number of reference waveforms and load them for further use.

6.2.2.1 Reference Waveform Setup

Access: [REF] key

In the "Setup" tab, you select the reference waveform and its source. The source can be an active waveform - trace of an input channel, math waveform or another reference waveform - or a stored waveform.



Ref 1/2/3/4 254
Source 254
Update with selected source 255
Show reference waveform 255
Clear reference waveform 255
Save to or load from file 255

Ref 1/2/3/4
Each tab contains the settings for one of the four available reference waveforms.

Source
Selects the source waveform from the active waveforms of input channels, math signals and other reference waveforms.

Remote command:
REFCurve<m>:SOURce on page 1183

Update with selected source
Copies the selected source waveform with all its settings to the memory of the reference waveform. If the acquisition is running, the reference waveform is a snapshot.

Remote command:
REFCurve<m>:UPDate on page 1184

Show reference waveform
Displays the reference waveform in the diagram.

Remote command:
REFCurve<m>:STATe on page 1183

Clear reference waveform
The selected reference waveform disappears, its memory is deleted.

Remote command:
REFCurve<m>:CLEar on page 1185

Save to or load from file
Enter the filename of the stored reference waveform and select the file format with the format button on the right. Double-tap the filename to open the file selection dialog box, see also Chapter 11.7, "File Selection Dialog", on page 471.

By default, the filename has the prefix "RefCurves_". You can define a pattern for automatic naming in the "Autonaming" tab.

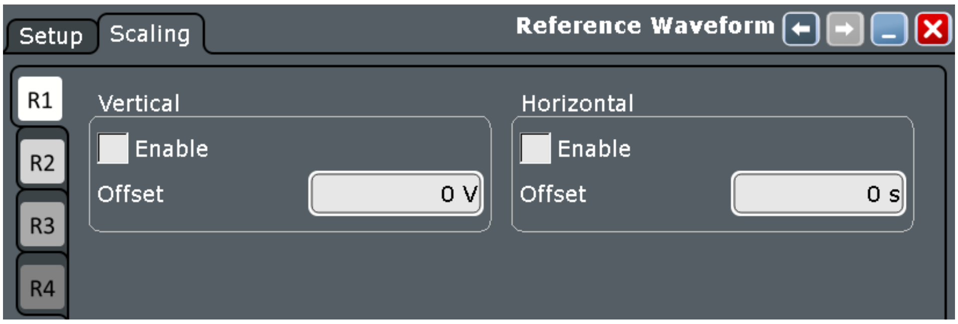
Note: Note that reference waveforms can be loaded from .bin files only. xml and csv formats are meant for further processing in other applications.

"Load"	Loads the specified reference waveform.
"Open"	Opens a file selection dialog box and loads the selected reference waveform file.
"Save"	Saves the waveform as a reference waveform in the selected file.
"Save As..."	Opens the file selection dialog box and saves the waveform to the selected file.
".bin/.xml/.csv"	Selects the file format.

Remote command:
REFCurve<m>:OPEN on page 1184
REFCurve<m>:SAVE on page 1184
REFCurve<m>:DELete on page 1184

6.2.2.2 Scaling

A reference waveform can have its own vertical and horizontal offset.



Vertical

Vertical reference scaling changes the vertical display of the reference waveform independently of the settings of the source waveform.

Enable ← Vertical

If enabled, the vertical offset is applied to the reference waveform.

Remote command:

`REFCurve<m>:RESCale:VERTical:STATe` on page 1185

Offset ← Vertical

Moves the reference waveform vertically. Enter a value with the unit of the waveform. Like vertical offset of a channel waveform, the offset of a reference waveform is subtracted from the measured value. Negative values shift the waveform up, positive values shift it down.

Note: As for all waveforms, a vertical offset of a reference waveform can be set using the vertical [POSITION / REF POINT] knob. This offset is independent from the reference scaling offset, which is described here. If both offsets are set, their values are added up.

Remote command:

`REFCurve<m>:RESCale:VERTical:OFFSet` on page 1185

Horizontal

Horizontal reference scaling changes the horizontal display of the reference waveform independently of the settings of the source waveform and of the diagram settings.

Enable ← Horizontal

If enabled, the horizontal offset is applied to the reference waveform.

Remote command:

`REFCurve<m>:RESCale:HORizontal:STATe` on page 1186

Offset ← Horizontal

Moves the waveform horizontally. Enter a value with a time unit suitable for the time scale of the diagram. Positive values shift the waveform to the right, negative values shift it to the left.

Remote command:

`REFCurve<m>:RESCale:HORizontal:OFFSet` on page 1186

6.3 Mathematics

Math waveforms are calculated waveforms. You can define up to eight math waveforms and display them on the screen, and use it as source for further analysis.

Math waveforms are defined by mathematical expressions (formulas). You can enter mathematical expressions using different methods:

- "Basic": you define a simple math function in a graphical editor by selecting the source waveforms and the operator.
- "Advanced": you define sophisticated math functions in a formula editor, as required to your needs.

The result of an FFT analysis is a specific math waveform. For information on FFT and spectrograms, see Chapter 8, "Spectrum Analysis", on page 359.

The vertical scale of a math waveform is adapted automatically to the measurement results to ensure optimal display. Furthermore, you can scale each math waveform manually in vertical direction like a channel waveform.

As for channel waveforms, you can also change the arithmetic mode for the waveform to display the envelope or an average over several calculations.

You can store a math waveform as a reference waveform and restore it later, see "To save a reference waveform" on page 253.

- Displaying Math Waveforms....257
- Math Setup - General Settings....258
- Basic Editor....261
- Advanced Expressions....263
- Filters....271

6.3.1 Displaying Math Waveforms

Math waveforms can be displayed in addition to the channel and other waveforms. They also can be used for analysis, e.g. measurements, even if the math waveform is not active.

1. In the "Math" menu, select "Math Setup".
   Alternatively, press the [MATH] key.
2. Define the math expression for calculation in one of the following ways:
   - Chapter 6.3.3.2, "Defining a Formula in the Basic Editor", on page 262
   - Chapter 6.3.4, "Advanced Expressions", on page 263
   - Chapter 8.1.2, "Configuring Spectrum Waveforms", on page 363
3. In the "Math Setup" dialog box, in the "Setup" tab, tap the "Enable math signal".
   The math waveform is displayed on the screen.
4. To change the vertical scaling of the math waveform, tap the "Manual" icon.

5. Enter the "Vertical scale" factor (per division). If necessary, add a "Vertical offset". By default the instrument performs an automatic scaling.

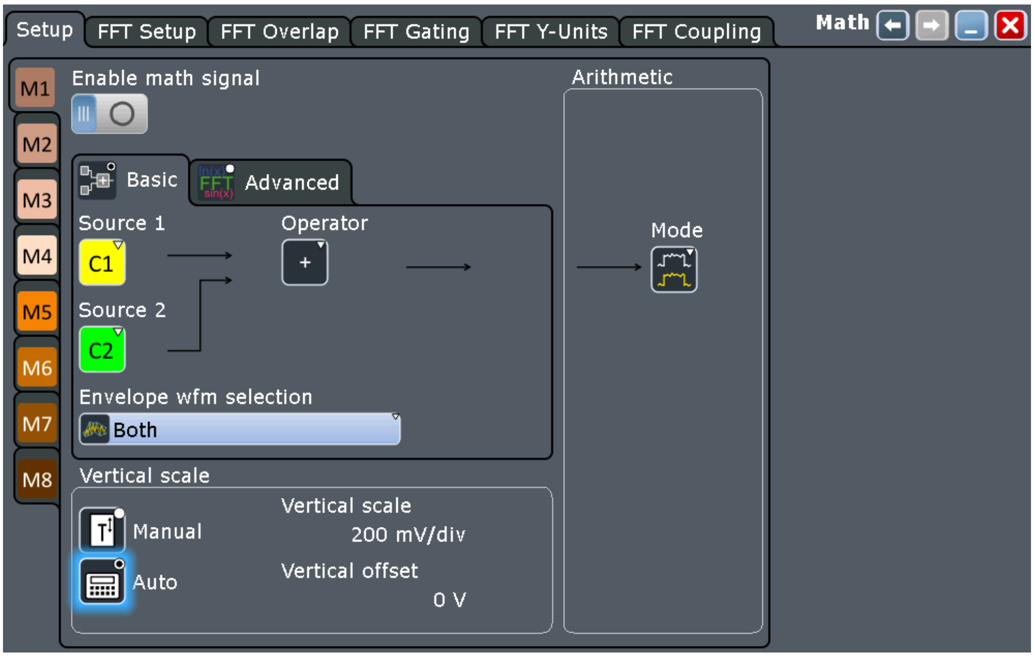
   Tip: You can also use the vertical [SCALE] rotary knob for scaling. In this case, the scale mode is set to "Manual" temporarily.

6. If you need the envelope or average of the math waveform over several calculations, change the arithmetic mode for the waveform as for channel waveforms.
   See also: "Arithmetic" on page 144.

7. Close the "Math Setup" dialog box.

6.3.2 Math Setup - General Settings

You can define up to eight different math waveforms. Each waveform is defined in a separate tab in the "Math" dialog box ("M1" to "M8").



The settings for input of mathematical formulas in basic and advanced editors are described in separate chapters:

- Chapter 6.3.3.1, "Settings in the Basic Editor", on page 261
- Chapter 6.3.4, "Advanced Expressions", on page 263

The general settings for enabling, scaling and waveform arithmetic are:

Enable Math Signal 259
Vertical Scale 259
└ Vertical scaling mode (Manual/Auto) 259
└ Vertical Scale 259
└ Vertical Offset 259
Arithmetic 259

└ Mode...260
└ Reset..260
└ Average count (N-single count).. 260

Enable Math Signal
If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:
`CALCulate:MATH<m>:STATe` on page 1190

Vertical Scale
Functions to set the vertical parameters of the math waveform.

Note: If an FFT expression is defined, the vertical scaling for spectrum displays is available: "Vertical maximum" and "Vertical range" instead of "Vertical Scale" and "Vertical Offset". See Chapter 8.1.3.2, "FFT Overlap", on page 370.

Vertical scaling mode (Manual/Auto) ← Vertical Scale
By default, the vertical scale is adapted to the current measurement results automatically to provide an optimal display. However, if necessary, you can define scaling values manually to suit your requirements.

Note: When you change the scaling values manually using the "Scale" rotary knob, the scale mode is set to "Manual" temporarily. When you edit the math function, scaling is automatically set back to "Auto" mode. "Manual" mode is only maintained during math function changes if you select it yourself.

"Manual"	Enter the required values for "Vertical scale" and "Vertical offset". For FFT, set "Vertical range" and "Vertical maximum".
"Auto"	"Vertical scale" and "Vertical offset" are read-only. For FFT, only the "Vertical maximum" is read-only.

Vertical Scale ← Vertical Scale
Defines the scale of the y-axis in the math function diagram. The value is defined as "<unit> per division", e.g. *50m V/div*. In this case, the horizontal grid lines are displayed in intervals of 50 mV.

If the "Vertical scaling mode (Manual/Auto)" on page 259 is set to "Auto", this setting is read-only.

Remote command:
`CALCulate:MATH<m>:VERTical:SCALe` on page 1192

Vertical Offset ← Vertical Scale
Sets a voltage offset to adjust the vertical position of the math function on the screen. Negative values move the waveform, positive values move it down.

If the "Vertical scaling mode (Manual/Auto)" on page 259 is set to "Auto", this setting is read-only.

Remote command:
`CALCulate:MATH<m>:VERTical:OFFSet` on page 1191

Arithmetic
Functions to specify the waveform arithmetic for the math waveforms.

Mode ← Arithmetic

Waveform arithmetic builds the resulting waveform from several consecutive acquisitions and subsequent math calculations of the signal. For details, see "Arithmetic" on page 144.

"Original"	The original results are displayed.
"Envelope"	The envelope curve of all acquired and calculated results is displayed.
"Average"	The average of all acquired and calculated results is displayed.
"RMS"	The root mean square of the math data is displayed. The result is the average power spectrum. If you measure the channel power on this RMS spectrum, you get the same result as for the average channel power measurement on waveforms.
"MinHold"	Determines the minimum result for each input value from the data of the current acquisition and some acquisitions before.
"MaxHold"	Determines the maximum result for each input value from the data of the current acquisition and some acquisitions before.

Remote command:

`CALCulate:MATH<m>:ARIThmetics` on page 1190

Reset ← Arithmetic

Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:

`ACQuire:ARESet:IMMediate` on page 1078

Average count (N-single count) ← Arithmetic

Access:

- TRIGGER > "Control" tab > "Average count (N-single count)"
- [ACQUISITION] > "Average count (N-single count)"
- [HORIZONTAL] > "Fast Segmentation" tab > disable "Acquire maximum" > "Required"
- [MATH] > "Setup" tab > "Mode" is not "Off" > "Average count"

The acquisition and average count has several effects:

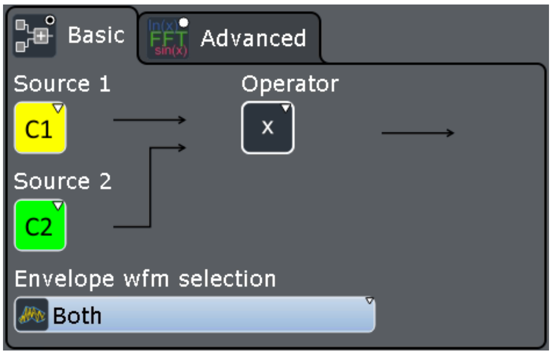
- It sets the number of waveforms acquired with [RUN N× SINGLE]
- It defines the number of waveforms used to calculate the average waveform.
  Thus, the instrument acquires sufficient waveforms to calculate the correct average if "Average" is enabled for waveform arithmetic. The higher the value is, the better the noise is reduced.
- It sets the number of acquisitions to be acquired in a fast segmentation acquisition series. Thus, you can acquire exactly one fast segmentation acquisition series with [RUN N× SINGLE].
  If fast segmentation is enabled and configured to acquire the maximum number of acquisitions, the acquisition count is set to that maximum number and cannot be changed. See also: "Number of acquisitions" on page 146.
- It is the "Finished" criteria for the state of a mask test.

Remote command:

`ACQuire:COUNt` on page 1078

6.3.3 Basic Editor

In the basic editor, you can define the most common mathematical formulas without knowing their correct syntax.



Remote command:

- `CALCulate:MATH<m>[:EXPRession][:DEFine]` on page 1189

6.3.3.1 Settings in the Basic Editor

Source 1 / 2 261
Operator 261
Noise reject 262
a / b 262
Envelope wfm selection 262

Source 1 / 2

Defines the signal source to be evaluated by the math function. Channel waveforms can be selected.

Note: If you require other signal sources not listed here, use the formula editor provided in the "Advanced" tab. In advanced mode, any waveform of any input channel can be used as a source. See: Chapter 6.3.4, "Advanced Expressions", on page 263.

Operator

Defines the type of operation to be performed on the selected signal sources. The following functions are available:

Note: If you require other operators not listed here, use the formula editor provided in the "Advanced" tab. See: Chapter 6.3.4, "Advanced Expressions", on page 263.

"+"	Adds up the sources
"-"	Subtracts source 2 from source 1.
"x"	Multiplies source 1 by source 2.
"|x|"	Determines the absolute value of the source.
"dx/dt"	Differentiates the source value with respect to the time value. Not possible on envelope waveforms and waveforms with "Peak detect" decimation.
"log(x)"	Calculates the logarithm of the source value based on 10.

"ln(x)"	Calculates the natural logarithm of the source value (based on e).
"ld(x)"	Calculates the binary logarithm of the source value (binary logarithm, based on 2).
"Rescale"	Rescales the source values by a factor "a" and an offset "b": *ax+b*. See also: "a / b" on page 262.
"FIR filter"	Finite impulse response filter - highpass, lowpass or bandpass filter for a specified cut-off frequency and characteristic. See Chapter 6.3.5.1, "FIR Filter in the Basic Math Editor", on page 272.
"Mag(FFT(x))"	Determines the magnitude of the FFT for the source values.

Noise reject

Only available for the "dx/dt" operator.

Sets the number of neighboring samples that are skipped for differentiation.

To suppress noise effects during differentiation, it can be useful not to consider two directly neighboring points to calculate *dx* (x_n-x_{n-1}). Instead, some samples in-between are skipped and a point a few samples further is used (e.g. x_n-x_{n-3}).

a / b

Defines the values for the "Rescale" function (ax+b).

"a"	Is the factor the signal source is multiplied with.
"b"	Is the offset of the signal source on the y-axis.

Envelope wfm selection

Selects the upper or lower part of the waveform for mathematic calculation, or a combination of both.

The setting is relevant for waveforms with waveform arithmetic mode "Envelope" or with "Peak detect" decimation. All mathematic operations - except for derivation - can be applied to envelope waveforms and waveforms with "Peak detect" decimation.

Remote command:
`CALCulate:MATH<m>:ENVSelection` on page 1190

6.3.3.2 Defining a Formula in the Basic Editor

1. In the "Math" menu, select "Math Setup".
   Alternatively, press the [MATH] key.
2. In the "Setup" tab, select the "Basic" tab.
3. Tap the "Source 1" and "Source 2" icons and select the signal sources to which the math function is applied. For details on available signal sources, see "Source 1 / 2" on page 261.
4. Tap the "Operator" icon and select the mathematical function.
   For details on available operators, see "Operator" on page 261.

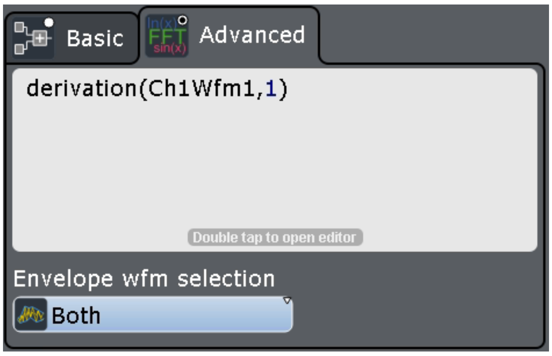
5. If the operator requires additional parameters, enter them in the input fields.

6.3.4 Advanced Expressions

In the "Advanced" tab, you can enter complex formulas to define a math waveform. The formula editor helps to enter formulas easily with correct syntax, using a large selection of operators and signal sources.

All formulas in the advanced formula editor are linear regardless whether a [dB] is set or not.

► Double-tap the "Advanced" tab to display the formula editor.



6.3.4.1 Advanced Formula Editor

Using the formula editor you can define math functions freely, using a large selection of operators and signal sources. For a procedure on using the editor, see Chapter 6.3.4.10, "Defining a Formula in the Advanced Formula Editor", on page 271.

Remote command:

- `CALCulate:MATH<m>[:EXPRession][:DEFine]` on page 1189

Buttons of the formula editor and their usage

Table 6-1: Basics

Icon	Description	Usage/Comment, *FormulaEditor expression*
(	left bracket	enclose operands
,	comma	separates operands
)	right bracket	enclose operands
e / π	math. constants	e: Euler number: 2.7182... Pi: 3.1415...
[	left square bracket	enclose unit
V / A / Ω	units	[<unit>]
]	right square bracket	enclose unit
x^a	exponentiation with base x	x: base, a: exponent *x^a*
/	division	
*	multiplication	
-	subtraction	
+	addition	
0...9	numeric characters	
.	decimal point	

Icon	Description	Usage/Comment, *FormulaEditor expression*
Exp	exponentiation with base 10	*e*
Enter	expression complete	insert expression in "Setup" dialog and close the formula editor
Clear	clear expression in editor	restart editing
Del	Delete	remove selected part of expression
Back	Backspace	remove last symbol, operator or operand to the left of the cursor
M / k / µ	SI-prefix for unit	<SI-prefix>[<unit>]

Table 6-2: Signal sources

Icon	Description	Usage/Comment, *FormulaEditor expression*
Ch	signal waveform	*Ch<1...4>Wfm<1...3>*
Math	math waveform	*Math<1...4>*
Ref	reference waveform	*Ref<1...4>*
Meas	measurement waveform	*Meas<1...8>*
Track	track waveform	*Track<1...8>*

Table 6-3: Cursor keys

Icon	Description	Usage/Comment, *FormulaEditor expression*
|←	move cursor to beginning	
←	move cursor 1 step to the left	
→	move cursor 1 step to the right	
→|	move cursor to end	

6.3.4.2 Math Functions: Algebra

Table 6-4: Algebra

Icon	Description	Usage/Comment, *FormulaEditor expression*
$\|x\|$	absolute x value	*abs(x)*
$\sqrt{x}$	square root of x	*sqrt(x)*
x^2	x*x	*pow(x)*
$\log_{10}$	common logarithm (base 10)	*log(x)*
$\log_e$	natural logarithm (base e)	*ln(x)*
$\log_2$	binary logarithm (base 2)	*ld(x)*
e^x	exponentiation with base e	*exp(x)*
$\int x dx$	integral of x	*integral(x)*

Icon	Description	Usage/Comment, *FormulaEditor expression*
d/dx	derivation of x	*derivation(x,y)* with x = waveform and y = number of skipped samples (noise reject)
ax+b	scaling of x	*rescale(x,a,b)*

6.3.4.3 Math Functions: Trigonometry

Table 6-5: Trigonometry (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
sinh	hyperbolic sine	*sinh(x)*
cosh	hyperbolic cosine	*cosh(x)*
tanh	hyperbolic tangent	*tanh(x)*

6.3.4.4 Math Functions: Bit Operations

Table 6-6: Bit operations

Icon	Description	Usage/Comment, *FormulaEditor expression*
digitize	convert to 0 or 1	*digitize(x)*
not	negation	*not(x)*
and		*and*
nand	negation of and	*nand*
or		*or*
nor	negation of or	*nor*
xor	exclusive or	*xor*
nxor	negation of exclusive or	*nxor*

6.3.4.5 Math Functions: Comparison

Table 6-7: Comparison

Icon	Description	Usage/Comment, *FormulaEditor expression*
=	equal	=
≠≠	not equal	<>
<	smaller	<
>	greater	>
≤	smaller or equal	<=

Icon	Description	Usage/Comment, *FormulaEditor expression*
≥	greater or equal	>=
More	display additional keys	

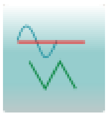
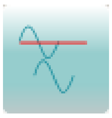
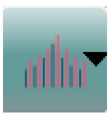
6.3.4.6 Math Functions: FFT

Table 6-8: FFT (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
|FFT|	magnitude of FFT value	*fftmag(x)*
FFT (φ)	FFT phase value	*fftphi(x)*
FFT -dφ*df	FFT group delay	*fftgroupdelay(x)*
FFT (re)	real part of FFT value	*fftre(x)*
FFT (im)	imag part of FFT value	*fftim(x)*

6.3.4.7 Math Functions: Correlation

Table 6-9: Correlation (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
	Cross correlation function of two waveforms Measures the similarity of two waveforms as a function of a time-lag applied to one of them. Function limits the maximum record length to 4 MSa. Two modes of normalization are supported: biased and unbiased. The length of the correlation buffer is $N_0 + N_1 - 1$ samples. The length of the first input signal is N_0 samples and the length of the second signal is N_1 samples.	*correlation(x1, x2, biased)* *correlation(x1, x2, unbiased)* with x1 = waveform 1 and x2 = waveform 2 *correlation(x1, x2)* performs an unbiased correlation
	Auto correlation Used to find repeating patterns, for example, a periodic signal obscured by noise. The length of the auto correlation buffer is $2N - 1$ samples, if the length of the input signal is N samples. Two modes of normalization are supported: biased and unbiased.	*autocorrelation(x, biased)* *autocorrelation(x, unbiased)* with x = channel waveform *autocorrelation(x)* performs an unbiased autocorrelation
	biased / unbiased normalization for correlation and auto correlation	see above

Mathematic background for correlation:

$$Temp1\_R_{xy}(m)=\sum_{n=0}^{N1} y_n^{*}x_{n+m} \quad m\in[0;N_1[$$

$$Temp0\_R_{xy}(m)=\sum_{n=1}^{N0} x_n^{*}y_{n+m} \quad m\in[1;N_0[$$

The R&S RTE uses only the real part of the signal. Two modes of normalization are supported: biased and unbiased.

$$R_{xy}(m)=\begin{cases}\dfrac{1}{\min(N_0,N_1)}Temp1\_R_{xy}(m) & m\in[N_0-1;N_1+N_0-1[\\ \dfrac{1}{\min(N_0,N_1)}Temp0\_R_{xy}^{*}(-m) & m\in[0;N_0-1[\end{cases}$$

Equation 6-1: Biased correlation

$$R_{xy}(m)=\begin{cases}\dfrac{1}{a(m)}Temp1\_R_{xy}(m) & m\in[N_0-1;N_1+N_0-1[\\ \dfrac{1}{a(m)}Temp0\_R_{xy}^{*}(-m) & m\in[0;N_0-1[\end{cases}$$

Equation 6-2: Unbiased correlation

Mathematic background for auto correlation:

$$R_{xx}(m)=\begin{cases}\dfrac{1}{N}\sum_{n=0}^{N-m-1}x_n^{*}x_{n+m} & m\ge 0\\ R_{xx}^{*}(-m) & m<0\end{cases}$$

Equation 6-3: Biased auto correlation

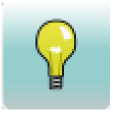
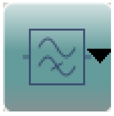
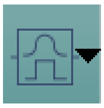
$$R_{xx}(m)=\begin{cases}\dfrac{1}{N-|m|}\sum_{n=0}^{N-m-1}x_n^{*}x_{n+m} & m\ge 0\\ R_{xx}^{*}(-m) & m<0\end{cases}$$

Equation 6-4: Unbiased auto correlation

The R&S RTE uses only the real part of the signal.

6.3.4.8 Math Functions: Filter and Power

Table 6-10: Filter and power (More keys)

Icon	Description	Usage, comment, *FormulaEditor expression*
	Electric power	Electric power is calculated from voltage, based on measurement impedance (see "Measurement impedance" on page 151) *elecpower(x)* = U^2/R
	Finite impulse response (FIR) filter	*FIR(tpye,source,limit,shape)* or *FIR(userdef,source,path)* See Chapter 6.3.5.2, "FIR Filter in the Advanced Editor", on page 273
	Type of FIR filter	*highpass* or *lowpass*, see FIR filter
	Characteristics of FIR filter	*gaussian* or *rectangle*, see FIR filter
	Moving average	Calculates a mean value of several adjacent sample points. The result is a smoothed waveform. The moving average uses the full data and can be used for non-periodic signals. It works like a lowpass filter and increases the vertical resolution at the expense of bandwidth reduction. *MovingAverage(x,y)* with: x = source (channels only), y = number of samples to be averaged Example: *MovingAverage(Ch1Wfm1,1000)* Averages 1000 subsequent samples of the channel 1 waveform

6.3.4.9 Transfer Functions of a Step-Like Signal

The R&S RTE provides two functions to calculate the transfer function of a system out of its step response. These functions are not supported by the formula editor, but you can enter them directly in the "Advanced" tab in the "Math Setup" dialog box. To enter the formula, you need a keyboard (onscreen keyboard or a connected usual one).

Math expression	Description
Step2FreqRespNormMag(x,points) Where: • x is the waveform (channel, math, reference) • points is the number of points in the resulting math waveform (min. 1000 points)	Normalized magnitude of transfer function in frequency domain Example: *Step2FreqRespNormMag(Ch1,2000)*
Step2FreqRespNormPhi(x,points,time offset) Where: • x is the waveform (channel, math, reference) • points is the number of points in the resulting math waveform (min. 1000 points) • time offset in s	Normalized phase of transfer function in frequency domain Example: *Step2FreqRespNormPhi(Ch1,2000,2e-9)*

The calculation of transfer functions has several steps:

- Calculate the frequency domain representation of a step-like signal. A detailed description is given in: A. M. Nicolson, „Forming the fast Fourier Transform of a step response in Time-Domain Metrology,“ in Electronic letters, Bd. 9, Nr. 14, p. 317, 1973.
- Calculate the derivative to convert the result to the transfer function.
- Normalize the result at DC to a magnitude of 1 V and subtract specified time offset from phase.

The step-like signal to be analyzed is a time domain waveform, and the result is a math waveform in the frequency domain.

To get useful results, check and adjust the following settings:

1. Place the analyzed step of the signal in the left half of the time domain diagram. Use the "Reference point" or the horizontal "Position" of the trigger to move the signal. See also: Chapter 4.2, "Horizontal Settings", on page 138.
2. Adjust the record lengths of the source signal and the math signal. The "Record length" of the step-like source signal must be shorter than the number of points in the resulting math function, which is set in the formula. See also: "Record length" on page 140.
3. If the resulting math waveform is noisy, increase the "Average count" for the source signal until the waveform is clear. See also: Chapter 4.2.2, "Acquisition", on page 143.

The resulting math waveforms are specific FFT waveforms, thus only chosen FFT settings in the "Math" dialog are supported:

- "Setup" tab:
  - "Arithmetic Mode" = Off
  - "Vertical Scale"
- "FFT Gating" tab: all settings
- "FFT Y-Units" tab:
  - "Magnitude unit": recommended units are dBV and Linear
  - "Phase unit"

6.3.4.10 Defining a Formula in the Advanced Formula Editor

1. In the "Math" menu, select "Math Setup".
2. In the "Setup" tab, select the "Advanced" tab.
3. Double-tap the editing area.
   The "Formula Editor" is displayed.
4. Enter the math formula including all required signal sources and operators by selecting the corresponding keys in the editor. For details on the available keys, see Chapter 6.3.4.1, "Advanced Formula Editor", on page 263.
5. To insert a physical unit in the formula, proceed as follows:
   a) If necessary, insert a decimal prefix using the "M/k/µ" key.
   b) Insert an opening square bracket using the "[" key.
   c) Insert the physical unit using the "V/A/Ω" key.
   d) Insert a closing square bracket using the "]" key.

   The resulting expression could be, for example: `m[V]`
6. To perform a rescaling function, proceed as follows:
   a) Select the rescaling function using the "ax+b" key.
   b) Behind the left bracket, insert the signal source using one of the following keys:
      - "Ch" for a channel
      - "Math" for a math function
      - "Ref" for a reference waveform
      - "Meas" for a measurement

   c) Insert a comma using the "," key.
   d) Insert the "a" value, i.e. the scaling factor, using the number keys.
   e) Insert a comma using the "," key.
   f) Insert the "b" value, i.e. the scaling offset, using the number keys.
   g) Insert the closing bracket using the ")" key.

   The resulting expression could be, for example: `rescale(Ch1Wfm1,3,4)`

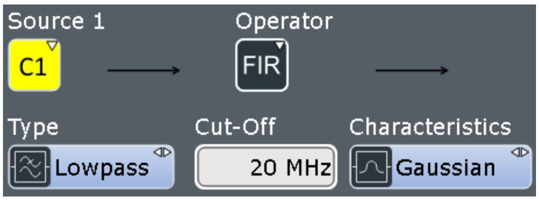
6.3.5 Filters

The R&S RTE provides several ways to filter the input signal:

- Lowpass filter by selecting a bandwidth limit in the acquisition path (vertical channel settings)
  See "Bandwith" on page 149
- Digital lowpass filter
  See Chapter 4.8, "Digital Filter Setup", on page 192
- FIR filter to create a filtered math waveform (lowpass, highpass, bandpass)
  You can set up a FIR-filtered math waveform in the basic math editor, or in the advanced editor. Both ways are described below in this chapter.

6.3.5.1 FIR Filter in the Basic Math Editor

The finite impulse response filter ("Operator" = FIR) is a filter to create filtered waveforms with lowpass, highpass, or bandpass. The filter requires additional settings.



- "Type": defines whether the FIR filter is a highpass, lowpass, or bandpass/bandstop (= "User defined") filter.
- "Cut-Off": sets the limit frequency for the highpass or lowpass FIR filter.
- "Characteristics": relevant for lowpass filter. Defines whether it has a Gaussian or a rectangular shape. The highpass is always Gaussian.
- "Select filter file": relevant for bandpass/bandstop filter. Opens a file dialog to select the file with the filter coefficients.

Cut-off frequency for lowpass FIR filter

The cut-off frequency depends on the horizontal resolution and the filter characteristics. The frequency for the lowpass filter can only be set in this range:

f_g_3dB = (0.001 ... 0.2) f_a_in* for Gaussian FIR filter

f_g_3dB = (0,001 ... 0.4) f_a_in* for rectangular FIR filter

Where: *f_g_3dB* = cut-off frequency to be set for the lowpass filter, and *f_a_in* = reciprocal of the resolution, or sample rate.

Cut-off frequency for highpass FIR filter

To check limit frequency for the highpass filter, convert it to an equivalent lowpass frequency:

f_LP = f_a_in/2 - f_HP

Where f_HP is the requested highpass limit frequency and f_LP the equivalent lowpass frequency that has to comply with the limits given above.

Bandpass and bandstop FIR filter: CSV file

To define a bandpass or bandstop, you need a CSV file that contains the comma-separated filter coefficients. No other parameters are allowed in the file. To create the CSV file, we recommend using the Matlab Filter Design & Analysis tool. In the tool, enter the filter type, filter order and filter frequencies. Make sure to set the sample frequency in Matlab and the sample rate at the oscilloscope to the same value. If the values differ, the filter is shifted in frequency.

6.3.5.2 FIR Filter in the Advanced Editor

You can type the FIR filter formula directly in the "Advanced" tab, or open the advanced formula editor and use the buttons to create the formula.

Highpass and lowpass FIR filter

The general syntax for highpass and lowpass filters is:

FIR(tpye,source,limit,shape)

- Type is *lowpass* or *highpass*
- Source is the input channel of the signal.
- Limit is the cut-off frequency
- Shape is *gaussian* or *rectangle* for the lowpass filter, and *gaussian* for the high-pass.

► For example, to set a rectangle lowpass filter on channel 1 with 10 MHz cut-off frequency, enter:
FIR(lowpass,Ch1,1e+07,rectangle)

► For example, to set a Gaussian highpass filter on channel 2 with 1.5 GHz cut-off frequency, enter:
FIR(highpass,Ch2,15e+08,gaussian)

Bandpass and bandstop FIR filter

The syntax for the bandpass filter is:

FIR(userdef,source,path)

- Source is the input channel of the signal.
- Path is a string containing the path and filename of the filter file. The file contains the comma-separated filter coefficients.

► For example, to set a bandpass on channel 1 with filter coefficients saved in the `bandpass.csv` file, enter:
FIR(userdef,Ch1,"C:\Users\Public\Documents\Rohde-Schwarz\RTx\bandpass.csv")

6.4 History

The history accesses the data of previous acquisitions and provides them for further analysis.

6.4.1 About History

If a continuous acquisition runs, the captured data is stored in the sample memory and the current acquisition is processed and shown on the display. After the acquisition was stopped, the history accesses the captured samples that were stored, displays these samples as history waveforms, and makes them available for further analysis. It considers all channels that were enabled during the running acquisition. When a new acquisition is started with [RUN CONT] or [RUN N× SINGLE], the memory is cleared and written anew.

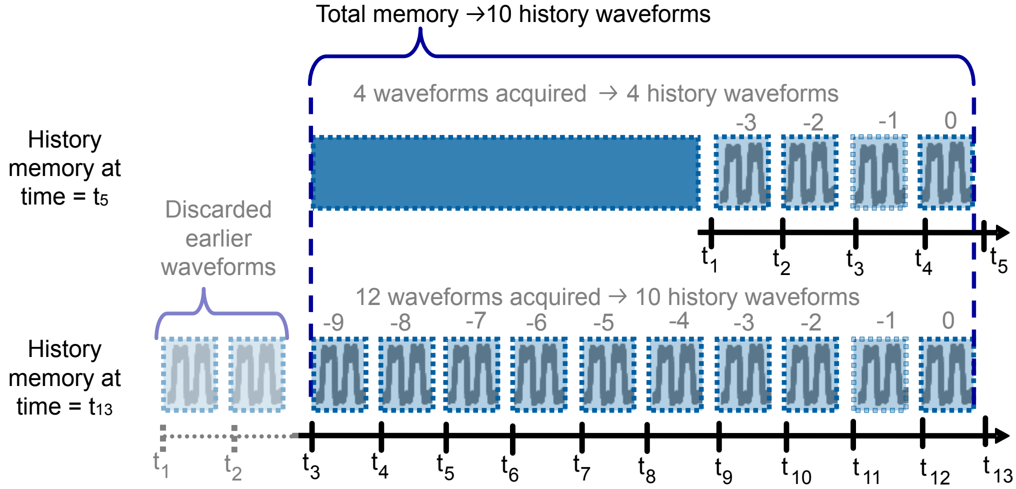


Figure 6-6: History memory. In this example, the memory can store 10 waveforms.

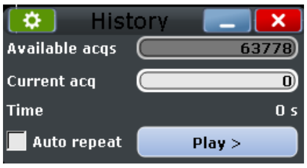
You can work with history waveforms in the same way as with the waveform of the latest acquisition: use zoom, cursor measurements, and automatic measurements, create math waveforms, perform mask testing and so on. Saving the history data is also possible, either completely or a part of the data.

The number of stored history waveforms depends on the memory size, the number of enabled channels, and the record length. The shorter the record length, the less the number of channels, and the larger the memory, the more history waveforms are saved.

Quick-access History dialog box

When you press the [HISTORY] key on the front panel or tap "Display" menu > "Show history", the history mode is enabled and the quick-access "History" dialog box is displayed. A running acquistion stops immediately.

The small quick-access "History" dialog box can remain visible on the screen during history replay, so that the history can be replayed at any time by a simple tap on the "Play" button. Closing the quick-access "History" dialog box, or starting a new acquisition disables the history mode.



Export of history waveforms

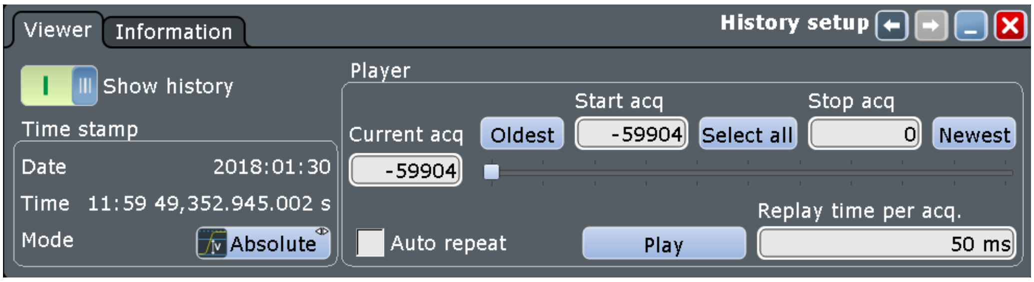
You can export history data, see "To save the history data" on page 279.

6.4.2 History Setup

The "History" dialog box contains the complete functionality on history viewing and information. The most important information and functions are also provided in the quick-access history dialog box.

6.4.2.1 Viewer

The settings in the "Viewer" tab control the display of history waveforms.



The numbering of the waveforms refers to the current memory content. With every [RUN CONT] or [RUN N× SINGLE] action, the memory content changes.

Show history / Export history
Enables the history mode and allows you to save history waveforms to file.

The history display is enabled automatically when you press the [HISTORY] button. It is disabled when you close the quick-access "History" dialog box.

For details on data export, see "Export history" on page 448.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]` on page 1194

Current acq
Accesses a particular acquisition in the memory to display it, or to save it. The newest acquisition always has the index "0". Older acquisition have a negative index.

If a history replay is running, the field shows the number of the currently shown acquisition.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent` on page 1195

Start acq
Sets the index of the first (oldest) acquisition to be displayed or exported. The index is always negative. The number of stored history acquisitions is shown in Available acquisitions on the "Information" tab.

Remote command:
`CHANNel<m>[:WAVeform<n>]:HISTory:STARt` on page 1195

Stop acq
Sets the index of the last (newest) acquisition to be displayed or exported. The newest acquisition of the complete acquisition series always has the index "0".

Remote command:
`CHANNel<m>[:WAVeform<n>]:HISTory:STOP` on page 1195

Select all
All acquisitions that are saved in the memory are in the viewer.

Current
Sets the newest acquisition in the sample memory as "Stop acq" and "Current acq". This acquisition always has the index "0".

Oldest
Sets the oldest acquisition in the sample memory as "Start acq" and "Current acq".

Auto repeat
If selected, the replay of the history waveform sequence repeats automatically. Otherwise, the replay stops at the "Stop index".

Remote command:
`CHANNel<m>[:WAVeform<n>]:HISTory:REPLay` on page 1196

Play
Starts and stops the replay of the history waveforms from "Start acq" to "Stop acq".

Remote command:
`CHANNel<m>[:WAVeform<n>]:HISTory:PLAY` on page 1196

Replay time per acq.
Sets the display time for one acquisition. The shorter the time, the faster the replay is. The setting takes effect for history replay and the display of a Fast Segmentation series, see Chapter 4.2.3, "Fast Segmentation", on page 145.

Remote command:
`CHANNel<m>[:WAVeform<n>]:HISTory:TPACq` on page 1196

Time stamp
The time stamp shows the time of the currently displayed history acquisition. Thus, the time relation between acquisitions is always available.

The time stamp "Mode" can be absolute or relative:

- In "Absolute" mode, the instrument shows the date and the daytime of the current acquisition.
- In "Relative" mode, the time difference to the newest acquisition (index = 0) is shown.

The time stamp can be included in waveform data export, see "Timestamps" on page 449.

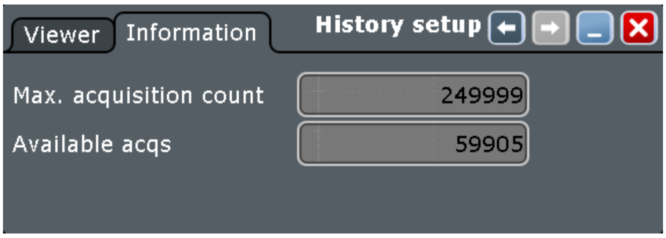
During history replay, the time value is displayed and updated if the replay speed ("Time per acquisition") is slow enough, that is 40 ms or slower.

The quick-access history dialog box always shows the relative time. In the "History Viewer" tab, you can select the time mode.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?` on page 1197
`CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?` on page 1197
`CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?` on page 1197
`CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?` on page 1198

6.4.2.2 Information

The "Information" tab shows the maximum number and the number of captures acquisitions.



Max. acquisition count
Displays the maximum number of acquisitions that can be saved in the sample memory and displayed with the history viewer. With Fast Segmentation, it is also the maximum number of acquisitions in a Fast Segmentation acquisition series.

Available acquisitions
Displays the number of acquisitions currently saved in the sample memory. This memory is also used to save a Fast Segmentation acquisition series, so the number of acquisitions available for history viewing is the same as the number of acquisitions in a Fast Segmentation acquisition series.

Remote command:
`ACQuire:AVAilable?` on page 1194

6.4.3 Using History

You can access the history waveforms in two ways:

- Display a particular acquisition.
- Replay all or a part of the saved waveforms to track the signal run.

Furthermore, you can export history data to a file.

- "To open the history and get information" on page 278
- "To display a particular acquisition" on page 278
- "To replay history waveforms" on page 278

- "To exit the history" on page 279
- "To save the history data" on page 279

To open the history and get information

1. Press the [HISTORY] key on the front panel. A running acquisition is stopped, the history mode is enabled and the quick-access "History" dialog box is displayed.

   The [HISTORY] key is illuminated as long as the history mode is active.

2. Open the full configuration dialog box:
   - Tap the ⚙ icon.
   - Press the [HISTORY] key again.
   - On the "Display" menu, tap "History setup".

3. In the "History" configuration dialog box, select the "Information" tab to see how many history waveforms are saved, and how many can be saved as maximum.

To display a particular acquisition

1. In the quick-access "History" dialog box, enter the number of the required acquisition in the "Current index" field. The newest acquisition always has the index "0", older acquisitions have a negative index

2. Tap "Play" to start.

Alternatively, you can configure and start the history display from the "History" configuration dialog box:

1. Open the "History" configuration dialog box and select the "Viewer" tab.

2. If the history mode is off (the [HISTORY] key is not illuminated), select "Show history".

   The quick-access dialog box is displayed.

3. Drag the slider to the required acquisition. The current number is shown in the "Current index" field.
   Alternatively, enter the number of the required acquisition in the "Current index" field.

4. Tap "Play" to start.

To replay history waveforms

If you want to see the complete acquisition series without any setup, simply tap "Play" in the quick-access "History" dialog box. For specific analysis of history data, use the hestory "Viewer" setup.

1. In the "History" configuration dialog box, select the "Viewer" tab.

2. If the history mode is off (the [HISTORY] key is not illuminated), enable "Show history".

   The quick-access dialog box is displayed.

3. Define the part of the history you want to see by doing one of the following:
   - Tap "Select all" to see the complete history.
   - Enter the "Start Index" of the oldest acquisition to display and the "Stop Index" of the newest acquisition to display. All waveforms between the two indexes will be displayed.
     To enter the oldest or newest acquisition for either index, tap the appropriate button. The newest acquisition always has the index "0". The "Start index" is always negative.
4. Tap "Play" to start.

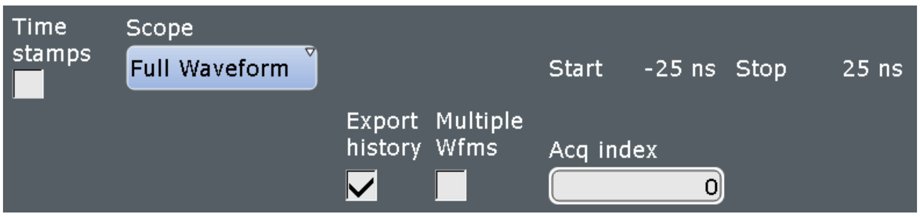
To exit the history

► Choose one of the following ways:
   - Close the quick-access "History" dialog box.
   - On the "Display" menu, tap "Show history".
   - In the "Viewer" tab, disable "Show history".
   - Start the acquisition.

To save the history data

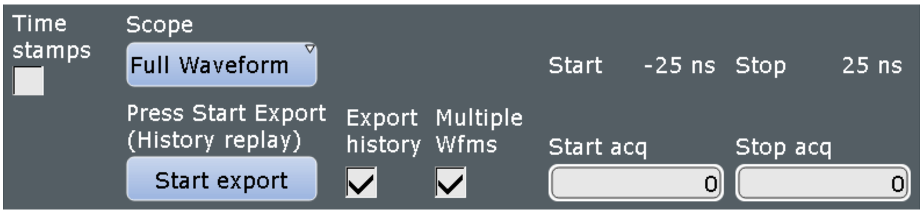
You can save the complete history, or some subsequent waveforms from the history, or a single history waveform. You can also decide to save the complete waveforms, or a part of each waveform.

1. Press the [FILE] key.
2. Select the "Save/Recall" tab.
3. Select the "Waveform" tab.
4. Tap the source icon to select the waveform you want to save.
5. If you want to save only a part of each waveform, set the "Scope".
   For settings, see "Scope" on page 447.
6. Enable "Export history".
7. If you want to write the timestamps into the data file, enable "Timestamps".
8. To save one waveform out of the history memory:
   a) Make sure that "Multiple Wfms" is disabled.
   b) Enter the number of the required acquisition in "Acq index". The newest acquisition in the memory always has the index "0". Older acquisition have a negative index.



   c) Tap "Save" or "Save As" to save the waveform data to the specified file.

9. To save several subsequent history waveforms:
   a) Enable "Multiple Wfms".
   b) Define the range of the waveforms to be saved with "Start acq" and "Stop acq".



   c) Tap "Start Export" to play the history and to save the history data to the specified file.

Se also Chapter 11.2.2, "Waveforms - Export Settings", on page 445.

6.5 XY-Diagram

XY-diagrams combine the voltage levels of two waveforms in one diagram. They use the voltage level of a second waveform as the x-axis, rather than a time base. This allows you to perform phase shift measurements, for example. You can display up to four different XY-diagrams.

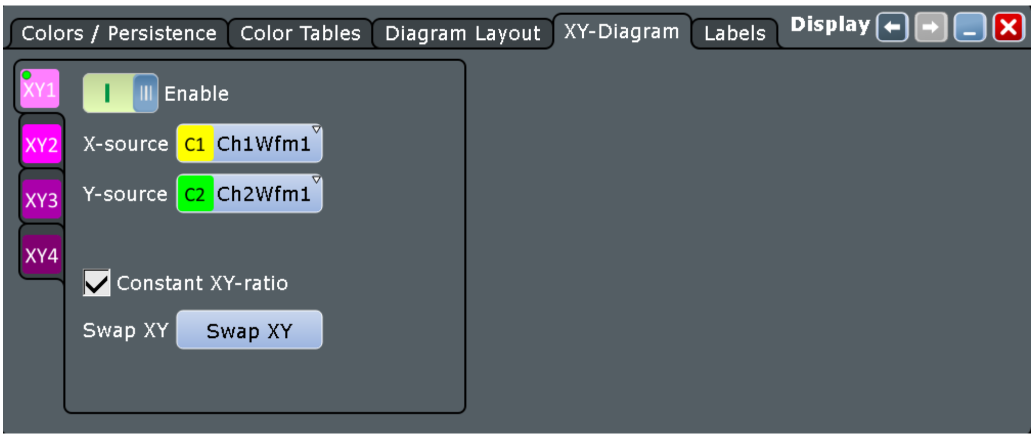
XY-diagrams can be used to display the IQ representation of a signal.

6.5.1 Settings for XY-Diagrams

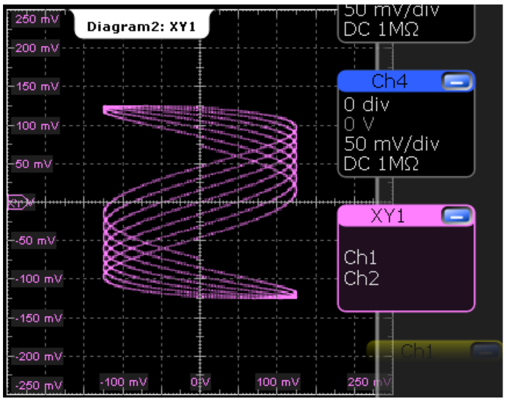
Access: "Display" > "XY-Diagram" tab

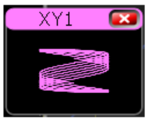
You can display up to four different XY-diagrams that use the voltage level of a waveform as the x-axis, rather than a time base.

Make sure to select the tab of the required XY-diagram.



Enable
If activated, the XY-waveform is active and shown in a diagram, or it is minimized in a signal icon.





Remote command:
WAVeform<m>:XYCurve:STATe on page 1199

X-source
Defines the signal source that supplies the x-values of the XY-diagram. Select one of the following:

- One of the waveforms of any channel
- A reference waveform
- The results of a mathematical function

Remote command:
WAVeform<m>:XYCurve:XSOurce on page 1200

Y-source
Defines the source to be used as the y-axis of the XY-diagram. Select one of the following:

- One of the waveforms of any channel
- A reference waveform
- The results of a mathematical function

Remote command:
WAVeform<m>:XYCurve:YSOurce on page 1200

Constant XY-ratio
If enabled, the x- and y-axes maintain a constant ratio in the diagram.

Remote command:
WAVeform<m>:XYCurve:RATio on page 1198

Swap XY
Replaces the source of the x-axis with the source of the y-axis and vice versa.

Remote command:
WAVeform<m>:XYCurve:SWAP on page 1199

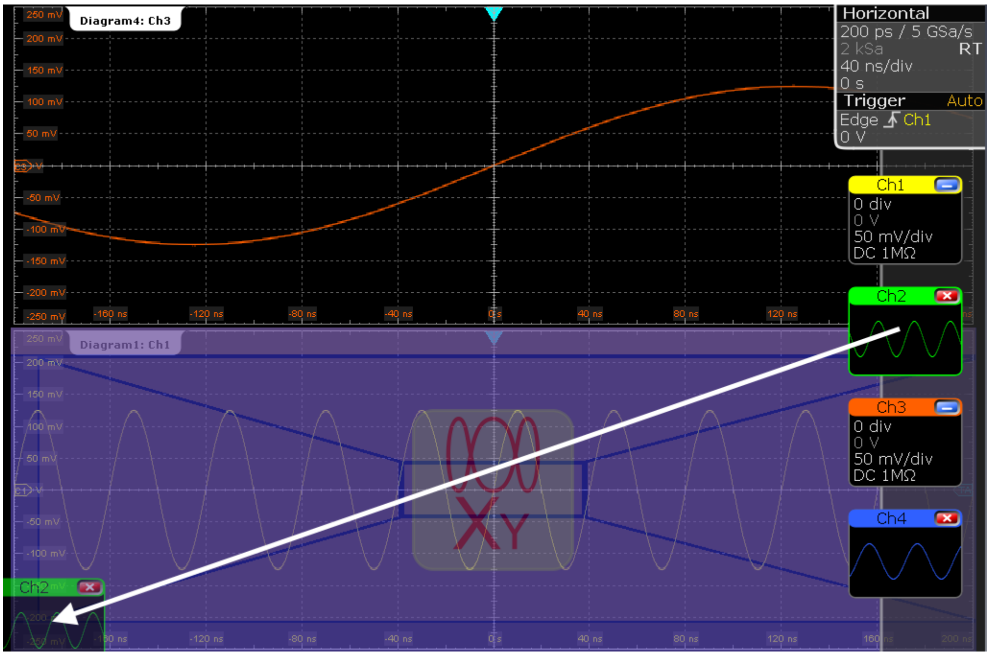
6.5.2 Displaying an XY-Diagram

You can create the diagram from active waveforms with drag&drop, or use the dialog box for setup.

To display an XY-diagram with drag&drop

Prerequisites: The source waveform for the y-axis is active in a diagram, the source waveform for the x-axis is either active or minimized.

1. Drag the x-axis waveform to the lower left corner of the diagram with the y-axis waveform.
2. Drop the icon when it overlaps the left and lower diagram borders.



The diagram is converted into an XY-diagram.

To set up an XY-diagram

1. On the "Display" menu, tap "XY-diagram".
2. Activate the "State" option.
3. In the "X-source" field, define the signal source that supplies the x-values of the XY-diagram. Select one of the following:
   - One of the waveforms of any channel
   - A reference waveform
   - The results of a mathematical function
4. In the "Y-source" field, define the signal source that supplies the y values of the XY-diagram.
5. To switch the x- and y-values quickly, tap the "Swap XY" button.

6. To maintain a constant ratio while the x- and y-axes are adapted to the acquired data dynamically, activate the "Constant XY-ratio" option.

If the XY-diagram is active or minimized, touch and hold the signal icon to open the "XY-diagram" tab.

7 Measurements

Using the R&S RTE you can perform and display different measurements simultaneously, based on the active signal or math waveforms. The color of the results in the result table corresponds with the source waveform color.

The following measurement methods are available:

- Cursor measurements: measurements can be configured for up to 2 cursor sets to determine specific results at the manually defined cursor positions of an active waveform; the results are displayed in a result box.
- Automatic measurements: up to eight measurements can be configured and performed simultaneously on different sources; the results of each measurement are displayed in a result box.
- Quick measurements: performs a set of automatic measurements on the selected waveform at the push of a button. You can configure the set of measurement types.

7.1 Cursor Measurements

- Cursors and Results of Cursor Measurements..284
- Using Cursors...286
- Settings for Cursor Measurements...288

7.1.1 Cursors and Results of Cursor Measurements

Cursor measurements determine the results at the current cursor positions. The cursors can be positioned manually, or can be configured to follow the waveform. You can measure on one waveform, or on two different waveforms (sources).

Up to 2 cursor sets can be configured and displayed. Each cursor set consists of a pair of horizontal or vertical cursors, or both. Cursor lines can be coupled so that the initially defined distance is always maintained.

The cursors are displayed in the diagrams of the source waveform only, or in all diagrams. For each measurement, labels can be defined for the cursors. By default, the cursors are labeled as C1.1, C1.2, C2.1, C2.2.

How to set up cursor measurements is described in Chapter 7.1.2, "Using Cursors", on page 286. The Chapter 7.1.3, "Settings for Cursor Measurements", on page 288 provides a detailed description of all settings.

Cursors can also define a gate to limit the measurement to the section of the waveform between the cursor lines. See Chapter 7.2.3.2, "Gate Settings for Measurements", on page 303.

The result display of cursor measurements is configurable. Results can be shown in a separate result box for each measurement, or in a separate table on the bottom. Similar to waveform diagrams, you can also minimize the result box to a result icon on the signal bar.

For details on using the result box, see Chapter 2.4.8, "Displaying Results", on page 85 and "Result position" on page 292.

7.1.1.1 Cursor Measurements on Time-Based Waveforms

The cursor for measurement on time-based waveforms returns the following results. The results are displayed automatically when a cursor measurement is enabled.

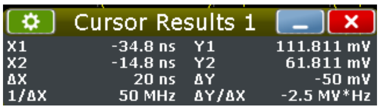


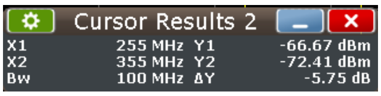
Figure 7-1: Measurement results of a cursor measurement

Label	Description
X1, X2	Time at the position of the vertical cursors.
Y1, Y2	Vertical values of the waveform at the position of the horizontal cursors in V or A.
ΔX	Difference between the vertical cursor (time) values
ΔY	Difference between the horizontal cursor values
1/ΔX	Inverse time difference
ΔY/ΔX	Slope of the waveform between the cursors (if measured on one source)

7.1.1.2 Cursor Measurements on Spectrum Waveforms

If the measurement source is a spectrum waveform, the results have a different meaning. Measurement on 2 spectrum waveforms is not possible.

The result box for measurement on spectrum waveforms shows the following information.



Label	Description
X1, X2	Frequency at the position of the vertical cursors
Y1, Y2	Vertical values of the waveform at the position of the horizontal cursors in dB.
Bw	Difference between the vertical cursor (frequency) values
ΔY	Difference between the horizontal cursor values

To set the cursor lines to the peaks, various functions are available in the "Peak Search" tab, see Chapter 7.1.3.3, "Peak Search ", on page 292.

7.1.2 Using Cursors

Cursor measurements can be started simply by using the "Cursor" icon on the toolbar. For detailed configuration, use the "Cursor" dialog box.

- Starting a Simple Cursor Measurement....................286
- Configuring a Cursor Measurement....................286
- Configuring the Cursor Display....................288

7.1.2.1 Starting a Simple Cursor Measurement

To display cursors using the toolbar

1. Select the waveform that you want to measure.
2. Tap the "Cursor" icon on the toolbar.

3. Tap the diagram where you want to set the cursors. Alternatively, you can draw a rectangle in the diagram to position the cursor lines.

   The cursor lines appear and the "Cursor Results" box for the selected waveform opens.

To display cursors using the [CURSOR] key

1. Select the waveform that you want to measure.
2. Press the [CURSOR] key.

   The cursor lines appear and the "Cursor Results" box for the selected waveform opens.

To disable one cursor measurement

► If the results are shown in a floating result box, or in a signal bar icon: Close the result box or icon.

► If the results are shown in a docked table:
   a) Tap the "Delete" icon on the toolbar.
   b) Tap the result table.

7.1.2.2 Configuring a Cursor Measurement

To modify the position of the cursor lines, you can simply drag the lines on the screen. In addition, various settings are possible to refine the measurement.

The complete configuration of cursor measurements is provided in the "Cursors" dialog box.

1. To open the "Cursors" dialog box, use one of these ways:
   - Tap the icon in the result box.
   - Press the [CURSOR] key.
   - Use the "Cursor" menu.
2. Select the "Setup" tab.
3. Select the subtab for the cursor set that you want to use.
4. Tap the source icon and select the measured waveform. you can select any active input channel, math, reference or XY-waveform.
5. If necessary, enable and select a 2nd waveform.
6. Select the cursor type: horizontal, vertical, or both.
7. Define the position of the cursors:
   a) To define exact positions of the cursors manually, enter the X-position for each vertical cursor and the Y-position for each horizontal cursor. Horizontal cursors can only be positioned manually if the "Track waveform" setting is disabled.
   b) To position the horizontal cursors automatically, select "Track waveform".
   In this case, cursor 1 indicates the current maximum, cursor 2 indicates the current minimum. If both horizontal and vertical cursors are displayed, the horizontal cursors are placed at the crossing points of the vertical cursors with the waveform. Adjust the vertical cursors manually.
   If the waveform arithmetics are set to "Envelope" and the "Track waveform" is active, select which horizontal cursor is positioned to the maximum and which to the minimum envelope values.
   c) To keep the distance between the vertical cursors when one cursor is moved, select "Coupling".
8. Optionally, select "Show in all diagrams" in the "Label / Display" tab. This setting enables the cursor display in all diagrams that are in the same domain as the selected source (time or spectrum).
9. To set the cursors for a spectrum measurement to peak values automatically, select the "Peak Search" tab.
   Tap one of the search function buttons to place the cursors on the selected peak value. For details, see Chapter 7.1.3.3, "Peak Search ", on page 292.
   Optionally, define a peak excursion. Peak excursion is the minimum level value by which the waveform must rise or fall so that it is identified as a maximum or a minimum by the search functions.
10. Tap the "Enable" icon in the "Setup" tab to activate the cursor measurement.

    The cursors are displayed in the waveform diagrams of the measurement source and the results are displayed. For details on cursor measurement results, see Chapter 7.1.1, "Cursors and Results of Cursor Measurements", on page 284.

7.1.2.3 Configuring the Cursor Display

By default, the cursors are displayed as lines in the diagrams and labeled according to the syntax: C<cursor set number>.<1|2>

The cursors for the cursor set 2, for example, are labeled 2.1 and 2.2. The horizontal and the vertical cursors lines have the same labels.

You can change the default cursor display and labels.

1. Press the [CURSOR] key.
2. Select the subtab for the cursor set you want to configure.
3. To change the display of the cursor lines:
   a) Select the "Setup" tab.
   b) Select the cursor style. See also "Cursor style" on page 289.
4. Select the "Label / Display" tab.
5. For each vertical and horizontal cursor, enter a label.
6. Select "Show labels".

7.1.3 Settings for Cursor Measurements

Cursor measurements are configured in the "Cursors" dialog box.

7.1.3.1 Cursor Setup

Access: [CURSOR] key

The "Setup" tab contains the settings for cursor measurements. If you want to save the measurement results to a file, tap "Result export". See also: Chapter 11.2.4, "Numeric Results", on page 452.

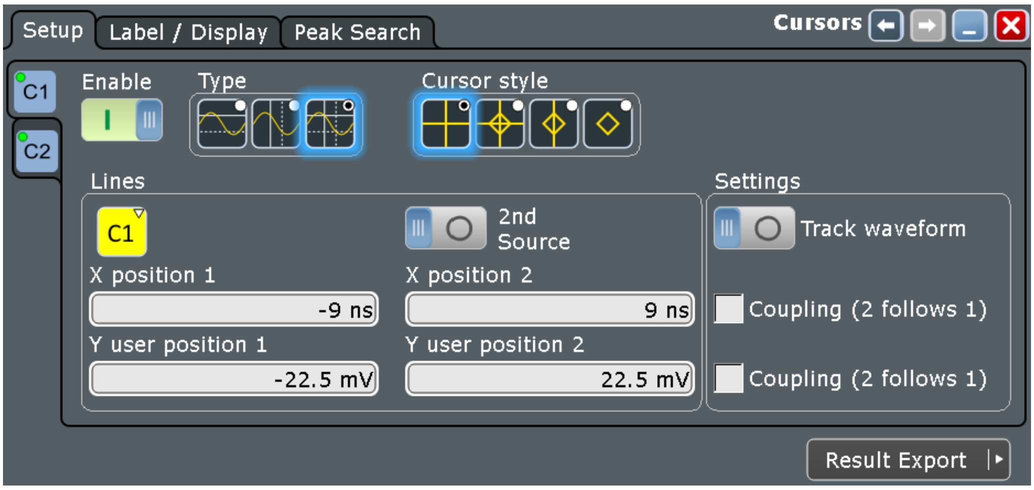


Figure 7-2: Setup for cursor on 1 source

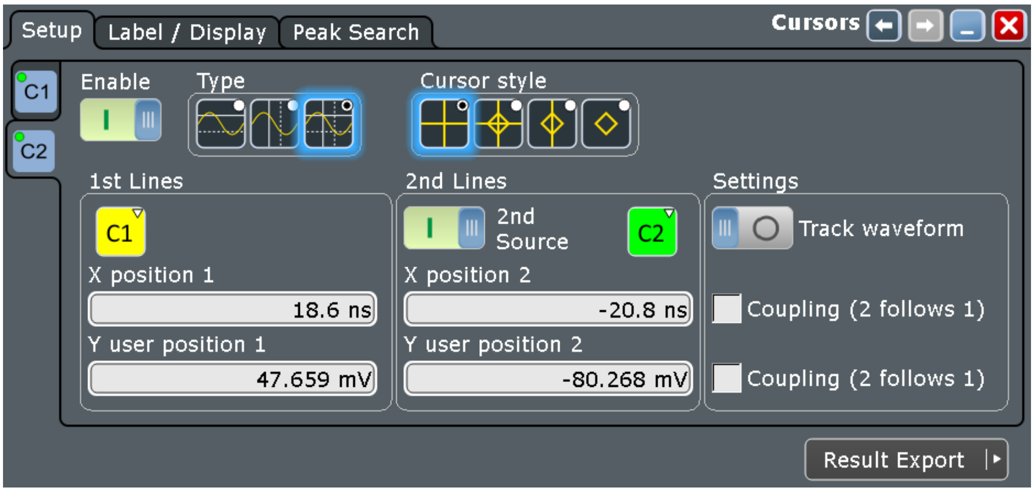


Figure 7-3: Setup for cursor on 2 sources

Cursor set (vertical tab)
The settings for each cursor measurement (or cursor set) are configured on separate tabs. For each measurement, a horizontal pair of cursors, a vertical pair of cursors, or both can be displayed.

Enable
Enables the selected cursor measurement.

Remote command:
`CURSor<m>:STATe` on page 1201

Type
Defines the cursor type to be used for the measurement.

"Horizontal cursors"	The horizontal cursors are positioned automatically along the waveform or can be positioned manually.
"Vertical cursors"	The vertical cursors are positioned manually.
"Both horizontal and vertical cursors"	The horizontal cursors are positioned automatically along the waveform or can be positioned manually. The vertical cursors are positioned manually.

Remote command:
`CURSor<m>:FUNCtion` on page 1202

Cursor style
Defines how the cursor is displayed in the diagram.

"Lines" The cursors are displayed as lines.

"Line & Rhombus" The cursors are displayed as lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

"Vertical line and rhombus" The cursors are displayed as vertical lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

"Rhombus" The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

Remote command:
`CURSor<m>:STYLe` on page 1207

Source
Defines the source of the cursor measurement. Any of the input signal, math, reference or XY waveforms can be selected.

Remote command:
`CURSor<m>:SOURce` on page 1203

2nd source
Enables and selects a second source for the cursor measurements. If enabled, the second cursor lines Cx.2 measure on the second source. Using a second source, you can measure differences between two channels with cursors.

Remote command:
`CURSor<m>:USSource` on page 1204
`CURSor<m>:SSOurce` on page 1203

X position 1|2
Defines the position of the vertical cursors.

Remote command:
`CURSor<m>:X1Position` on page 1204
`CURSor<m>:X2Position` on page 1204

Y user position 1|2
Defines the position of the horizontal cursor lines. The setting corresponds to the V1 and V2 values in the "Cursor Results" box.

If "Track waveform" is enabled, the user setting is disabled and the measurement results are displayed in the "Cursor Results" box.

Remote command:
`CURSor<m>:Y1Position` on page 1205
`CURSor<m>:Y2Position` on page 1205

Track waveform
The horizontal cursors track the waveform. The first cursor line indicates the current vertical minimum, and the second cursor line indicates the maximum. If the waveform changes, e.g. during a running measurement, the cursors move along with it. If both horizontal and vertical cursors are displayed, the horizontal cursors are positioned to the crossing points of the vertical cursors with the waveform. The measurement results are displayed in the "Cursor Results" box.

Tracking disables the Y-coupling (coupling horizontal cursor lines) and the Y user position settings.

Remote command:
`CURSor<m>:TRACking[:STATe]` on page 1202

Coupling (2 follows 1)
Couples the horizontal and vertical cursor pairs so that the distance between the two lines remains the same if one cursor is moved.

Remote command:
`CURSor<m>:YCOupling` on page 1206
`CURSor<m>:XCOupling` on page 1205

Envelope wfm selection 1|2
Envelope selection is effective under the following conditions:
- The waveform arithmetic of the cursor source waveform is set to envelope waveform (see "Arithmetic" on page 144)
- "Track waveform" is enabled.
- Both horizontal and vertical cursors are enabled ("Type" = *Both*).

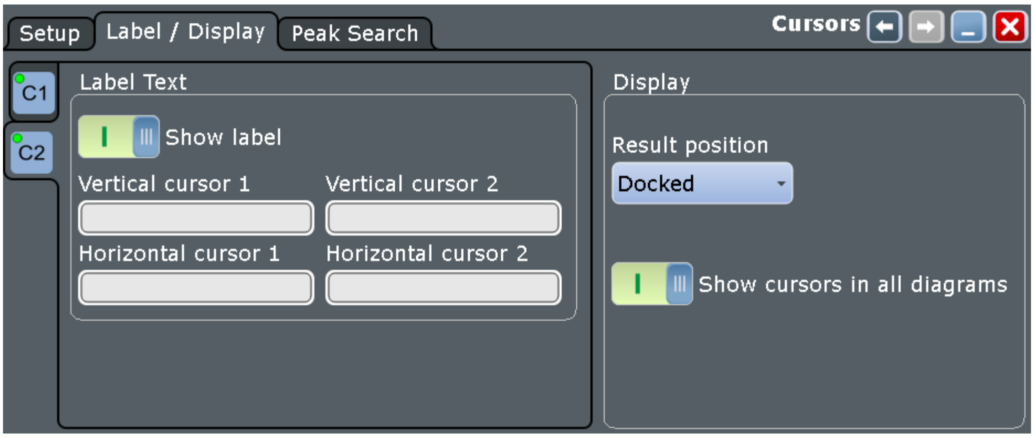
The setting defines which horizontal cursor is positioned to the maximum and which to the minimum envelope values.

"Minimum"	The horizontal cursor is set to the crossing point of the vertical cursor with the minimum waveform envelope.
"Maximum"	The horizontal cursor is set to the crossing point of the vertical cursor with the maximum waveform envelope.

Remote command:
`CURSor<m>:X1ENvelope` on page 1206
`CURSor<m>:X2ENvelope` on page 1207

7.1.3.2 Cursor Labels and Display

The settings in this tab configure the display of the cursors.



Cursor set (vertical tab)
The cursor labels for each cursor measurement (or cursor set) are configured on separate tabs.

Vertical cursor 1|2
Defines a label to be displayed with the vertical cursors.

By default, the cursors are labeled as C1.1, C1.2, C2.1, C2.2.

Horizontal cursor 1|2
Defines a label to be displayed with the horizontal cursors.

Show label
Shows the cursor labels in the diagram.

Remote command:
`CURSor<m>:LABel` on page 1208

Result position
Defines the position of the cursor measurement results.

"Floating"	Floating result box in front of the diagrams.
"Preview"	Result icon on the signal bar.
"Docked"	Fixed tab below the diagrams.

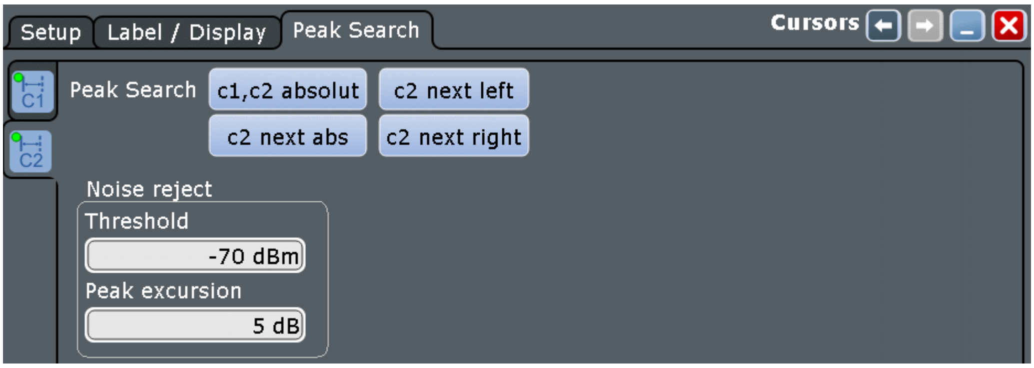
Remote command:
`DISPlay:RESultboxes:CUPosition` on page 1208

Show in all diagrams
Shows the enabled cursor measurements in all active diagrams of the same (time/ spectrum) domain.

Remote command:
`CURSor<m>:SIAD` on page 1208

7.1.3.3 Peak Search

The settings on this tab are only available in spectrum mode, i.e. the source of the cursor measurement is an FFT math waveform. In this case, the cursors can indicate the results of a peak search on the waveform. You can define which peaks the instrument determines by defining the noise reject settings.



Threshold
Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

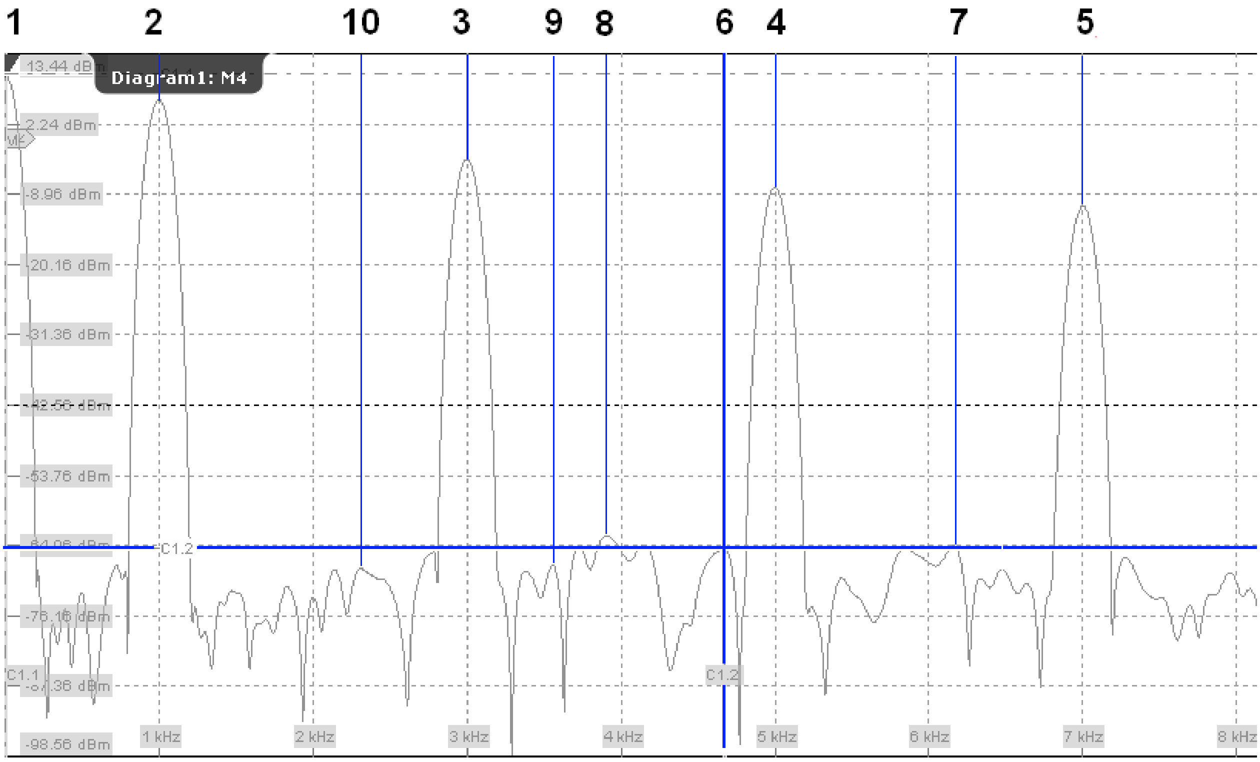
This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

Remote command:
`CURSor<m>:THReshold` on page 1211
`MEASurement<m>:SPECtrum:ATHReshold` on page 1235

Peak excursion
Defines a relative threshold, the minimum level value by which the waveform must rise or fall to be considered as a peak. To avoid identifying noise peaks, enter a peak excursion value that is higher than the noise levels.

This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

The following figure shows a cursor measurement on a spectrum waveform:



If "Peak excursion" is 30 dB, the peaks 1 to 5 are found. If "Peak excursion" is 20 dB, also the peaks 6 to 10 are found. The cursor position is on peak 6.

Remote command:
`CURSor<m>:PEXCursion` on page 1212
`MEASurement<m>:SPECtrum:PEXCursion` on page 1235

c1, c2 absolute
Both cursors are set to the absolute peak value.

Remote command:
`CURSor<m>:MAXimum[:PEAK]` on page 1210

c2 next abs
Cursor 2 is set to the next smaller absolute peak from the current position.

Remote command:
`CURSor<m>:MAXimum:NEXT` on page 1211

c2 next right
Cursor 2 is set to the next peak to the right of the current position.

Remote command:
`CURSor<m>:MAXimum:RIGHt` on page 1211

c2 next left
Cursor 2 is set to the next peak to the left of the current position.

Remote command:
`CURSor<m>:MAXimum:LEFT` on page 1211

Cursor to center
Sets the vertical cursor line C1 to the center frequency.

Remote command:
`CURSor<m>:FFT:TOCenter` on page 1210

Center to cursor
Sets the center frequency to the frequency value that is measured at cursor line C1.

Remote command:
`CURSor<m>:FFT:SETCenter` on page 1210

7.2 Automatic Measurements

The R&S RTE can perform up to 8 automatic measurements and a quick measurement simultaneously. For each measurement, various measurement types are available to measure the characteristics of a source waveform. The measurement types are grouped in categories.

The basic measurement settings are source, category, and the selection of the measurement type. You can refine the setup to get more specific results:

- **Gating**
  A gate limits the measurement to a user-defined part of the waveform. See Chapter 7.2.3, "Measurement Gates", on page 302.
- **Statistics and long term measurements**
  To evaluate time-dependent behavior of measurement results, you can use statistics, long term measurements, and tracks. You can also decide, how many measurement results per acquisition contribute to the calculation. See: Chapter 7.2.10, "Result Analysis", on page 343.
- **Limit checks and actions on test result**
  Limit and margin checks evaluate if the measurement result exceeds a specified value. You can define actions that are performed on limit or margin violation. See: Chapter 7.2.11, "Limit and Margin Checks", on page 353.

Measurement categories

The various measurement type are grouped in several categories. The category defines which sources can be analyzed.

Time domain:

- Amp/Time: amplitude and time measurements
- Eye: eye diagram measurements
- Histogram: measurements on histograms
- Protocol: available for audio signals (option R&S RTE-K5)
  Protocol: advanced analysis (only available if option R&S RTE-K35 is installed)

Frequency domain

- Spectrum: measurements on spectrum waveforms
- Histogram: measurements on histograms

Details on automatic measurements are described in the following chapters:

- Measurement Setup in General....295
- Measurement Results....298
- Measurement Gates....302
- Reference Levels....305
- Amplitude/Time Measurements....310
- Eye Diagram Measurements....321
- Spectrum Measurements....324
- Histograms and Histogram Measurements....330
- Protocol Measurements (Option R&S RTE-K35)....337
- Result Analysis....343
- Limit and Margin Checks....353

7.2.1 Measurement Setup in General

Up to 8 measurements can be defined. The "Overview" tab shows the general settings for all 8 measurements. For each measurement, the source, category, and measurement type are defined in the "Overview" tab, and you can also enable statistic evaluation for all measurements here. For some measurement types, specific settings are available. These settings are defined on the "Setup" tab.

7.2.1.1 Starting an Automatic Measurement

There are three methods to start an automatic measurement, each with slightly different effects:

- Using the "Measurement" icon on the toolbar:
  See: "To start a measurement using the toolbar icon" on page 296.
- Pressing the [MEAS] key on the front panel.
  See: "To start a measurement with the [MEAS] key" on page 296.
- Using the "Meas" menu.
  See: Chapter 7.2.1.2, "Configuring Measurements", on page 296.

To start a measurement using the toolbar icon

1. Select the waveform that you want to measure.
2. Tap the "Measurement" icon on the toolbar.

3. Define the measurement range in one of these ways:
   - To measure the complete waveform, tap the diagram with the waveform.
   - To define a gate that limits the measurement, draw a rectangle on the screen.

   The "Meas Results" are displayed.

To start a measurement with the [MEAS] key

1. Select the waveform on the screen.
2. Press the [MEAS] key.

   The measurement for the selected waveform is enabled using the next available measurement configuration. The "Meas Results" are displayed.

7.2.1.2 Configuring Measurements

1. To open the "Measurements" dialog box, choose one of these ways:

   If a measurement is already running:
   - Tap the icon in a floating result box, or double-tap a docked result box.
   - Tap the result box. Then press the [MEAS] key.

   If no measurement is running, open the "Meas" menu and select "Overview".
2. Select the measurement "Category", for example, "Amp/Time".
3. Tap "Source" and select the waveform to be measured.
   Spectrum measurements require an FFT math waveform as measurement source. Histogram measurements require a histogram as source.
4. Select the "Measurement type".
5. Depending on the selected measurement type, further settings can be required. Select the "Setup" tab. On the left side of the tab, select the subtab of the configured measurement. Adjust the settings.

   The settings are explained in the following chapters:
   - Chapter 7.2.5.2, "Settings for Amplitude/Time Measurements ", on page 315
   - Chapter 7.2.6.2, "Settings for Eye Diagram Measurements", on page 324
   - Chapter 7.2.7.2, "Settings for Spectrum Measurements ", on page 326
   - Chapter 7.2.8.5, "Settings for Histogram Measurement ", on page 336
6. Optionally, define a gate to restrict the measurement to a part of the waveform. See Chapter 7.2.3.1, "Using Measurement Gates", on page 302.

If you enabled the measurement with the toolbar icon and drew a rectangle on the diagram, the gate is already defined and enabled.

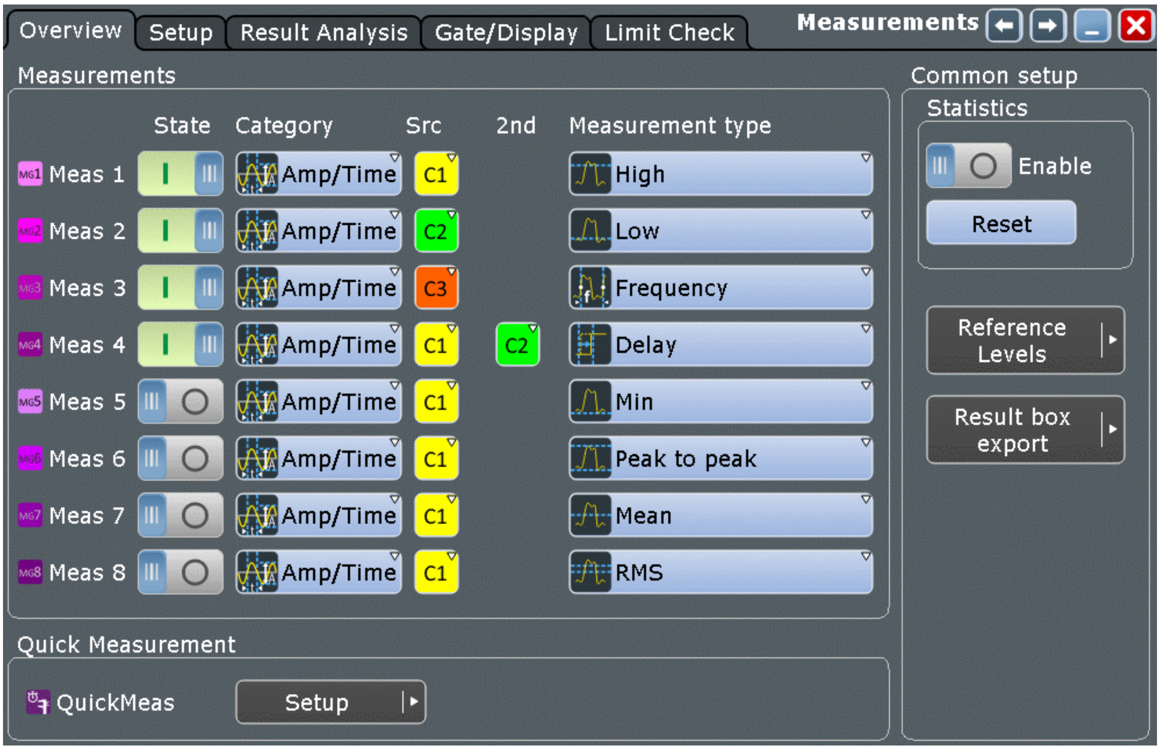
7. To compile and display statistics for the measurement, enable "Statistics". See also Chapter 7.2.10.1, "Statistics", on page 343.
8. Enable each measurement that you want to perform using "State".

   The measurement results are displayed by default below the waveform diagram.

7.2.1.3 General Measurement Settings

Automatic measurements are configured in the "Measurements" dialog box.

Access: "Meas" menu > "Overview"



This section describes the settings that relate to all measurements. Specific settings are described in the corresponding category chapters:

- Chapter 7.2.5.2, "Settings for Amplitude/Time Measurements ", on page 315
- Chapter 7.2.6.2, "Settings for Eye Diagram Measurements", on page 324
- Chapter 7.2.7.2, "Settings for Spectrum Measurements ", on page 326
- Chapter 7.2.8.5, "Settings for Histogram Measurement ", on page 336

State
Starts the selected measurement.

Remote command:
`MEASurement<m>[:ENABle]` on page 1213

Category
Measurement category. The following categories are available:

Time domain:
- Amp/Time: amplitude and time measurements
- Eye: eye diagram measurements
- Histogram: measurements on histograms
- Protocol: available for audio signals (option R&S RTE-K5)
  Protocol: advanced analysis (only available if option R&S RTE-K35 is installed)

Frequency domain
- Spectrum: measurements on spectrum waveforms
- Histogram: measurements on histograms

Remote command:
`MEASurement<m>:CATegory` on page 1215

Source, 2nd source
Define the source of the measurement. The 2nd source is required for amplitude/time measurements that are performed on two waveforms (e.g. delay, phase). Availability of sources depends on the selected category and installed options.

Remote command:
`MEASurement<m>:SOURce` on page 1213

Measurement type
Defines the measurement type for the selected measurement and category.

For details on the available measurement types, see:
- Chapter 7.2.5.1, "Overview of Amplitude/Time Measurements", on page 310
- Table 7-5
- Table 7-6
- Table 7-7

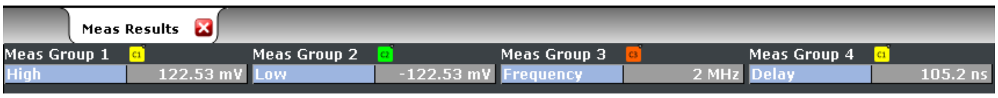
Remote command:
`MEASurement<m>:MAIN` on page 1215

Statistics / Enable
Enables the calculation and display of statistical results.

Remote command:
`MEASurement<m>:STATistics[:ENABle]` on page 1252

7.2.2 Measurement Results

By default, the results of automatic measurements are displayed below the waveform diagram when an automatic measurement is enabled.



If you want to save space in the display, drag the results to the signal bar. The most important results are displayed and updated in a results icon.

The function "Clear all" in the "Display" menu resets all results including long-term measurement and statistic results, and also deletes all waveforms and the history.

Which results are displayed depends on the selected measurements and is described in detail in the following chapters.

The following additional results are available:

- **Statistics**
  You can enable statistical evaluation of the measurement results, and select the statistical results that you want to see. Statistic information is provided in the result box. Stopping and restarting the acquisition does not reset statistics but only stops and continues them.
  See Chapter 7.2.10, "Result Analysis", on page 343
- **Measurement histograms**
  The results of measurements can be displayed in a histogram which shows the density distribution of the measurement results in a graphic and thus illustrates the statistics of the measurements.
  See Chapter 7.2.8, "Histograms and Histogram Measurements", on page 330
- **Long-term measurements**
  Long-term measurements show the behavior of measurement results over a longer time or for many samples. You can define the number of long-term points and export the long-term data, including statistical results. The measurement histogram is a vertical histogram shown in the long-term diagram.
  See: Chapter 7.2.10, "Result Analysis", on page 343
- **Intermediate results**
  You can display auxiliary result lines and reference levels in the source diagram, see Chapter 7.2.2.2, "Configuring the Results Display", on page 300.

Remote commands:

- `MEASurement<m>:ARES?` on page 1217
- `MEASurement<m>:ARNames` on page 1218
- `MEASurement<m>:RESult[:ACTual]?` on page 1218
- `MEASurement<m>:RESult:COUNt?` on page 1219

7.2.2.1 Measurement Status

The overall status of measurement results is indicated by various icons. In general, a question mark before the result value indicates that the measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. The icon colors indicate the state of the limit and margin checks.

Icon	Description
No icon, no result value ("---")	The instrument cannot measure the required value, for example, if the acquisition does not contain at least one complete period for frequency and cycle measurements. Check and adjust the waveform settings to get results.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Limit and margin checks are disabled.
	Limit and margin checks passed, measurement results are reliable.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Limit and margin checks passed.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Margin checks failed.
	Margin checks failed.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Limit checks failed.
	Limit checks failed.

7.2.2.2 Configuring the Results Display

The measurement results can be displayed in a table below the waveform diagrams, in a floating result box, or in a minimized result icon on the signal bar.

The display settings for measurements are provided on the "Gate/Display" tab, see Chapter 7.2.2.3, "Display Settings for Results", on page 301.

To display measurement information and results

You can display auxiliary lines in the diagram to determine how a measurement result was obtained. Auxiliary lines show gate areas, reference levels or intermediate result lines, such as the signal thresholds for rise and fall time measurements.

1. From the "Meas" menu, select "Gate/Display".
2. Select the subtab for the measurement you want to configure.
3. To display intermediate result lines, select "Display result lines".
4. To display reference levels, select "Display reference levels".

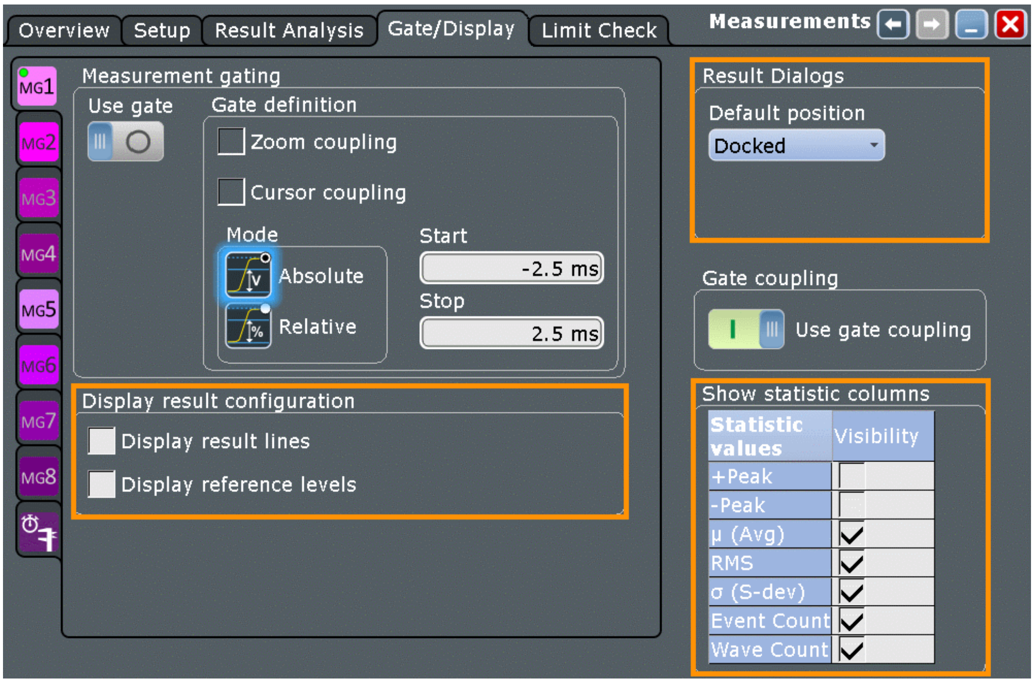
To clear the measurement results

1. On the "Display" menu, tap "Clear all".
2. To restart measurement statistics, without deleting other results, select "Reset" on the "Result Analysis" tab.

   The results in the selected measurement result box are cleared and written anew.

7.2.2.3 Display Settings for Results

The display settings for measurement results are set on the "Gate/Display" tab. Display settings are specific for each measurement.



Gate settings are described in Chapter 7.2.3.2, "Gate Settings for Measurements", on page 303.

Display result lines
Result lines mark the samples in the waveform that are used to obtain the measurement result. For example, maximum, minimum, high and low values, and mean are merked by horizontal lines; start and end are marked by vertical lines.

Remote command:
`MEASurement<m>:DISPlay:RESults` on page 1250

Display reference levels
Displays the reference levels used for the measurement in the diagram.

Remote command:
`MEASurement<m>:DISPlay:LEVels` on page 1250

Default position

Defines the default position of the result table:

- "Docked": fixed tab below the diagrams
- "Preview": result icon on the signal bar
- "Floating": floating result box in front of the diagrams

Remote command:

`DISPlay:RESultboxes:MEPosition` on page 1251

Show statistic values

Select the statistical results that you want to see in the result box.

7.2.3 Measurement Gates

Gate areas limit the measurement to a user-defined range of the waveform. The gate settings are defined on the "Gate/Display" tab.

Each measurement can use its own gate. Make sure to select the correct measurement on the left.

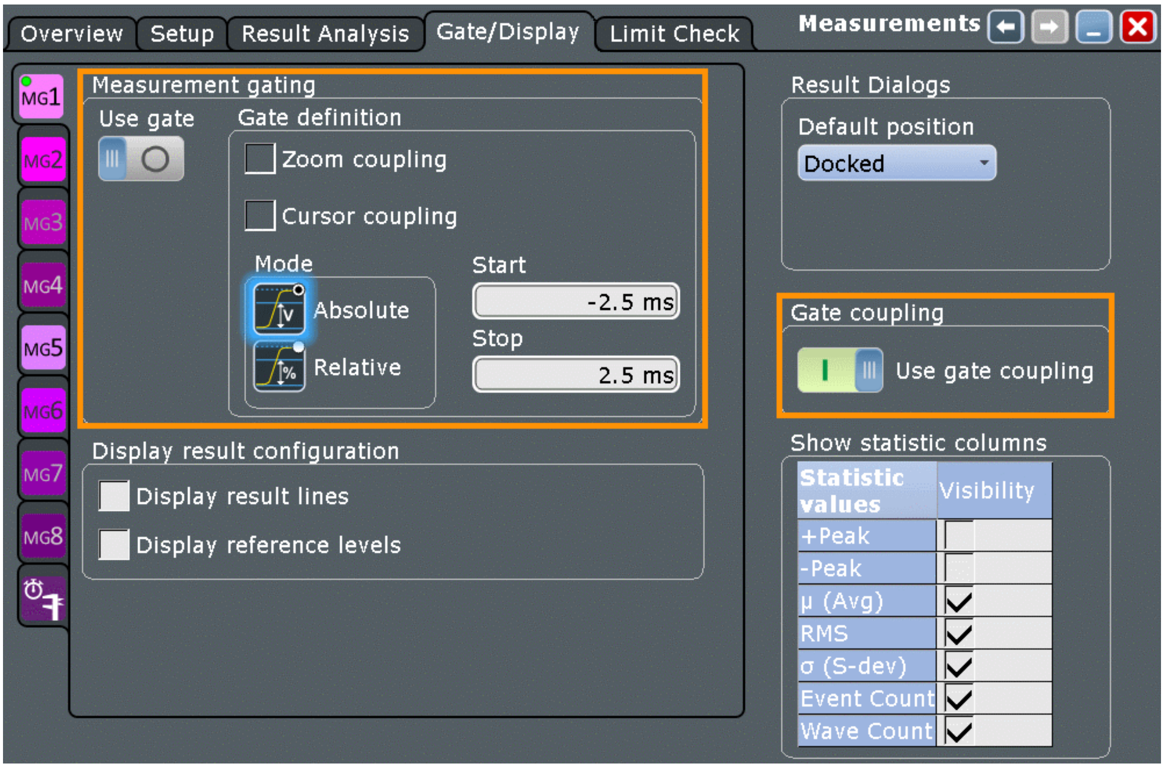
7.2.3.1 Using Measurement Gates

If you have enabled the measurement with the toolbar icon and drew a rectangle on the diagram, the gate is automatically defined and enabled. If you want to create a gate for a running measurement, or if you want to modify the gate area, configuration is done in the "Measurements" > "Gate/Display" dialog box.

1. On the "Meas" menu, tap "Gate/Display".
2. Select the subtab for the measurement you want to configure.
3. To define the gate, use one of the following methods:
   - Define the start value and the stop value of the gate area by entering either absolute or relative values.
   - If a zoom area has already been defined for the waveform, couple the gate area to the zoom area by selecting the "Zoom coupling" option.
   - If a cursor measurement has already been defined for the waveform, couple the gate area to the cursor lines by selecting the "Cursor coupling" option.
4. If you want to use the same gate for all measurement, enable "Use gate coupling".
5. Tap the "Use gate" icon to enable the gate usage.

   The measurement is performed on the selected part of the waveform. The gate is shown in the diagram.

7.2.3.2 Gate Settings for Measurements



Result display settings are described in Chapter 7.2.2.3, "Display Settings for Results", on page 301.

Use gate

Considers the gating settings for the selected measurement and displays the gate.

Remote command:
`MEASurement<m>:GATE[:STATe]` on page 1265

Zoom coupling

Zoom coupling is available if a zoom is defined. As long as "Zoom coupling" is enabled, the gate area is defined identically to the zoom area - if you change the zoom, the gate changes as well.

If several zoom diagrams are defined, select the zoom diagram to be used for gating. The "Start" and "Stop" values of the gate are adjusted accordingly.

Zoom coupling can be set for measurement gates, FFT gates, and search gates. The zoom must be defined on the diagram that contains the signal source of the measurement, FFT, or search.

Remote command:
`MEASurement<m>:GATE:ZCOupling` on page 1267
`MEASurement<m>:GATE:ZDIagram` on page 1268
`CALCulate:MATH<m>:FFT:GATE:ZCOupling` on page 1287
`SEARch:GATE:ZCOupling` on page 1344
`SEARch:GATE:ZDIagram` on page 1344

Cursor coupling
If enabled, the gate area is defined by the cursor lines of an active cursor measurement. If several cursor measurements are enabled, select the cursor set to be used for gating. The "Start" and "Stop" values of the gate are adjusted to the values of the cursor line positions. The measurement is limited to the part of the waveform between the cursor lines.

Remote command:
`MEASurement<m>:GATE:CCOupling` on page 1267
`MEASurement<m>:GATE:CURSor` on page 1267

Mode
Defines whether the gate settings are configured using absolute or relative values.

"Absolute" The gate is defined by absolute start and stop values.

"Relative" The gate's start and stop values are defined by a percentage of the value range.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:MODE` on page 1286
`MEASurement<m>:GATE:MODE` on page 1266
`SEARch:GATE:MODE` on page 1342

(Relative) Start
Defines the starting value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt` on page 1286
`CALCulate:MATH<m>:FFT:GATE:RELative:STARt` on page 1286
`MEASurement<m>:GATE:ABSolute:STARt` on page 1266
`MEASurement<m>:GATE:RELative:STARt` on page 1266
`SEARch:GATE:ABSolute:STARt` on page 1343
`SEARch:GATE:RELative:STARt` on page 1343

(Relative) Stop
Defines the end value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP` on page 1286
`CALCulate:MATH<m>:FFT:GATE:RELative:STOP` on page 1287
`MEASurement<m>:GATE:ABSolute:STOP` on page 1266
`MEASurement<m>:GATE:RELative:STOP` on page 1266
`SEARch:GATE:ABSolute:STOP` on page 1343
`SEARch:GATE:RELative:STOP` on page 1344

Use gate coupling
If you enable the gate coupling, the gate settings of the selected measurement are copied to all other measurements. Thus, all measurements use the same gate. If zoom or cursor coupling is active in a measurement, the zoom size and cursor positions are adjusted.

Remote command:
`MEASurement<m>:GATE:GCOupling` on page 1268

7.2.4 Reference Levels

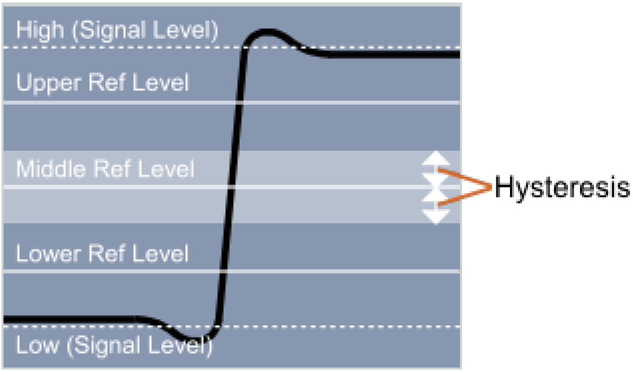
Some measurement require reference levels to obtain the measurement points, e.g. time measurements or pulse count. Reference levels are referred to the signals, for each waveform you can define specific reference levels. Thus, for all measurements on a waveform the same reference levels are used.

Usually, reference levels are determined automatically. The instrument determines the high and low signal levels based on amplitude and histogram measurements of the acquisition. The reference levels are set relatively to the determined signal levels.

However, for irregular data and in special measurement setups it may be useful to configure the levels manually:

- Data signals can contain intervals where no data is transmitted, so that a high and low state cannot be determined for each acquisition. In this case, you can define the high and low signal levels manually to evaluate other measurement results.
- If the signal levels vary strongly or have large overshoots, the rise and fall levels may be difficult to determine.
- If fixed levels are defined for the DUT, you can configure the reference levels in the R&S RTE correspondingly and analyze the resulting measurement data.

In manual configuration, the reference levels can be set relatively to defined signal levels or as absolute values.



The instrument sets a default hysteresis for the middle reference level. Hysteresis is useful for measurements that determine zero-crossings. Period, frequency, and pulse measurements are based on hysteresis - the instrument returns results if the amplitude of the signal exceeds the hysteresis. Thus, measurement during the transient oscillation is also possible.

Reference levels and result lines can be displayed in the diagram, see "To display reference levels and result lines" on page 306.

7.2.4.1 Configuring Reference Levels

To determine reference and signal levels automatically

By default, the histogram of the measurement data is evaluated to determine the required levels automatically. However, you can define several parameters to adapt the evaluation to your data.

1. On the "Meas" menu, select "Reference Levels".
2. Define the "Source", the waveform for which the reference is defined. The source can be any signal input, math or reference waveform.
3. Select automatic "Reference level mode".
4. By default, the lower reference level is defined at 10% of the signal amplitude, the middle reference level at 50% and the upper reference level at 90%. You can select other "Relative levels" to be used for evaluation.
   If default percentages do not fit, select "User defined" and enter the percentages for the upper, middle, and lower reference levels.
   The signal levels are determined by the instrument.
5. To determine the reference levels using average values from several histograms, enable the "Histogram averaging" option and define an "Average Count" to define how many histograms are averaged.

To determine reference levels manually

You can configure the reference levels manually as fixed absolute or relative values.

1. On the "Meas" menu, select "Reference Levels".
2. Define the "Source", the waveform for which the reference is defined. The source can be any signal input, math or reference waveform.
3. Select manual "Reference level mode".
4. Under "Level definition", select whether you want to define the levels using absolute or relative values.
5. If you have selected absolute level definition, enter the values of the high and low signal levels; and the distances of the reference levels to the signal levels.
6. If you have selected relative level definition, enter the values of the high and low signal levels; and the relative reference levels.

To display reference levels and result lines

1. On the "Meas" menu, select "Gate/Display".
2. Select the tab for the measurement you want to configure.

3. Enable "Display result lines" or "Display reference levels" option, or both.

   The reference levels and intermediate results are displayed in the waveform diagram.

7.2.4.2 Level Settings

Access: "Meas" menu > "Reference Level" > "Levels" tab.

On the "Levels" tab, you define how the reference levels are calculated, or you set them directly.

In automatic reference level mode, the reference levels are always relative values. You can select one of the predefined sets, or define individual percentage values.

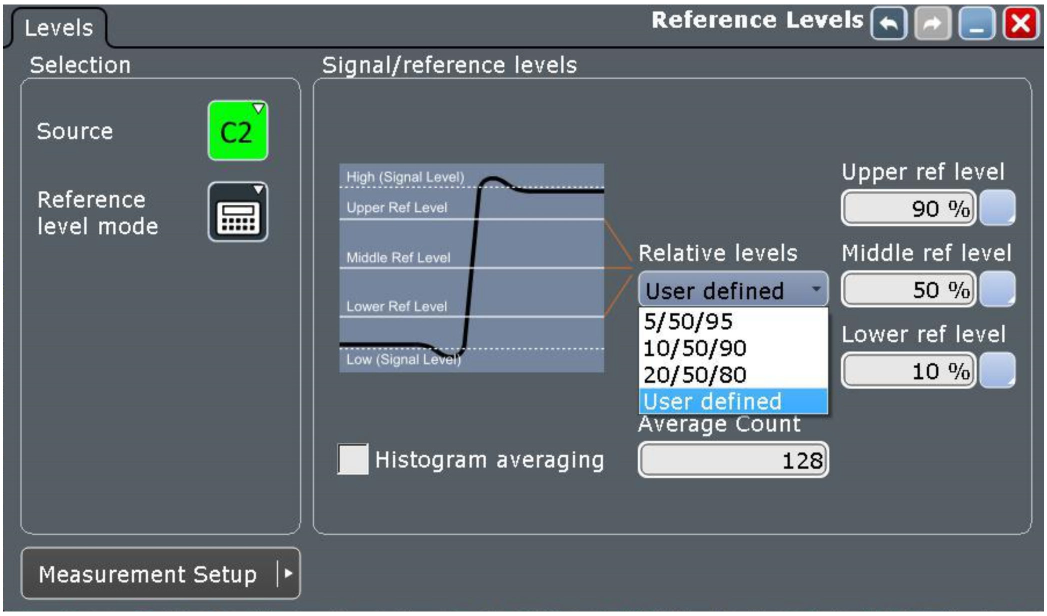


Figure 7-4: Automatic reference level definition

In manual reference level mode, relative and absolute level definitions are possible.

In manual reference level mode with relative level definition, you define the absolute values of high and low signal levels, and the reference levels as percentages of the signal amplitude.

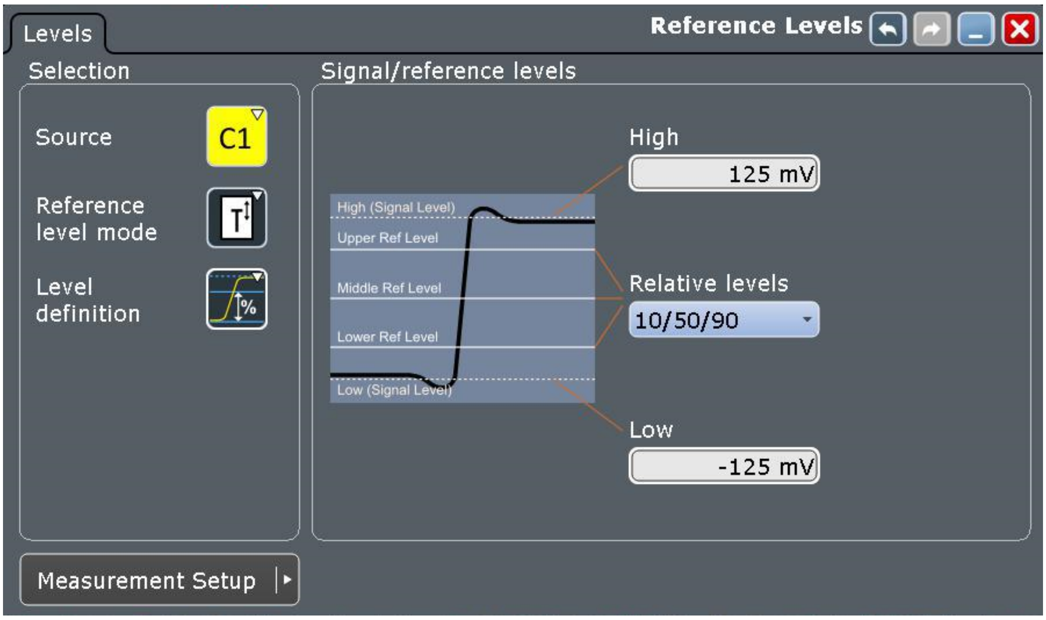


Figure 7-5: Manual reference level mode, relative level definition

In manual reference level mode with absolute level definition, you define the absolute values of high and low signal levels, and the distances between reference and signal levels.

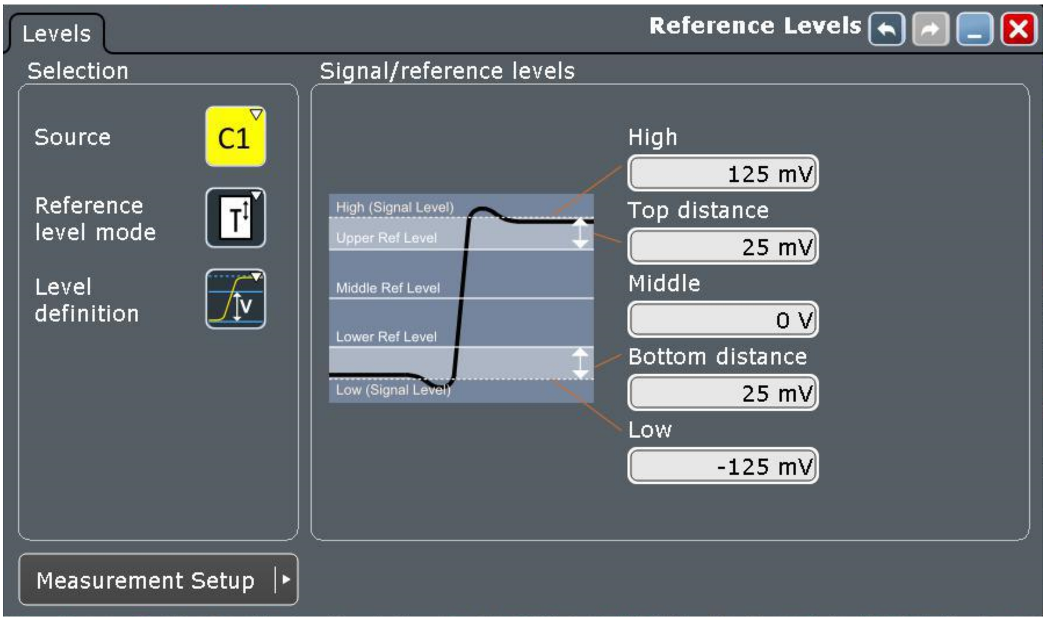


Figure 7-6: Manual reference level mode, absolute level definition

Source

Defines the source for which the reference levels are defined. The source can be any signal input, math or reference waveform.

Remote command:
Source is defined by suffix <m> in "REFLevel" subsystem, see Chapter 17.12.13, "Reference Levels", on page 1271

Reference level mode
Defines whether the reference level is configured manually or automatically.

Remote command:
`REFLevel<m>:LDETection` on page 1272

Level definition
In manual reference level mode, the setting defines whether the reference is configured using absolute or relative values.

Remote command:
`REFLevel<m>:LMODe` on page 1272

Relative levels
Sets the lower, middle and upper reference levels, defined as percentages of the signal amplitude.

Available relative levels:

- 5/50/95
- 10/50/90
- 20/50/80
- User defined: Enter "Upper ref level", "Middle ref level", and "Lower ref level".

For example, for "5/50/95" the levels are set to the following values:

- Lower reference level = 5% of the signal amplitude
- Middle reference level = 50% of the signal amplitude
- Upper reference level = 95% of the signal amplitude

Remote command:
`REFLevel<m>:RELative:MODE` on page 1272

Upper ref level, Middle ref level, Lower ref level
Define the reference levels in percent, if "Relative levels" is set to "User-defined".

Remote command:
`REFLevel<m>:RELative:UPPer` on page 1276
`REFLevel<m>:RELative:MIDDle` on page 1276
`REFLevel<m>:RELative:LOWer` on page 1277

High
Sets the high signal level.

The high signal level is set in manual reference level mode for absolute level definition.

Remote command:
`REFLevel<m>:ABSolute:HIGH` on page 1274

Low
Sets the low signal level.

The low signal level is set in manual reference level mode for absolute level definition.

Remote command:
REFLevel<m>:ABSolute:LOW on page 1274

Middle
The middle level between high and low signal level. The value is adjusted automatically if you change the high or low signal levels. Vice versa, if you change the middle level, the high and low signal levels are adjusted.

Remote command:
REFLevel<m>:ABSolute:MLEVel on page 1275

Top distance
The distance between the high signal level and the upper reference level - for manual reference level mode and absolute level definition.

Remote command:
REFLevel<m>:ABSolute:TDIStance on page 1275

Bottom distance
The distance between the lower reference level and the low signal value - for manual reference level mode and absolute level definition.

Remote command:
REFLevel<m>:ABSolute:BDIStance on page 1275

Histogram averaging
Enables averaging over several histograms to determine the reference levels.

This function is only available in automatic reference level mode.

Remote command:
REFLevel<m>:AUTO[:STATe] on page 1273

Average Count
Defines the number of histograms to calculate the average from.

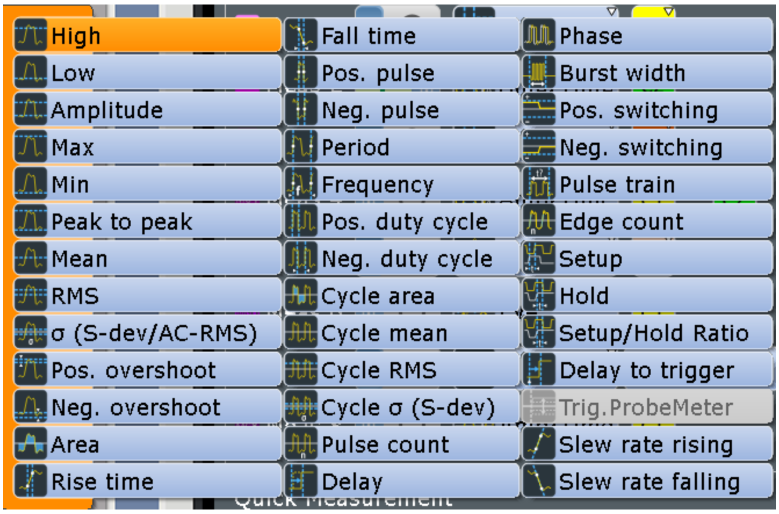
This function is only available in automatic reference level mode.

Remote command:
REFLevel<m>:AUTO:COUNt on page 1273

7.2.5 Amplitude/Time Measurements

7.2.5.1 Overview of Amplitude/Time Measurements

Access: "Meas" menu > "Setup" > "Amp/Time" category



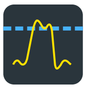
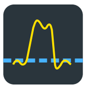
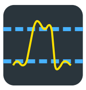
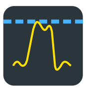
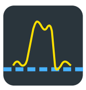
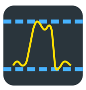
The R&S RTE provides various voltage, time, area and counting measurements in the category "Amp/Time". Some measurements require reference levels to be set according to the measurement purpose.

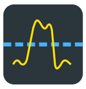
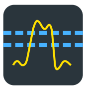
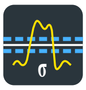
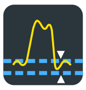
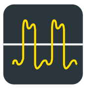
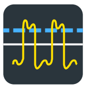
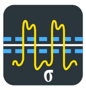
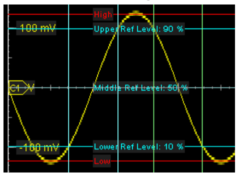
Reference levels are explained in Chapter 7.2.4, "Reference Levels", on page 305.

- Amplitude Measurements......311
- Time Measurements......312
- Area Measurements......314
- Counting......315

Amplitude Measurements

Table 7-1: Amplitude measurement types

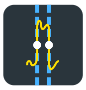
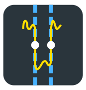
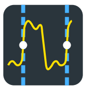
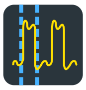
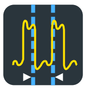
	Meas. type	Symbol	Description/Result
	High	X_{High}	High signal level
	Low	X_{Low}	Low signal level
	Amplitude	X_{Ampl}	Amplitude of the signal: the difference of high and low signal levels $X_{Ampl} = X_{High} - X_{Low}$
	Max	X_{Max}	Absolute maximum value of the waveform
	Min	X_{Min}	Absolute minimum value of the waveform
	Peak to peak	X_{PkPk}	Peak-to-peak value of the waveform: the difference of maximum and minimum values $X_{Ampl} = X_{Max} - X_{Min}$

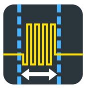
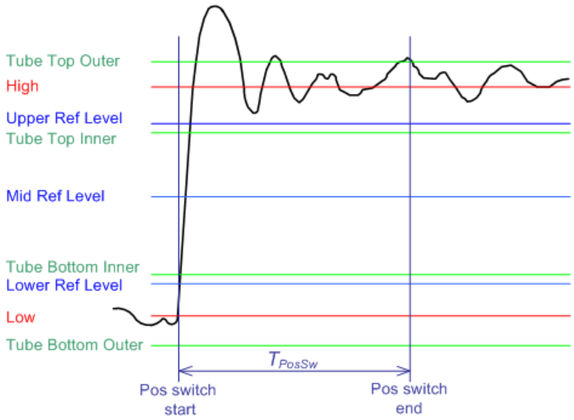
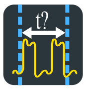
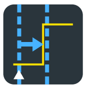
	Meas. type	Symbol	Description/Result
	Mean	X_{Mean}	Arithmetic average of the waveform voltage values $X_{Mean} = \frac{1}{N_{Eval}} \sum_{i=1}^{N_{Eval}} x(i)$
	RMS	X_{RMS}	RMS (root mean square, quadratic mean) of the waveform voltage values $X_{RMS} = \sqrt{\frac{1}{N_{Eval}} \sum_{i=1}^{N_{Eval}} x^2(i)}$
	σ (S-dev)	σ_X	Standard deviation of the waveform samples $\sigma_X = \sqrt{\frac{1}{N_{Eval}-1} \sum_{i=1}^{N_{Eval}} (x(i) - X_{Mean})^2}$
	Pos. overshoot	R_{Pos}	Positive overshoot of a square wave, calculated from measurement values High, Max, and Amplitude $+Ovr = \frac{V_{top} - V_{P+}}{V_{Amp}} \cdot 100\%$
	Neg. overshoot	R_{Neg}	Negative overshoot of a square wave, calculated from measurement values Min, Low, and Amplitude $-Ovr = \frac{V_{base} - V_{P-}}{V_{Amp}} \cdot 100\%$
	Cycle mean		The mean value of one cycle
	Cycle RMS		The RMS (root mean square) value of one cycle
	Cycle σ (S-dev)		The standard deviation of one cycle
	Trig. ProbeMeter		The DC voltage from the connected probe. Only available if an active Rohde & Schwarz probe with ProbeMeter is connected.
	Slew rate rising		Steepness of the rising edge: voltage difference between the lower and higher reference level, divided by the rise time. Result in V/s = V*Hz (blue vertical lines in the picture below).
	Slew rate falling		Steepness of the falling edge: voltage difference between the higher and lower reference level, divided by the fall time. Result in V/s = V*Hz (green vertical lines in the picture below). 

Time Measurements

See also: Chapter 7.2.4, "Reference Levels", on page 305.

Table 7-2: Time measurement types

	Meas. type	**Symbol**	**Description/Result**
	Rise time	T_{Rise}	Rise time of the left-most rising edge of the waveform. Rise time is the time it takes the signal to rise from the low reference level to the high reference level. Measurement all events in the acquisition is possible.
	Fall time	T_{Fall}	Falling time of the left-most falling edge of the waveform. Fall time is the time it takes the signal to fall from the high reference to the low reference. Measurement all events in the acquisition is possible.
	Pos. pulse	$T_{PosPulse}$	Width of a positive pulse: time between a rising edge and the following falling edge measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Measurement all events in the acquisition is possible.
	Neg. pulse	$T_{NegPulse}$	Width of a negative pulse: time between a falling edge and the following rising edge measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Measurement all events in the acquisition is possible.
	Period	T_{Period}	Time between two consecutive waveform edges of the same direction, measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Measurement all events in the acquisition is possible.
	Frequency	f_{Period}	Frequency of the signal, reciprocal value of the period. $f_{Period} = 1 / T_{Period}$
	Pos. duty cycle	R_{PosCyc}	Positive duty cycle: Width of a positive pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $$R_{PosCyc} = \frac{T_{PosPulse}}{T_{Period}} \cdot 100\%$$
	Neg. duty cycle	R_{NegCyc}	Negative duty cycle: Width of a negative pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $$R_{NegCyc} = \frac{T_{NegPulse}}{T_{Period}} \cdot 100\%$$
	Delay		Time difference between any two slopes at middle reference level. The measurement result is negative if the edge of the second source comes before the edge of the first source. See: "Settings for delay and phase measurements (analog sources)" on page 317
	Phase		Phase difference between two waveforms. *Phase = Delay / Period * 360*

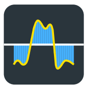
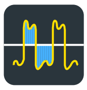
	Meas. type	Symbol	Description/Result
	Burst width		Duration of one burst, measured from the first edge to the last
	Pos. switching	T_{PosSw}	Settling time at rising edges: Time between crossing the lower reference level and the last return of the signal into the top tolerance tube. 
	Neg. switching	T_{NegSw}	Settling time at falling edges: Time between crossing the upper reference level and the last return of the signal into the bottom tolerance tube. See also "Pos. switching" above.
	Pulse train		Duration of N positive pulses, measured from the rising edge of the first pulse to the falling edge of the N-th pulse. Define N for the measurement.
	Setup Hold Setup/Hold time	T_{Setup} and T_{Hold}	Setup and Hold time measurement with positive and/or negative clock edge. See: "Setup/Hold measurement settings" on page 318
	Setup/Hold ratio	$T_{Setup} / (T_{Setup} + T_{Hold})$	Setup/Hold ratio measurement with positive and/or negative clock edge. See: "Setup/Hold measurement settings" on page 318
	Delay to trigger		Time between the trigger event and a following signal slope. High accuracy even if the trigger event is outside the acquisition data. See: ● "Delay to trigger measurement settings" on page 319 ● Chapter 7.2.5.3, "Measuring the Delay to Trigger", on page 320

Area Measurements

Access: "Meas" menu > "Setup" > "Amp/Time" category

Area measurements are voltage over time measurements.

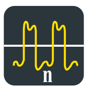
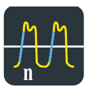
Table 7-3: Area measurement types

	Meas. type	Symbol	Description/Result
	Area	A_{Ref}	Area between the waveform and a reference level ("Area level", X_{Ref}). $A_{Ref} = \frac{T_{Eval}}{N_{Eval}} \cdot \sum_{i=1}^{N_{Eval}} (x(i) - X_{Ref})$ T_{Eval}: Evaluation time, time of a full waveform or limited by a gate
	Cycle area	A_{RefCyc}	Area between the waveform and a reference level ("Area level") measured for one period, see also "Area" measurement. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible.

Counting

Access: "Meas" menu > "Setup" > "Amp/Time" category

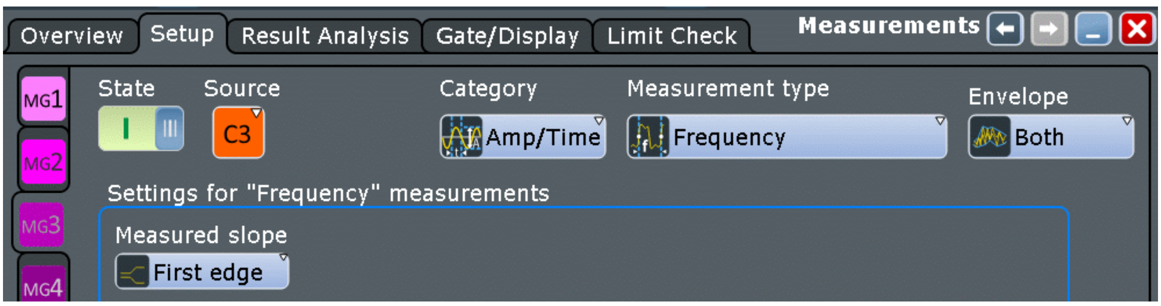
Table 7-4: Counting measurement types

	Meas. type	Symbol	Description/Result
	Pulse count		The number of positive or negative pulses of the waveform, or of both positive and negative pulses. The mean value of the signal is determined. If the signal passes the mean value, an edge is counted. A positive pulse is counted if a rising edge and a following falling edge are detected. A negative pulse is counted if a falling edge and a following rising edge are detected.
	Edge count		The number of positive or negative edges, or of both positive and negative edges. The instrument determines the mean value of the signal and counts an edge every time the signal passes the mean value.

7.2.5.2 Settings for Amplitude/Time Measurements

Access: "Meas" menu > "Setup" > "Amp/Time" category

Amplitude and time measurements are available for sources in the time domain. For some amplitude/time measurements, such as delay, setup/hold and delay to trigger, further setting are required to get a measurement result.



This chapter explains all settings for amplitude/time measurements. For a description of available measurement type, see Chapter 7.2.5.1, "Overview of Amplitude/Time Measurements", on page 310.

Envelope....316
Pulses slope....316
Measured slope....316
Settings for delay and phase measurements (analog sources)....317
Settings for delay and phase measurements (digital sources)....317
Setup/Hold measurement settings....318
Clock ref level....318
Data ref level....319
Pulse count....319
Edges slope....319
Delay to trigger measurement settings....319

Envelope

This setting is only available for measurements on envelope waveforms, see "Arithmetic" on page 144.

"Both"	The upper and the lower envelope are used in measurements. For time measurements, the averages of min and max values are used, that is, the measurement is performed on the average waveform built from the upper and lower envelope.
"Maximum"	Measurements are performed on the upper envelope.
"Minimum"	Measurements are performed on the lower envelope.

Remote command:
`MEASurement<m>:ENVSelect` on page 1222

Pulses slope

Sets the first slope of the pulses to be counted.

The setting is available only for the "Pulse count" measurement.

"Positive"	Positive pulses are counted.
"Negative"	Negative pulses are counted.
"Either"	Both positive and negative pulses are counted.

Remote command:
`MEASurement<m>:AMPTime:PSLope` on page 1224

Measured slope

Selects the slope direction for frequency and period measurements.

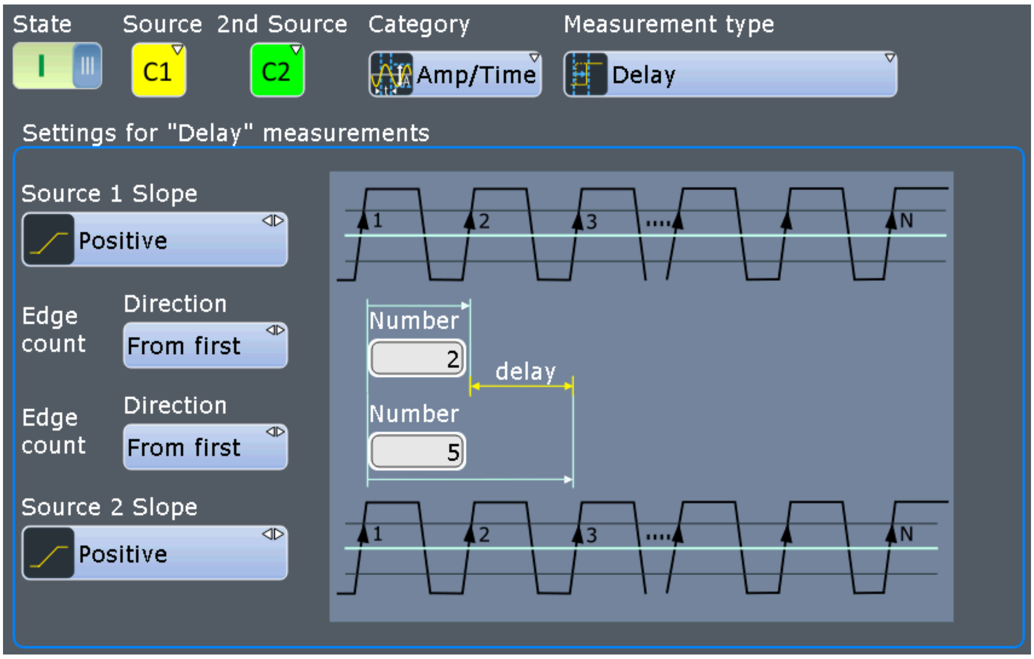
"Positive / Negative"	Measures the time between rising or falling edges, respectively.
"Either"	In multiple measurements, the time is measured both between rising edges and between falling edges. In single measurements. The first edge is taken for the measurement.

"First edge" Time is measured either between rising edges or between falling edges. The first edge is taken for the measurement. In single measurements, it works the same way as "Either".
Only available for analog measurement sources.

Remote command:
MEASurement<m>:AMPTime:PFSLope on page 1223

Settings for delay and phase measurements (analog sources)
You can measure the time between any two slopes at middle reference level. Therefore, the slopes must be defined for each source individually. The measurement result is negative if the edge of the second source comes before the edge of the first source.

With the settings shown in the picture, the time between the second and the fifth rising edge is measured.



"Slope" Sets the edge of each source, between which the delay is measured: positive, negative, or either of them.

"Direction" Selects the direction for counting slopes for each source: from the beginning of the waveform, or from the end.

"Number" Sets the number of the edge that is relevant for delay measurement.

Remote command:
MEASurement<m>:AMPTime:DELay<n>:SLOPe on page 1225
MEASurement<m>:AMPTime:DELay<n>:DIRection on page 1224
MEASurement<m>:AMPTime:DELay<n>:ECOunt on page 1225

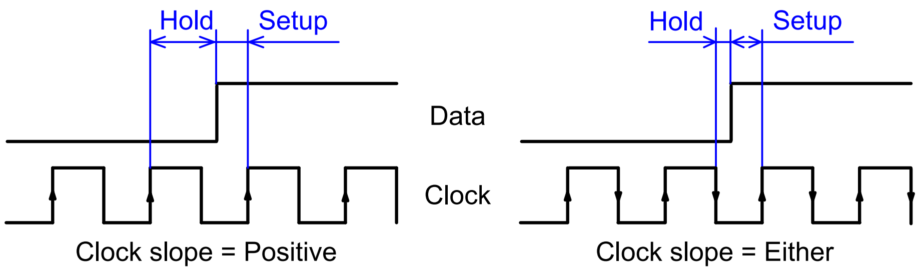
Settings for delay and phase measurements (digital sources)
Delay measurement on digital channels is reduced to measure the time between two subsequent rising or two subsequent falling edges.

The edge direction is set with Edges slope.

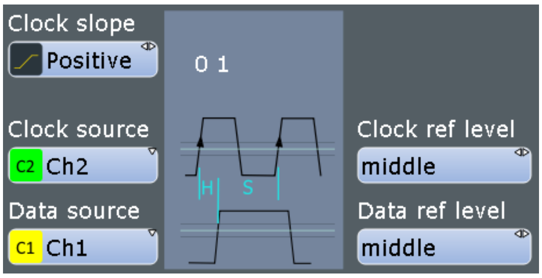
Setup/Hold measurement settings
Setup/Hold measurements analyze the relative timing between two signals: a data signal and the synchronous clock signal. Setup time is the time that the data signal is steady before clock edge - the time between a data transition and the next specified clock edge. Hold time is the time that the data signal is steady after clock edge - the time between a data transition and the previous specified clock edge.

"Setup/Hold time" measures and displays the setup and hold durations. "Setup/Hold ratio" measurements return the ratio of the setup time to the sum of hold and setup time: $T_{Setup} / (T_{Setup} + T_{Hold})$.

The clock edge can be defined, the polarity of the data signal does not matter.



If at least one of the setup/hold measurements is selected, more settings appear to specify the measurement.



"Clock slope" Sets the edge of the clock from which the setup and hold times are measured: positive, negative, or either of them. If "Either" is selected, the clock edges next to the data edge are considered regardless of the clock slope.

"Clock source" The "Clock source" is identical to the measurement "Source". It defines the waveform used as clock in the setup/hold measurement.

"Data source" The "Data source" is identical to the "2nd Source" of the measurement. It sets the data signal.

"Clock ref level" See "Clock ref level" on page 318.

"Data ref level" See "Data ref level" on page 319.

Remote command:
Clock slope: `MEASurement<m>:AMPTime:CSLope` on page 1226

Clock ref level
Selects the reference level of the clock on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

Remote command:
`MEASurement<m>:AMPTime:CLCK<n>:LSELect` on page 1227

Data ref level
Selects the reference level of the data on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

Remote command:
`MEASurement<m>:AMPTime:DATA<n>:LSELect` on page 1227

Pulse count
Sets the number N of positive pulses for the "Pulse train" measurement. This measurement measures the duration of N positive pulses from the rising edge of the first pulse to the falling edge of the N-th pulse.

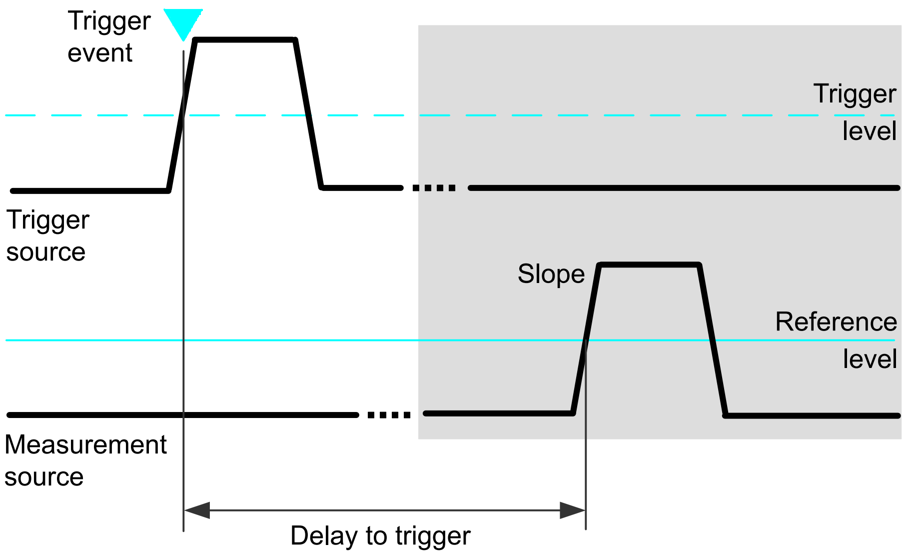
Remote command:
`MEASurement<m>:AMPTime:PTCount` on page 1226

Edges slope
Sets the edge direction to be considered. The setting is relevant for edge count measurement and delay measurement on digital channels.

"Positive"	Positive edges are considered.
"Negative"	Negative edges are considered.
"Either"	Both positive and negative edges are counted (edge count). Delay is measured either between rising edges or between falling edges. The first edge is taken for the measurement.

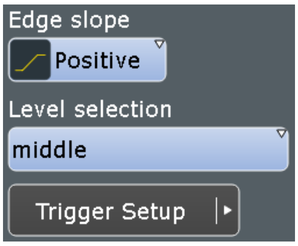
Remote command:
`MEASurement<m>:AMPTime:ESLope` on page 1226

Delay to trigger measurement settings
Delay to trigger measures the time between the trigger point and the following slope of a waveform. The delay between the trigger and the slope can be high compared to the accuracy of the acquisition, and the trigger point can even be outside of the current acquisition.



See also: Chapter 7.2.5.3, "Measuring the Delay to Trigger", on page 320.

To configure the trigger conditions, use the trigger setup. To set up the slope, additional settings appear in the measurements "Setup" dialog box.



"Edge slope" Sets the edge direction to be used for delay measurement: positive, negative, or either edge.

"Level selection" Selects the reference level of the measurement source on which the delay is measured: upper, middle, or lower level.

Remote command:
`MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe` on page 1227
`MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect` on page 1228

7.2.5.3 Measuring the Delay to Trigger

Delay to trigger measures the time between the trigger point and the following slope of a waveform. If the delay is unknown, it can be measured in two stages - first a coarse and then a precise measurement.

See also: "Delay to trigger measurement settings" on page 319.

1. Set the horizontal scale and horizontal position so that the trigger point and the slope both are visible on the screen.
2. Configure the delay to trigger measurement:

a) On the "Overview" tab, select "Delay to trigger" as measurement type.
b) Open the "Setup" tab.
c) Select the source, that is the waveform with the delayed slope.
d) Select the slope, and the reference level.
e) Check the trigger settings.

3. Enable the measurement. Note the result.
4. Turn the horizontal [POSITION / REF POINT] knob and enter the measured delay as horizontal position.

   Thus, the slope is moved to the center of the screen.
5. Adjust the horizontal scale and the horizontal resolution parameters ([RES REC LEN]) to the required accuracy: "Sample rate", "Resolution", or "Acquisition time".

   The trigger is outside the display and is not part of the current acquisition.
6. Repeat the "Delay to trigger" measurement.

   Now the delay is measured with high accuracy. You can analyze the variance of delay values using statistical evaluation and histogram functions.

7.2.6 Eye Diagram Measurements

The eye diagram is a tool for evaluation of signal quality and shows the combined effects of channel noise and intersymbol interference. It is a significant means of visualizing jitter and allows you to analyze the reasons for it. By creating histograms of the eye diagram, important jitter parameters can be determined.

The eye diagrams are a superposition of repetitively sampled waveforms, which have a length of about 1 bit.

The waveform display style must be set to vectors: [DISPLAY] > "Signal Colors / Persistence" tab > "Style = Vectors"

The following characteristic values can be determined:

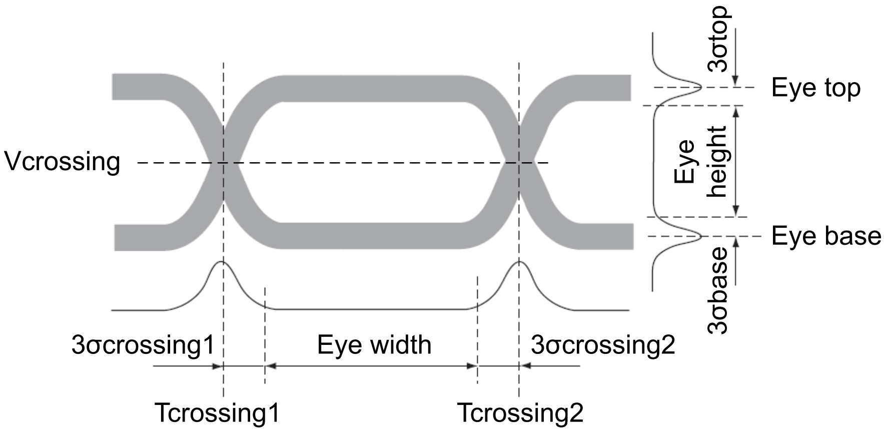


Figure 7-7: Basic eye diagram characteristics

Eye top	= Mean of the upper vertical histogram
σtop	= Standard deviation of the upper vertical histogram
Eye base	= Mean of the lower vertical histogram
σbase	= Standard deviation of the lower vertical histogram
Tcrossing	= First and second mean of the horizontal histogram
σcrossing	= Standard deviation of the horizontal histogram

7.2.6.1 Overview of Eye Diagram Measurements

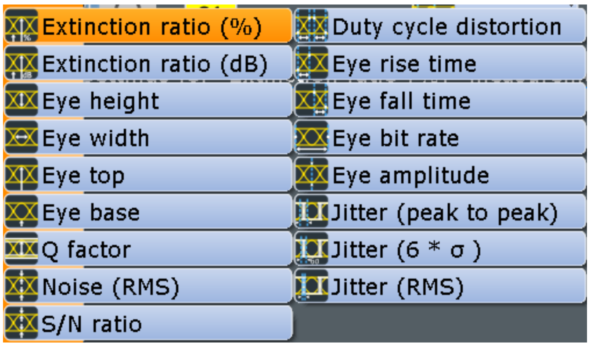
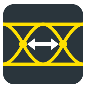
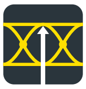
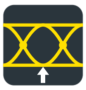
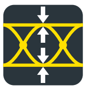
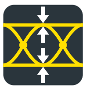
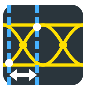
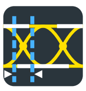
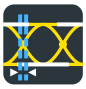


Table 7-5: Eye measurement types

	Meas. type	Description/Result
	Extinction ratio (%)	The extinction ratio is an indication of efficiency. It describes the ratio of the power used to transmit a logic level 1, to the energy used to transmit a logic level 0. The R&S RTE provides extinction ratio measurements as a percentage, and in decibels: ER (%) = Eye base / Eye top *100 Prerequisite: Eye base > 0 and Eye top > 0 because extinction ratio is defined only for positive values.
	Extinction ratio (dB)	ER (dB) = 10*log (Eye top / Eye base)

	Meas. type	Description/Result
	Eye height	The vertical eye opening indicates the sensitivity of the transmission to noise. (Eye top – 3 * σtop) – (Eye base + 3 * σbase)
	Eye width	The horizontal eye opening indicates the time range during which the sampling of the logical state is possible. (Tcrossing2 – 3 * σcrossing2) – (Tcrossing 1 – 3 * σcrossing1)
	Eye top	Mean of the upper vertical histogram
	Eye base	Mean of the lower vertical histogram
	Q factor	Q factor = (Eye top – Eye base) / (σtop + σbase)
	Noise (RMS)	Quadratic mean of the noise of eye top and eye base
	S/N ratio	Signal-to-noise ratio SNR = 10 * log (Eye amplitude / Noise RMS)
	Duty cycle distortion	Duty cycle distortion = 20 * log (Eye amplitude / Noise RMS)
	Eye rise time	Duration for signal to rise from 10% to 90% of the high signal level
	Eye fall time	Duration for signal to fall from 90% to 10% of the high signal level
	Eye bit rate	Frequency between two crossings
	Eye amplitude	Eye top - Eye base
	Jitter (peak to peak)	Average of the jitter for both crossing points. Jitter = (σcrossing1 + σcrossing2) / 2

	Meas. type	Description/Result
	Jitter (6*σ)	Jitter (6*σ) = Jitter * 6
	Jitter (RMS)	Quadratic mean of the jitter at both crossing points

7.2.6.2 Settings for Eye Diagram Measurements

Access: "Meas" menu > "Setup" > "Eye" category

Eye diagram measurements are only available for sources in the time domain.

For eye measurements, no further settings are required.

7.2.7 Spectrum Measurements

Spectrum analysis determines the frequencies of a given input signal over time. Various measurements can then be performed based on the signal spectrum.

7.2.7.1 Overview of Spectrum Measurements

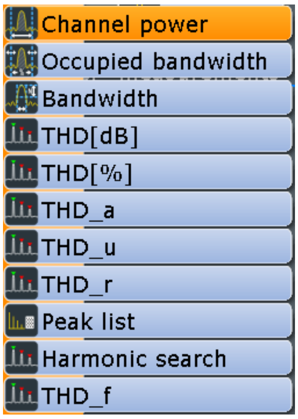
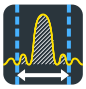
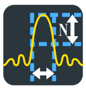


Table 7-6: Spectrum measurement types

	Meas. type	Description, result
	Channel power	Power integrated over the sample values defined by a center frequency and a bandwidth; based on a defined impedance. The result is given in dBm. To get best results, enable "Measure all events in each acquisition" on page 351, and set the "Limit" to the maximum value. In particular, these settings are important when measuring pulsed signals.
	Occupied bandwidth	From the defined center frequency, symmetric sample value pairs to the left and right are integrated until a user-defined percentage of the total power is reached. The occupied bandwidth is the difference between the frequencies at which the requested power was reached.
	Bandwidth	n dB down bandwidth; the samples to the left and right of the peak value are analyzed until the n dB threshold is exceeded. The frequencies at which the threshold is exceeded define the limits of the requested bandwidth.
	THD[dB], THD[%] Total harmonic distortion	Power sum of the harmonic waves divided by the power of the fundamental wave: $THD = \frac{\sum_{n=2}^{\infty} P_n}{P_1}$
	THD_f, THD_a, THD_u, THD_r Total harmonic distortion	These measurements require option R&S RTE-K18 Spectrum Analysis. THD_f is the root mean square of the sum of all amplitudes of the harmonic waves in relation to the amplitude at the fundamental frequency (first harmonic): $THD_F = \frac{\sqrt{\sum_{i=2}^{n} U_i^2}}{U_1}$ THD_a corresponds to THD[dB]: $THD_a = \frac{\sum_{i=2}^{n} U_i^2}{U_1^2}$ THD_u: $THD_u = \frac{\sqrt{U^2 - U_1^2}}{U_1}$ Distortion factor: $THD_R = \frac{\sqrt{U^2 - U_1^2}}{U}$ Where: • U_i: effective value of the harmonic with index i • U_1: effective value of the first harmonic • U: effective value of the signal

	Meas. type	Description, result
	Peak list	This measurement requires option R&S RTE-K18 Spectrum Analysis. Table with measured peaks. For each peak, the frequency and the power value are listed in a table row. The number of determined peaks can be defined. You can sort the results by frequency or power value, and the peak labels are adjusted accordingly.
	Harmonic search	Table with measured harmonics. For each harmonic, the frequency and the value are listed in a table row.

For remote command parameters and suffix types, see Table 17-7.

7.2.7.2 Settings for Spectrum Measurements

Access: "Meas" menu > "Setup" > "Spectrum" category

Spectrum measurements require a source in the frequency domain, i.e. a math waveform with an FFT operation.

For spectrum measurements, make sure that the start frequency and other FFT parameters are set correctly, and the fundamentals are not covered by the DC component of the signal. Consider also a gated measurement if the instrument cannot return any result.

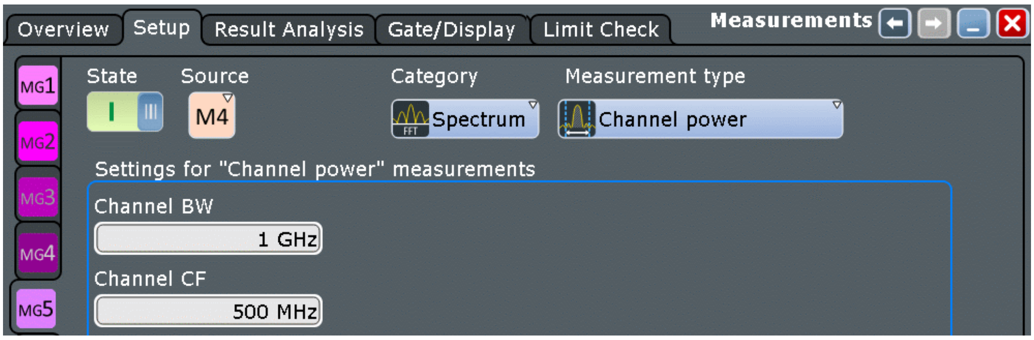


Figure 7-8: Parameters of channel power measurement

N db down

The threshold until which the samples to the left and right of the peak value are analyzed to determine the "Bandwidth".

Remote command:
`MEASurement<m>:SPECtrum:NDBDown` on page 1234

Channel BW

Bandwidth over which the channel power is calculated.

Remote command:
`MEASurement<m>:SPECtrum:CPOWer:BANDwidth` on page 1233

Channel CF
Center frequency from which the channel power is calculated over the specified bandwidth.

Remote command:
MEASurement<m>:SPECtrum:CPOWer:CFRequency on page 1234

Occup. BW
Percentage of the total power used to determine the occupied bandwidth.

Remote command:
MEASurement<m>:SPECtrum:OBANdwidth on page 1234

Threshold
Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

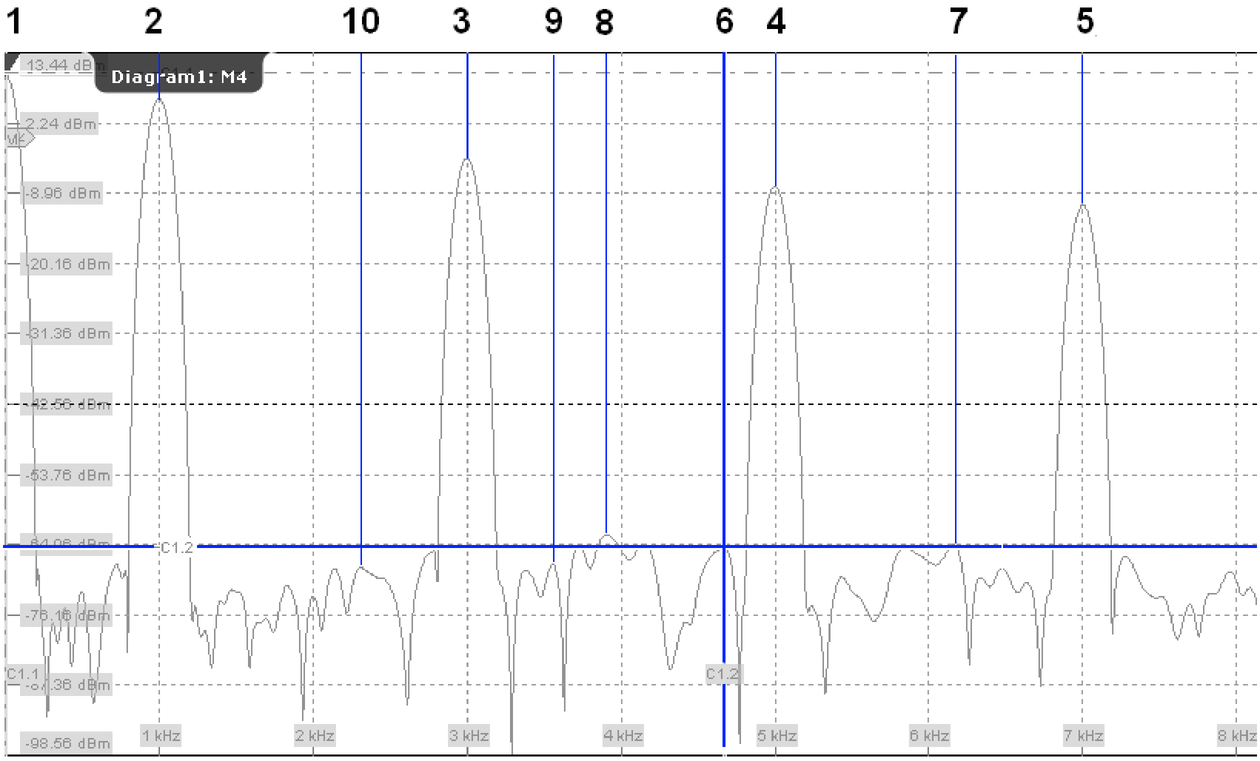
This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

Remote command:
CURSor<m>:THReshold on page 1211
MEASurement<m>:SPECtrum:ATHReshold on page 1235

Peak excursion
Defines a relative threshold, the minimum level value by which the waveform must rise or fall to be considered as a peak. To avoid identifying noise peaks, enter a peak excursion value that is higher than the noise levels.

This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

The following figure shows a cursor measurement on a spectrum waveform:



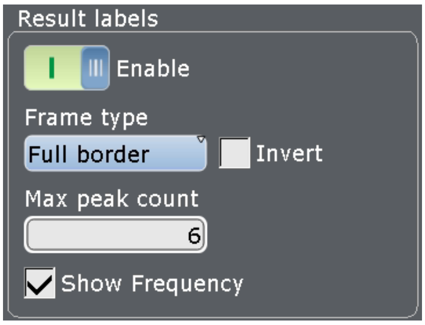
If "Peak excursion" is 30 dB, the peaks 1 to 5 are found. If "Peak excursion" is 20 dB, also the peaks 6 to 10 are found. The cursor position is on peak 6.

Remote command:
`CURSor<m>:PEXCursion` on page 1212
`MEASurement<m>:SPECtrum:PEXCursion` on page 1235

Max results
Sets the maximum number of measurement results that are listed in the result table. Available for peak list and harmonic search measurements.

Remote command:
`MEASurement<m>:SPECtrum:RESult<n>:COUNt` on page 1235

Result labels
For peak lists only, requires option R&S RTE-K18: Use labels to describe the detected peaks in the spectrum diagram. You can configure the look of the labels.



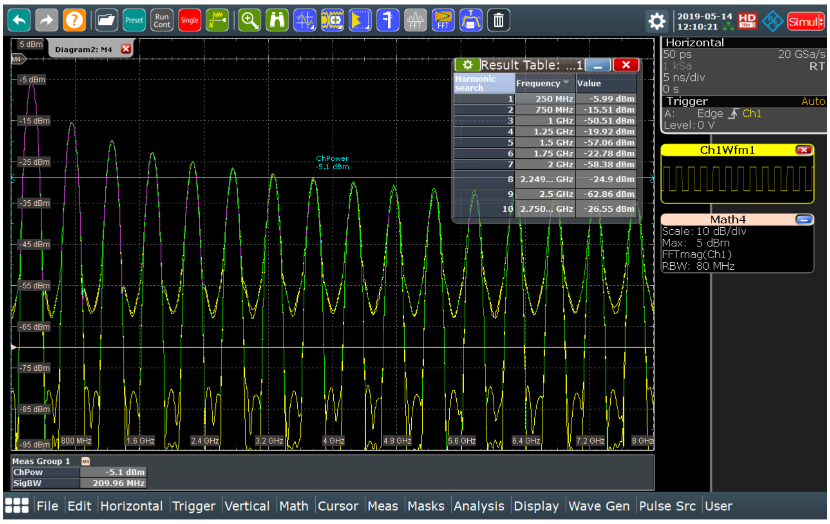
"Enable"	Displays a description for each detected peak in the spectrum diagram.
"Frame type"	Defines the layout of the labels (full border, underline, or none).
"Invert"	Displays black font on white background using the "Full frame" label type.
"Max. peak count"	Defines the maximum number of peaks that are labeled in the diagram. The result table lists all peaks.
"Show Frequency"	Includes the frequency of the detected peak in the diagram labels.

Remote command:
MEASurement<m>:RESult:SHLabels on page 1238
MEASurement<m>:RESult:LABorder on page 1237
MEASurement<m>:RESult:INVerse on page 1237
MEASurement<m>:RESult:MAXCount on page 1236
MEASurement<m>:RESult:SHFRequency on page 1238

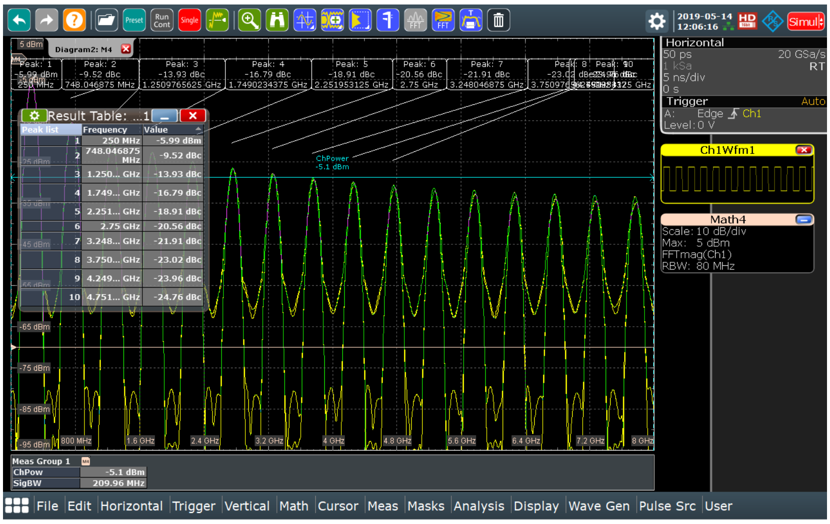
Result mode

Selects the the way the measurement results are displayed. Available for peak list and harmonic search measurements.

"Absolute" The harmonics/peaks are shown in absolute value, dBm.



"Relative" The level of the carrier is shown in absolute value dBm. The values the harmonics/peaks are shown relatively to the carrier in dBc.



Remote command:

MEASurement<m>:SPECtrum:RESult<n>:MODE on page 1236

7.2.8 Histograms and Histogram Measurements

7.2.8.1 Histogram Characteristics

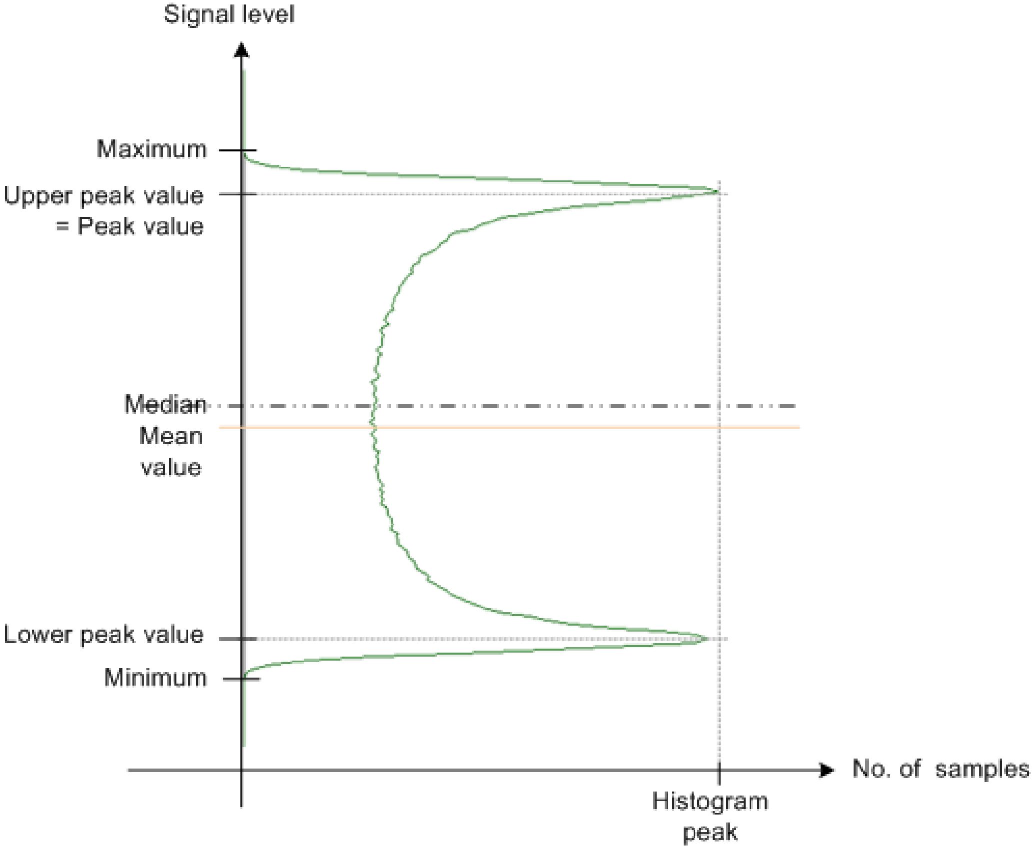
Histograms are used to plot density of data, i.e. to display graphically how often which signal values occur. The histogram can be based on the input signal levels (amplitudes) or the timebase in a time domain measurement, or on frequencies or frequency levels in a spectrum measurement. They are a prerequisite for histogram measurements.

Depending on which data the histogram is based on, a vertical or horizontal histogram can be selected. A vertical, or amplitude, histogram displays horizontal bars across amplitude values. A horizontal or time/frequency histogram displays vertical bars over time/frequencies.

You can define up to 8 histograms in a diagram, one of them is displayed. They can be created quickly using toolbar icons, or in the "Meas" menu > "Histogram" dialog box. To switch the histogram display, tap the required histogram area, or select it in the "Histogram" dialog box. For histogram measurements, the measured histogram is selected independently in the measurement setup.

In a histogram, the maximum count of a waveform value is assigned to the full height (histogram peak). All other count values are displayed relative to the maximum.

The following characteristic values can be determined for histograms (illustrated for a vertical histogram):



7.2.8.2 Overview of Histogram Measurements

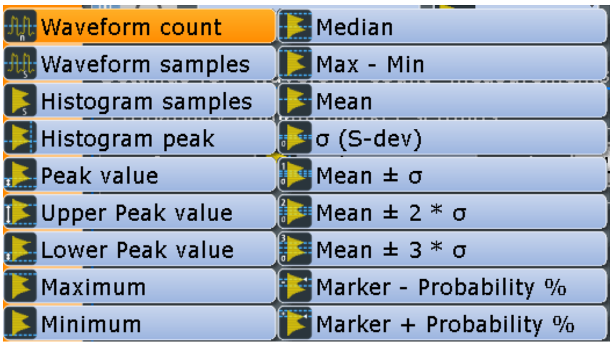
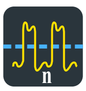
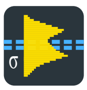


Table 7-7: Histogram measurement types

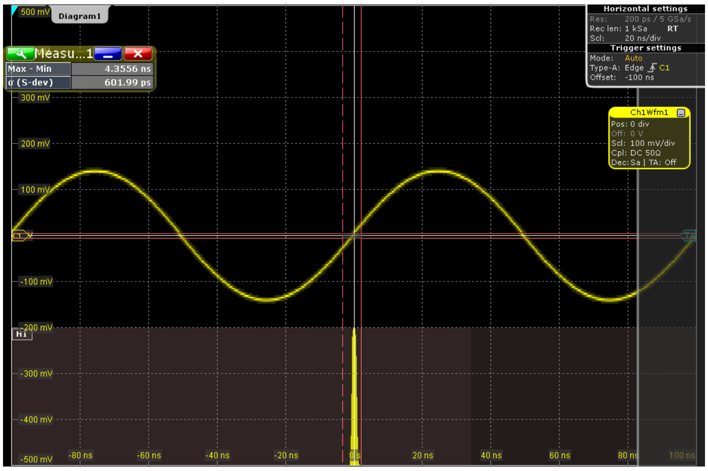
	Meas. type	Description/Result
	Waveform count	The number of acquisitions (waveforms) the histogram is based on
	Waveform samples	The number of samples from the most recent acquisition included in the current histogram

	Meas. type	Description/Result
	Histogram samples	The number of samples from all acquisitions included in the current histogram
	Histogram peak	The maximum count value in the histogram
	Peak value	The signal value at the histogram peak
	Upper peak value	The signal value at the maximum count value in the upper half of the histogram
	Lower peak value	The signal value at the maximum count value in the lower half of the histogram
	Maximum	The highest signal value with a probability > 0
	Minimum	The lowest signal value with a probability > 0
	Median	The signal value for which half the samples lie above, the other half below in the histogram The sample count of one signal value after the other are accumulated until half the total number of samples in the histogram is reached. The signal value for which 50% of the samples are accumulated is the median.
	Max - Min	The range of signal values with a probability > 0
	Mean	The weighted arithmetic average of the histogram
	σ (S-dev)	Standard deviation of the sample numbers
	Mean ±σ	The range between (mean value + standard deviation) and (mean value - standard deviation)
	Mean ±2*σ	The range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)
	Mean ±3*σ	The range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)

	Meas. type	Description/Result
	Marker + Probability %	The marker value (according to the selected probability domain marker type) plus the defined limit. Note that the value is restricted to the histogram range.
	Marker - Probability %	The marker value (according to the selected probability domain marker type) minus the defined limit. Note that the value is restricted to the histogram range.

Rough jitter evaluation using a histogram

You can use a horizontal histogram to perform a rough jitter measurement. Define a histogram for a narrow amplitude range close to the trigger time. The "Max-Min" value indicates the peak jitter, while the "StdDev" value indicates the RMS jitter.



In addition to histograms on channel, math and reference waveforms, histograms can be created based on statistic measurement results. These histograms are enabled in the "Result Analysis" tab, see Chapter 7.2.10, "Result Analysis", on page 343.

7.2.8.3 Creating Histograms

Histograms can be used to evaluate the sample value occurrences directly. They are a prerequisite for histogram measurements.

To create a histogram quickly with toolbar icons

1. Select the waveform for which you need a histogram.
2. Touch the histogram icon on the toolbar, and drag your finger down. Tap the "Vertical histogram" or the "Horizontal histogram" icon.

3. Tap the diagram with the waveform to be measured, or draw a rectangle on the screen to define the area for histogram calculation.

   The histogram range is indicated in the diagram and a histogram with the selected waveform as a source is defined and displayed.

To create and configure a histogram in the dialog box

1. Select "Meas" > "Histogram", or touch and hold an existing histogram or histogram area.

   The "Histogram Setup" dialog box is displayed.

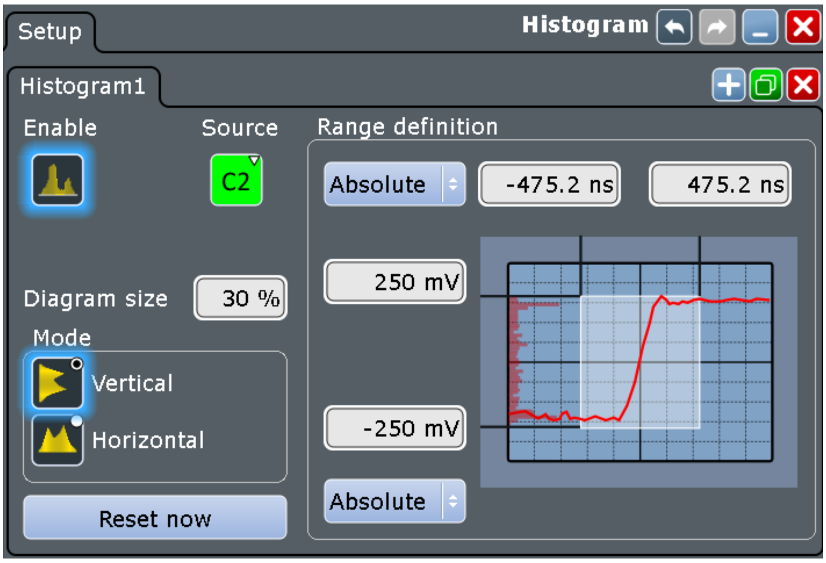
2. To create a histogram, tap the "Add" icon in the upper right corner of the dialog box.

3. To copy an existing histogram and configure a new one based on those settings, tap the "Copy" icon.
4. To change the name of a histogram, double-tap the tab label. Enter a name for the histogram using the on-screen keyboard.
5. Select a "Source" for the histogram. The source can be any input signal, math or reference waveform.
6. Define the histogram "Mode": vertical for an amplitude, horizontal for a time-based histogram.
7. Define the range of the waveform for which the histogram is to be generated. Enter the start value and the stop value in x and in y direction, either as absolute or relative values.
8. Enable the histogram.

7.2.8.4 Histogram Setup

Access: "Meas" menu > "Histogram"

In this dialog box, you configure histograms on which you can perform further measurements.



Enable
Enables or disables the histogram evaluation and display. The histogram settings are kept until the histogram is deleted.

Source
Defines the source of the histogram. Any analog channel waveform, math or reference waveform can be selected. Also measurements can serve as histogram source. In this case, the density distribution of the results of the main measurement is displayed.

Remote command:
`LAYout:HISTogram:SOURce` on page 1241

Diagram size
Defines the size of the histogram in percent of the diagram.

Mode
Defines the type of histogram.

"Vertical" Amplitude histogram (horizontal bars across amplitude)

"Horizontal" Time histogram (vertical bars over time). For spectrum waveforms, horizontal histograms over spectrum are not available.

Remote command:
`LAYout:HISTogram:MODE` on page 1242

Reset now
Resets the values to begin a new histogram.

Remote command:
`LAYout:HISTogram:RESet` on page 1245

Range definition mode (Absolute/Relative)
Defines whether the value range limits are entered as absolute or relative values.

Remote command:
LAYout:HISTogram:HORZ:MODE on page 1242
LAYout:HISTogram:VERTical:MODE on page 1244

Horizontal start/stop value
Defines the horizontal value range of the histogram.

Remote command:
LAYout:HISTogram:HORZ:ABSolute:STARt on page 1243
LAYout:HISTogram:HORZ:ABSolute:STOP on page 1243
LAYout:HISTogram:HORZ:RELative:STARt on page 1243
LAYout:HISTogram:HORZ:RELative:STOP on page 1243

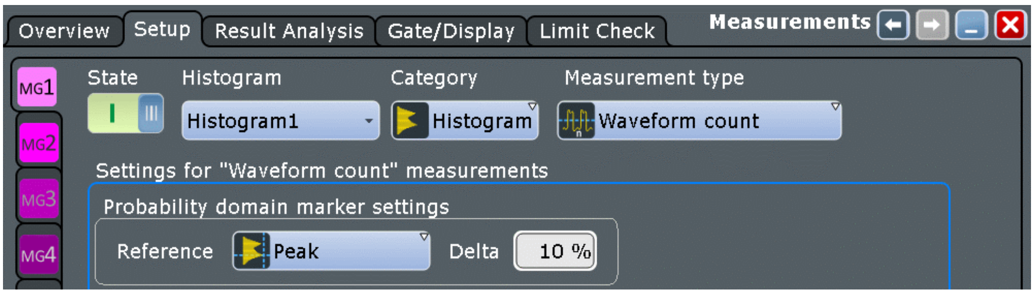
Vertical start/stop value
Defines the vertical value range of the histogram.

Remote command:
LAYout:HISTogram:VERTical:ABSolute:STARt on page 1244
LAYout:HISTogram:VERTical:ABSolute:STOP on page 1244
LAYout:HISTogram:VERTical:RELative:STARt on page 1245
LAYout:HISTogram:VERTical:RELative:STOP on page 1245

7.2.8.5 Settings for Histogram Measurement

Access: "Meas" menu > "Setup" > "Hist" category

You can perform measurements on histograms. Before, you have to define a histogram, see Chapter 7.2.8.3, "Creating Histograms ", on page 333.



The measurement parameters are common for all histogram measurement types.

Histogram
Selects the histogram on which the measurement is based. Histograms are defined via the "Meas > Histogram" menu item.

Remote command:
MEASurement<m>:HISTogram:SELect on page 1247

Probability domain marker reference
Defines the marker reference in the probability domain.

"Peak" The y-value with the maximum sample value in the histogram

"Upper Peak"	The y-value at the maximum sample value in the upper half of the histogram
"Lower Peak"	The y-value at the maximum sample value in the lower half of the histogram
"Maximum"	The highest y-value with a probability > 0
"Minimum"	The lowest y-value with a probability > 0
"Median"	The y-value for which half the samples lie above, the other half below in the histogram.
"Mean"	The weighted arithmetic average of the histogram

Remote command:
`MEASurement<m>:HISTogram:PROBability:TYPE` on page 1247

Delta
Defines a range around the marker.

Remote command:
`MEASurement<m>:HISTogram:PROBability:LIMit` on page 1248

7.2.9 Protocol Measurements (Option R&S RTE-K35)

Option R&S RTE-K35 provides specific measurements on automotive and Ethernet protocols.

7.2.9.1 Overview of Protocol Measurements

If option R&S RTE-K35 is installed, enhanced measurements on the supported serial buses are possible. With it, you can measure various dependencies between the frames.

Additional to option R&S RTE-K35 you need one of the following serial protocol options:

- I2C (R&S RTE-K1)
- SPI (R&S RTE-K1)
- UART/RS232 (R&S RTE-K2)
- CAN / CAN-FD (R&S RTE-K3/R&S RTE-K9)
- LIN (R&S RTE-K3)
- RFFE (R&S RTE-K40)
- SENT (R&S RTE-K10)
- Ethernet (100BASE-Tx) (R&S RTE-K8)
- 100BASE-T1 (R&S RTE-K57)

Table 7-8: Protocol measurement types

	Meas. type	Description / result
	Field value	Value of a field over time
	Frame to frame	The distance between two frames
	Trigger to frame	The distance between a defined frame and the next trigger signal
	Gap	Measures a gap, period at which the bus is idle. The distance of a gap can only be measured between two identified frames.
	Main bit rate	Bit rate as defined by the standard
	2nd bit rate	Additional bit rate, protocol dependent For example: "Data rate" for the CAN-FD protocol.
	Bus idle	Calculates the bus idle time
	Frame count	Number of all frames within the acquisition window
	Frame error count	Sum of all frames with errors within the acquisition window
	Frame error rate	Sum of all frames with errors divided by all frames within the acquisition window
	Consecutive frame error rate	Measures the rate at which at least two consecutive frames have an error

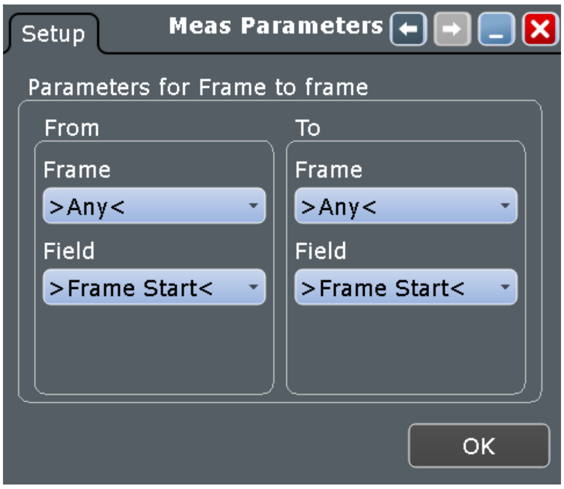
7.2.9.2 Settings for Protocol Measurements

Access: "Meas" menu > "Setup" > "Protocol" category

Protocol measurements require that a supported protocol is enabled first.

Frame to frame

Opens a dialog to set up the parameters for the frame to frame measurement. You can define a start frame ("From") and an end frame ("To") for the measurement.



The types of available frames and fields are depending on the enabled protocol.

Table 7-9: Available protocol frames

Protocol	Frame	Field
I2C	W R	Address 2. Addr ACK R/W Data
SPI	MISO MOSI	Word
CAN/CAN FD	Remote Data	ID FDF BRS ESI DLC SCV P SC Data CRC ACK
	Overload	-
LIN	WAKE	-
	Data	ID Data
RFFE	Register 0 Write Register Write Register Read Extended Register Write Extended Register Read Extended Register Write Long Extended Register Read Long Interrupt Summary and Identification Masked Write Master Ownership Handover Master Write Master Read Master Context Transfer Write Master Context Transfer Read	SA MID BC Addr Mask Data

Protocol	Frame	Field
Ethernet	MAC Sleep EOS	Preamble SFD Dest Src Len/Type Data FCS
	Idle	Triplet
100BASE-T1	MAC Fill	Preamble SFD Dest Src Len/Type Data FCS
	Idle	Triplet

Also, if a label list is loaded and enabled for the protocol, the symbolic label names can be selected from the "Frame"/"Field" value list.

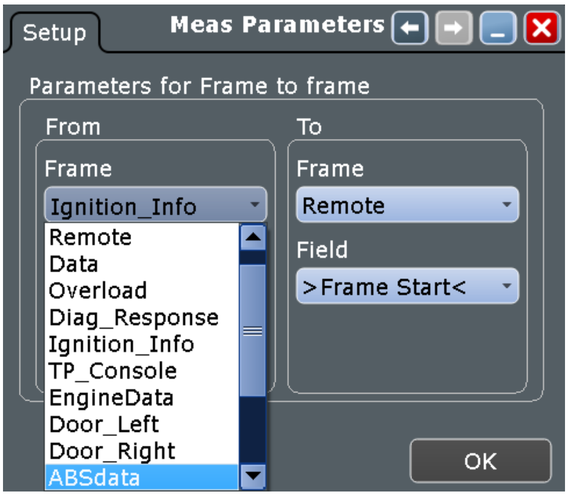


Figure 7-9: Frame list for the CAN protocol with applied label list

"Frame" Selects the type of start frame (from) / end frame (to).

"Field" Selects the type of start field (from) / end field (to).

"Value" Sets a value for the selected field.

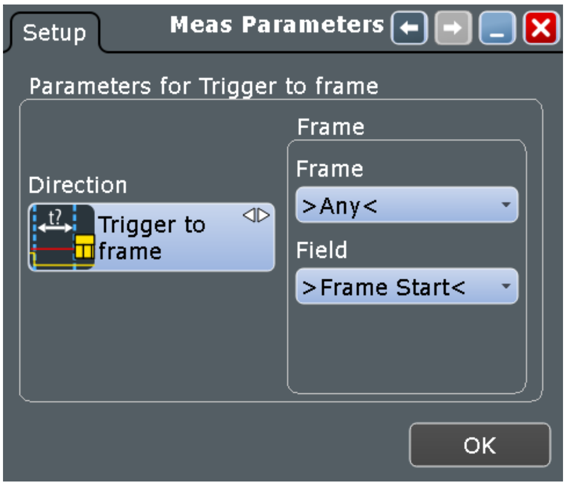
Remote command:
MEASurement<m>:PROTocol:F2FRame:FRMFrom on page 1260
MEASurement<m>:PROTocol:F2FRame:FRMTo on page 1260
MEASurement<m>:PROTocol:F2FRame:FLDFrom on page 1259
MEASurement<m>:PROTocol:F2FRame:FLDTo on page 1259

MEASurement<m>:PROTocol:F2FRame:VALFrom on page 1260
MEASurement<m>:PROTocol:F2FRame:VALTo on page 1260

Trigger to frame

Opens a dialog to set up the parameters for the trigger to frame measurement.

The types of available frames are depending on the enabled protocol, see Table 7-9.



"Direction"	Selects the direction for the measurement, from the trigger to the next frame (start),
"Frame"	Selects the type of frame.
"Field"	Selects the type of field.
"Value"	Sets a value for the selected field.

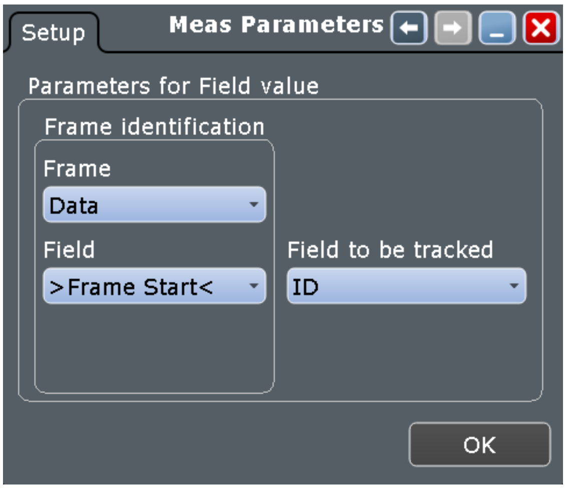
Remote command:
MEASurement<m>:PROTocol:T2FRame:DIRection on page 1263
MEASurement<m>:PROTocol:T2FRame:FLD on page 1263
MEASurement<m>:PROTocol:T2FRame:FRM on page 1263
MEASurement<m>:PROTocol:T2FRame:VALue on page 1264

Field value

Opens a dialog to set up the parameters for the field value measurement. You can first identify a frame with specific field value and then track a selected field of this frame.

The types of available frames and fields depend on the enabled protocol, see Table 7-9.

If a label list is loaded and enabled for the protocol, sometimes there are defined calculations and formatting for the displayed field value. In this case, the measurement displays the value according to the definition in the label list.



"Frame" Selects the type of frame.

"Field" Selects the type of field.

"Value" Sets a value for the selected field.

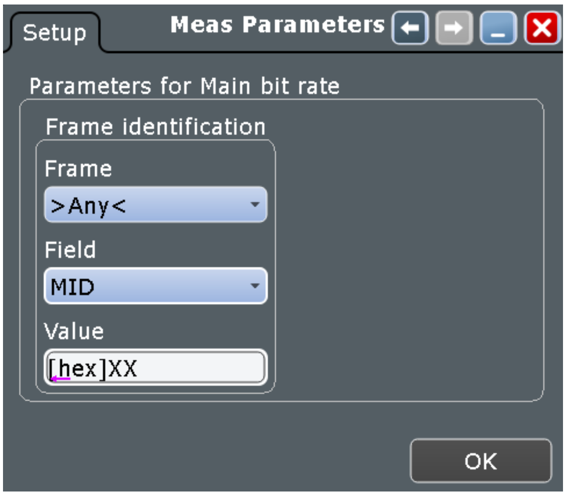
"Field to be tracked" Selects the type of field which value is tracked over time.

Remote command:
MEASurement<m>:PROTocol:FLDValue:FLD on page 1261
MEASurement<m>:PROTocol:FLDValue:FRM on page 1261
MEASurement<m>:PROTocol:FLDValue:TRCK on page 1261
MEASurement<m>:PROTocol:FLDValue:VAL on page 1261

Main bit rate / 2nd bit rate
Opens a dialog to set up the parameters for the main bit rate/ 2nd bit rate measurement. You can identify a frame and a specific field value.

The types of available frames and fields depend on the enabled protocol, see Table 7-9.

If a label list is loaded and enabled for the protocol, sometimes there are defined calculations and formatting for the displayed field value. In this case, the measurement displays the value according to the definition in the label list.



"Frame" Selects the type of frame.

"Field" Selects the type of field.

"Value" Sets a value for the selected field.

Remote command:
`MEASurement<m>:PROTocol:MBITrate:FLD` on page 1262
`MEASurement<m>:PROTocol:MBITrate:FRM` on page 1262
`MEASurement<m>:PROTocol:MBITrate:VAL` on page 1262
`MEASurement<m>:PROTocol:SBITrate:FLD` on page 1262
`MEASurement<m>:PROTocol:SBITrate:FRM` on page 1262
`MEASurement<m>:PROTocol:SBITrate:VAL` on page 1263

7.2.10 Result Analysis

The behavior of measurement results over time can be evaluated in different ways:

- Statistics
- Long-term measurements
- Histograms on measurement results
- Track

7.2.10.1 Statistics

Statistics are compiled for all measurements. If enabled, statistical results of all active measurements are shown in the result box.

By default, all events in each acquisition are measured and incuded in the statitics: the measurement result is not only determined once within one acquisition, but repeatedly, if available. More results provide a larger basis for statistical evaluation.

To obtain precise results, additional measurement settings can be useful:

- Reference/signal levels: configuring user-defined levels can compensate for irregular data, see Chapter 7.2.4.1, "Configuring Reference Levels", on page 306.

- Gate areas: restricting the waveform range for measurement can eliminate irregular data, see Chapter 7.2.3, "Measurement Gates", on page 302.
- Spectrum measurements: you can eliminate noise from the evaluation, see Threshold and "Peak excursion" on page 293

► To enable statistics, use one of the following ways.
  - On the sidebar, enable "Statistics".
  - On the "Meas" > "Result Analysis" tab, "Enable" statistics.

  From the "Meas" menu, select "Setup".

If statistics are enabled, the following results are calculated. You can select which results are provided in the result box, see Chapter 7.2.2.3, "Display Settings for Results", on page 301.

Label	Description
+Peak	Positive peak value (maximum)
-Peak	Negative peak value (minimum)
μ (Avg)	Average
RMS	Root mean square
σ (S-dev)	Standard deviation
Event count	Number of measured events (e.g. rising edges, pulses etc.)
Wave count	Number of waveforms (acquisitions) the measurement is based on

Remote commands:
- `MEASurement<m>:RESult[:ACTual]?` on page 1218
- `MEASurement<m>:RESult:AVG?` on page 1218
- `MEASurement<m>:RESult:COUNt?` on page 1219
- `MEASurement<m>:RESult:EVTCount?` on page 1218
- `MEASurement<m>:RESult:NPEak?` on page 1218
- `MEASurement<m>:RESult:PPEak?` on page 1218
- `MEASurement<m>:RESult:RMS?` on page 1218
- `MEASurement<m>:RESult:STDDev?` on page 1218
- `MEASurement<m>:RESult:WFMCount?` on page 1218
- `MEASurement<m>:RESult:STARt?` on page 1219
- `MEASurement<m>:RESult:STOP?` on page 1219
- `MEASurement<m>:ARNames` on page 1218
- `MEASurement<m>:ARES?` on page 1217

The peak and average values and the standard deviation of the long-term points are also shown in the graph of the long-term measurement.

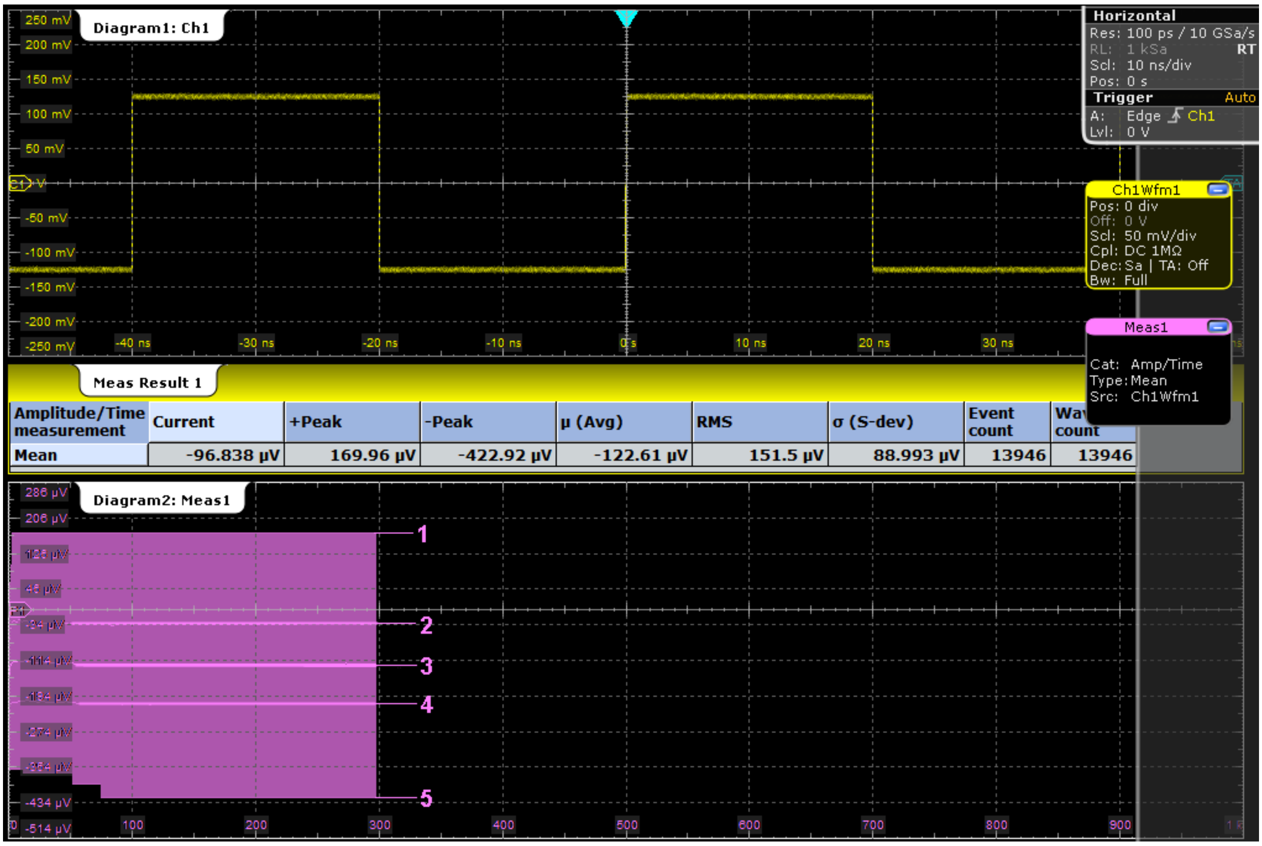


Figure 7-10: Long-term measurement with statistics

1 = +Peak, maximum
2 = Average + standard deviation
3 = Average
2 = Average - standard deviation
1 = -Peak, minimum

Stopping and starting the acquisition does not reset statistics but only stops and continues them.

The instrument only resets statistical evaluation if you change measurement setup:

- Select measurement type
- Create or modify gate
- Enable/disable long-term measurement and histogram
- Enable continuous autoscale with enabled histogram
- Switch on/off channels
- Enable/disable cursors
- Tap "Reset" or "Clear sreen results"

After a reset, new statistics are compiled beginning with the next acquired waveform.

7.2.10.2 Long-Term Measurements

Long-term measurements show the behavior of measurement results over a longer time or for many samples. Therefore the measurement results of a specified time period are summarized into one long-term point. For each point, the current value measured at the end of the time period is written to the long-term waveform. In addition, statistical results for each time period are calculated, saved, and reset. This reset avoids constantly rising maximum or constantly falling minimum values until the end of the measurement.

You can define the number of long-term points and export the long-term data, including statistical results. The measurement histogram is a vertical histogram shown in the long-term diagram.

See also: "Enable (Long term)" on page 349.

Performing Long-Term Measurements

1. On the "Meas" menu, select "Result Analysis".
2. Select the subtab for the measurement you want to configure.
3. Under "Long term", tap "Enable".
4. Since the waveform can change in the process of time, enable "Continuous auto scale" to adapt the scaling automatically.
   Alternatively, tap the "Auto scale" to adjust the scale once and to see the long-term waveform.
5. Tap "Horizontal scaling".
6. Define the "Number of points" to be shown in the long-term diagram.
7. Set the "Scale mode" that defines the period of time from which one long-term point is created.
   See "Scale mode" on page 352 for setting details.

   For each long-term measurement point, the current measurement value is added to the long-term waveform.
8. If you need the statistical data of the long-term points:
   a) Tap "Result Analysis" to return to the measurement settings.
   b) Enable statistics.
   c) Let the measurement run and export the data when finished, see Chapter 11.2.5, "Result Analysis", on page 454.

7.2.10.3 Histograms on Measurement Results

Histograms are available not only for channel, math and reference waveforms, but also on measurement results. These histograms cannot be configured, and they are shown in a separate diagram. The source is a measurement, and all events in all acquisitions are are measured and included in the histogram. If the histogram is based on long-term measurement, it is shown in the long-term diagram.

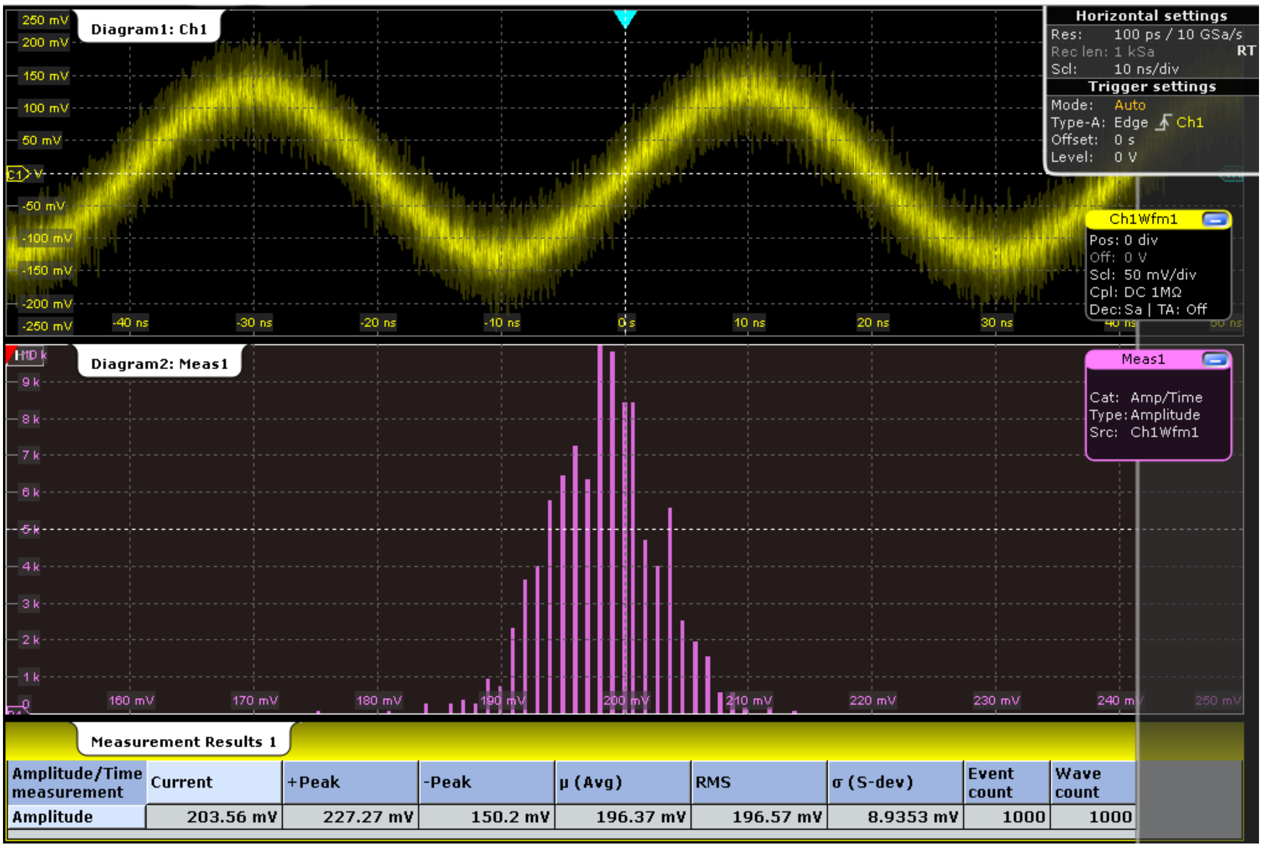


Figure 7-11: Statistical results and measurement histogram of an amplitude measurement

See also: "Enable (Histogram)" on page 349.

7.2.10.4 Track

The track is a waveform that shows measurement values in time-correlation to the measured signal. It is the graphical interpretation of all measurement values of a single acquisition.

The track is available for most amplitude/time measurements (except for High, Low, Amplitude, Max, Min, Peak to peak, Mean, RMS, S-dev, Pos. and Neg. overshoot, and Area), and for jitter measurements.

Enabling the track enables also the Continuous auto scale and Measure all events in each acquisition.

To analyze the track, you can use cursor measurements and zoom.

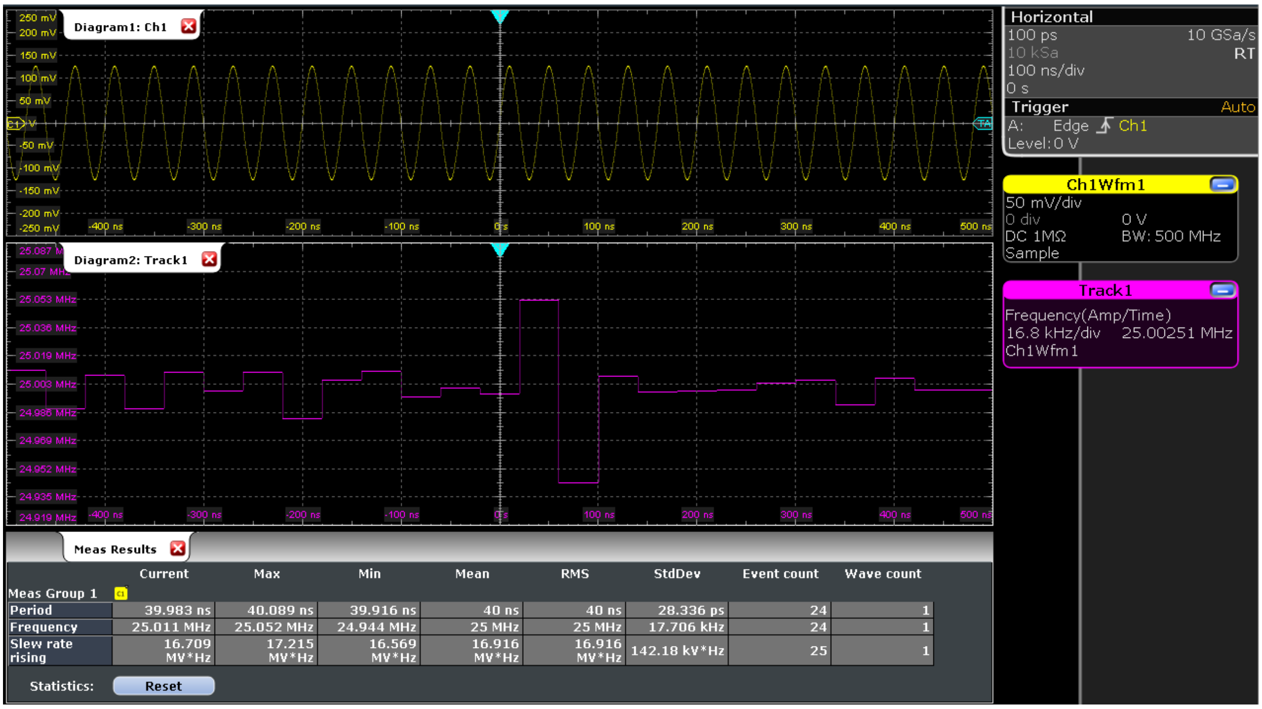


Figure 7-12: Track on a frequency measurement

Creating a track waveform

1. Configure the measurement.
2. Activate the measurement.
3. Select the "Result Analysis" tab.
4. Enable the track.

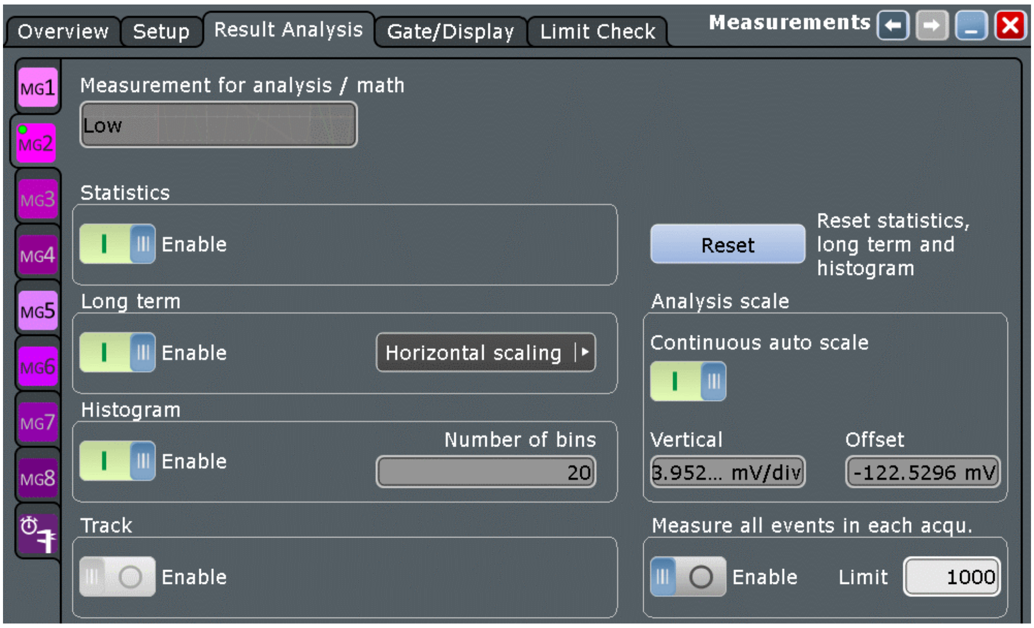
See also: "Enable (Track)" on page 350.

7.2.10.5 Settings for Result Analysis

Access: "Meas" menu > "Result Analysis"

The settings in the "Result Analysis" tab activate and configure long-term measurements, statistical calculations, and the histogram of measurement results over a longer period of time.

For scaling settings of the long-term diagram, see Chapter 7.2.10.6, "Horizontal Long-Term Scaling", on page 351.



Measurement for analysis / math

Shows the selected measurement type, see also "Measurement type" on page 298.

Statistics / Enable

Enables the calculation and display of statistical results.

Remote command:

`MEASurement<m>:STATistics[:ENABle]` on page 1252

Enable (Long term)

Enables long-term measurement of the main measurement.

Remote command:

`MEASurement<m>:LTMeas[:STATe]` on page 1257

Enable (Histogram)

Displays a histogram of measurement results - the cumulative occurrence distribution of measurement results in a graphic. Enabling the histogram enables also the statistics and "Measure all events in each acquisition".

Remote command:

`MEASurement<m>:STATistics:HISTogram` on page 1253

Number of bins

Sets the number of bins - the number of vertical bars that build the histogram.

If "Continuous auto scale" is enabled, the instrument determines the number of bins based on the timebase, the current measurements, and other settings. To set the number of bins manually, disable "Continuous auto scale".

Remote command:

`MEASurement<m>:STATistics:HBINs` on page 1254

Enable (Track)
Enables the track of measurement results over time and displays the track waveform. It is the graphical interpretation of all measurement values of a single acquisition.

The track is available for most amplitude/time measurements (except for High, Low, Amplitude, Max, Min, Peak to peak, Mean, RMS, S-dev, Pos. and Neg. overshoot, and Area), and for jitter measurements.

Enabling the track enables also the Continuous auto scale and Measure all events in each acquisition.

Before you can enable the track, activate the appropriate measurement.

With option R&S RTE-K5 I²S audio signals, you can use the track for protocol measurements on decoded audio buses, see Chapter 12.8.5, "Track", on page 612.

Remote command:
`MEASurement<m>:TRACk[:STATe]` on page 1264

Reset
Immediately resets the histogram, the long-term measurement and the statistics.

Stopping and starting the acquisition does not reset these analyses but only stops and continues them.

To delete all results, waveforms and history, select "Display" menu > "Clear all".

Remote command:
`MEASurement<m>:STATistics:RESet` on page 1255

Analysis scale
The measurement scale of a long-term measurement diagram or measurement histogram can be set automatically by the instrument, or manually.

Use automatic scaling if the measurement is running and you cannot see the expected results.

"Continuous auto scale"
Performs an automatic scaling whenever the long-term waveform or the histogram does not fit in the diagram during the measurement period.

"Vertical" Defines the vertical scaling per division for long-term measurement period and the measurement histogram.

"Offset" Defines an offset for the long-term measurement and the measurement histogram.

Remote command:
`MEASurement<m>:VERTical:CONT` on page 1256
`MEASurement<m>:VERTical:AUTO` on page 1256
`MEASurement<m>:VERTical:SCALe` on page 1257
`MEASurement<m>:VERTical:OFFSet` on page 1256

Measure all events in each acquisition
Normally, only one measurement is performed for each acquired waveform to get best performance. If "Measure all events in each acquisition" is enabled, more than one result is taken from one acquired waveform and the results are included evaluation. For example, the rise time is measured on all pulses in the waveform, not only on the first.

The result box shows only the first result of the waveform, the following results are used only for evaluation.

All event results are also considered in limit and margin checks and can initiate an action. However, the icons in the result box only indicate violations of the first result.

Measuring all events is enabled automatically when calculating statistics and histograms. It can also be useful when generating tracks; however, it reduces the performance of the instrument.

The number of considered results can be restricted: "Limit" sets the maximum number of measurement results per acquisition.

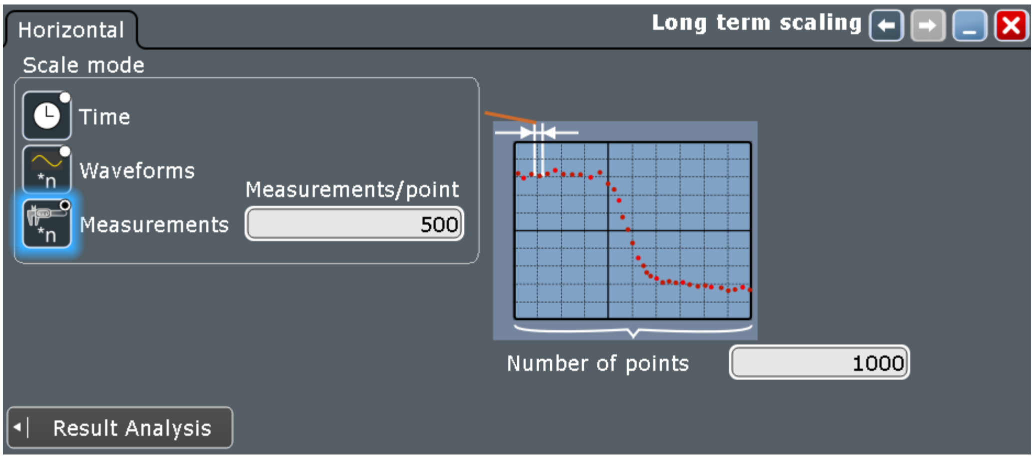
In firmware versions < 4.00, this setting was named "Multiple measurement".

Remote command:
`MEASurement<m>:MULTiple` on page 1253
`MEASurement<m>:MNOMeas` on page 1253

7.2.10.6 Horizontal Long-Term Scaling

In this dialog box, you define the horizontal scale of long-term measurement diagrams. The length of the long-term measurement is defined by the number of points.

If option R&S RTE-K5 I²S Audio Signals is installed, the trend diagram is configured here.



Number of points
Defines the total number of points to be displayed in the long-term measurement diagram.

Remote command:
`MEASurement<m>:LTMeas:COUNt` on page 1257

Scale mode

Defines when the points of a long-term measurement are created.

If statistics are enabled, each long-term measurement point shows the statistical mean and standard deviation of the results measured during the defined period.

If statistics are disabled, the first measurement result of each period is taken as long-term measurement point.

"Time"	Sets one long-term measurement point for the time defined in "Time/point".
"Waveforms"	Sets one long-term measurement point for several acquired waveforms. The number is defined in "Waveforms/point".
"Measurements"	Sets one long-term measurement point for several measurement results. The number is defined in "Measurements/point".

Remote command:
`MEASurement<m>:STATistics:MODE` on page 1254

Time / point

Defines the time to create one point of the long-term measurement. The "Time / point" value is a lower time limit. The actual time between two points depends on the acquisition and postprocessing time.

The long-term measurement is not a data logger with equidistant points as the time between two points varies.

This setting is only available if "Scale mode" is set to "Time".

Remote command:
`MEASurement<m>:STATistics:RTIMe` on page 1255

Measurement time

Defines the total duration of the long-term measurement: *Time/point * Number of points*.

This setting is only available if "Scale mode" is set to "Time".

Remote command:
`MEASurement<m>:LTMeas:TIME` on page 1258

Wfms / point

Defines the number of measured waveforms from which one point of the long-term measurement is created.

This setting is only available if "Scale mode" is set to "Waveforms".

Remote command:
`MEASurement<m>:STATistics:RCOunt` on page 1254

Measurements / point

Defines the number of measurement results from which one point of the long-term measurement is created.

This setting is only available if "Scale mode" is set to "Measurements".

Remote command:
`MEASurement<m>:STATistics:RMEascount` on page 1255

7.2.11 Limit and Margin Checks

Limit and margin checks evaluate whether the measurement result exceeds a specified value. Violations are indicated by icons in the result box. Furthermore, you can define actions that are performed on limit or margin violation, like saving the waveform or printing the measurement results.

The following results are considered in limit and margin checks:

- All selected measurements.
- All measured events in an acquisition: all results can initiate an action. However, the icons in the result box indicate only violations of the first result.
- Statistical results. Limit and margin violations of statistical results are indicated by icons in the result box. These violations do not initiate an action.

See also: Chapter 7.2.2.1, "Measurement Status", on page 299

7.2.11.1 Performing Limit Checks

1. On the "Meas" menu, select "Limit Check".
2. Select the subtab of the measurement you want to configure.
3. Under "Limit check", select "Limit only" to distinguish only between valid and non-valid values.
   Select "Margin&Limit" to check two values, where the margin is still valid, while the limit is not.
4. Define the valid value range for the measurement.
   The margins must always be within the valid value range. If necessary, the limit or margin values are adapted to match the selected valid range.
   See also "Upper limit, Lower limit, Upper margin, Lower margin, Valid range" on page 354
5. Define what happens when the defined limits and margins are exceeded.

   For each action, define when the instrument starts it:
   - If the limits or margins are exceeded.
   - If the measurement is completed without limit violations.
   - Not at all.

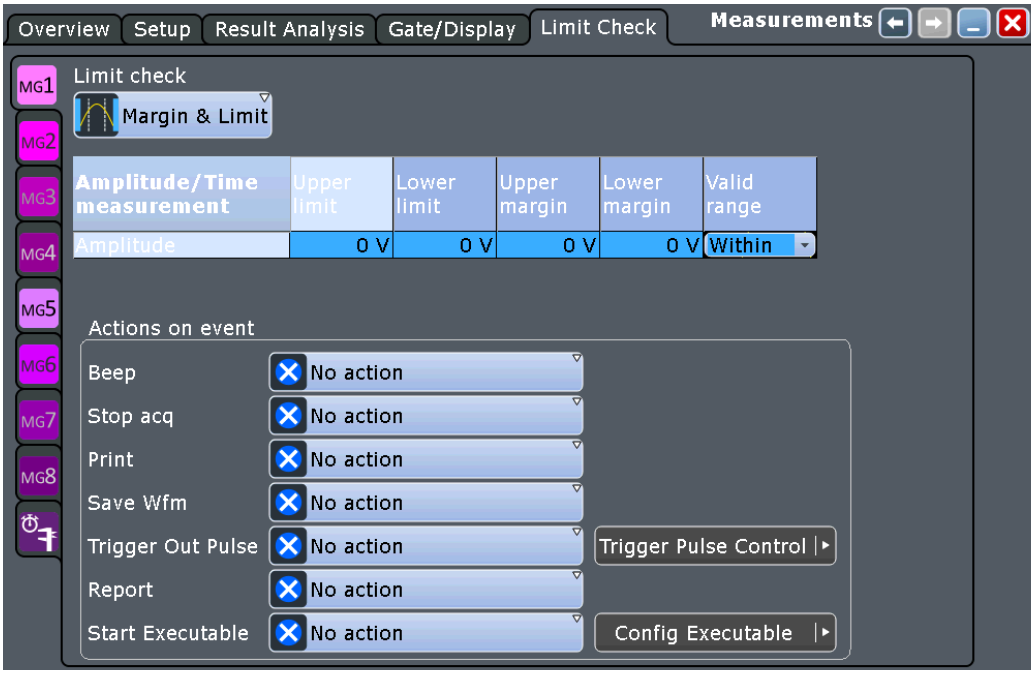
   If "Print" is selected, configure the print settings as described in Chapter 11.4.2, "Printing Screenshots", on page 466.

   As a result of the limit check, the specified actions are performed and an icon indicates the status in the result box.

7.2.11.2 Limit and Margin Settings

Access: "Meas" menu > "Limit Check".

If the check is enabled, a table is displayed where you can set the limit and margin values and the range of valid measurement results.



Limit check
Enables the limit or margin check.

"Off"	No limit check is performed.
"Limit only"	Limits are checked for violation.
"Margin & Limit"	Margins and limits are checked for violation.

Upper limit, Lower limit, Upper margin, Lower margin, Valid range
Set the limits and margins for each measurement, and also specify the valid range.

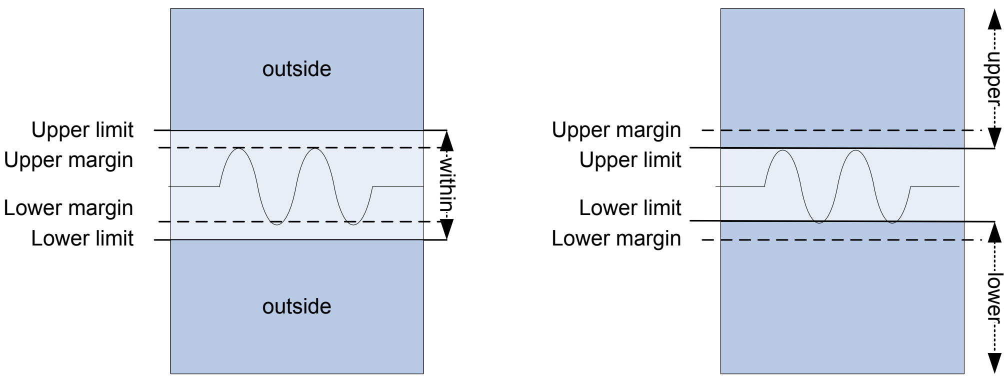


Figure 7-13: Limit and margin definition

Limits are stricter than the margins for the value check. Thus, the margins must be within the valid range. If necessary, the limit and margin values are adapted according to the selected valid range.

The settings are only visible if "Limit check" is enabled.

Remote command:
`MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit` on page 1228
`MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin` on page 1229
`MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit` on page 1228
`MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin` on page 1229
`MEASurement<m>:AMPTime:LCHeck<n>:VALid` on page 1228
`MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit` on page 1231
`MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin` on page 1232
`MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit` on page 1231
`MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin` on page 1232
`MEASurement<m>:EYEJitter:LCHeck<n>:VALid` on page 1231
`MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit` on page 1249
`MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin` on page 1249
`MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit` on page 1249
`MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin` on page 1249
`MEASurement<m>:HISTogram:LCHeck<n>:VALid` on page 1248
`MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit` on page 1239
`MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin` on page 1240
`MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit` on page 1239
`MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin` on page 1240
`MEASurement<m>:SPECtrum:LCHeck<n>:VALid` on page 1239

7.2.11.3 Actions on Limit Check Results

On the "Limit Check" tab, you also define what happens when the limits and margins are exceeded. Limit checking must be enabled.

Actions are initiated by all measurements and all results measured on one acquisition.

Note that the violation actions do not distinguish between a margin violation and a limit violation. However, different icons are displayed in the result box.

For each action, you can define the event on which the action is initiated:

- On violation
  The action is initiated when the limits or margins are exceeded during the measurement.
- On successful completion
  The action is initiated when a defined number of acquisitions has been captured, and the limits or margins were not exceeded.

Independent of these actions, an icon is displayed in the result box, see Chapter 7.2.2, "Measurement Results", on page 298.

Beep
Generates a beep sound.

Remote command:
MEASurement<m>:ONViolation:BEEP on page 1269

Stop acq
Stops data acquisition on violation.

Remote command:
MEASurement<m>:ONViolation:ACQStop on page 1269

Print
Prints a screenshot including the measurement results to the printer defined in the "Print" dialog box (see Chapter 11.4.2, "Printing Screenshots", on page 466).

Remote command:
MEASurement<m>:ONViolation:PRINt on page 1270

Save Wfm
Saves the waveform data to the file specified in [FILE] > "Save" > "Waveform".

Remote command:
MEASurement<m>:ONViolation:WFMSave on page 1270

Trigger Out Pulse
Creates a pulse on the TRIGGER OUTPUT connector on limit violation.

When "Trigger Out Pulse" is used, the trigger control option "Enable trigger out" is disabled. Thus, the trigger-out pulse is created only on limit violation but not when a trigger occurs. The pulse is provided always with the minimum delay of 800 ns, the "Delay" cannot be set.

See also: "Trigger out signal setup" on page 231.

Remote command:
MEASurement<m>:ONViolation:TRIGgerout on page 1271

Report
Creates and saves a report using the settings defined in "File" menu > "Report Setup".

Remote command:
MEASurement<m>:ONViolation:REPort on page 1270

Start Executable
Starts an external application. Tap "Config Executable" to set the application path and parameters.

See: Chapter 3.5, "External Application", on page 120.

Remote command:
MEASurement<m>:ONViolation:RUNexec on page 1271

7.3 Quick Measurements

Quick measurement performs a set of up to eight amplitude/time measurements on one source, simply by tapping the "Quick measurement" toolbar icon. The results are displayed in a results box. You can configure the measurement type to be included in quick measurement. The current configuration can be saved to repeat the measurement quickly.

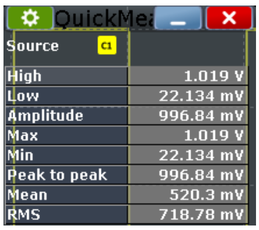
7.3.1 Starting the Quick Measurement

If the "Quick measurement" icon is not visible on the toolbar, add it to the toolbar: see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.

1. Tap the waveform that you want to measure.
2. Tap the "Quick measurement" icon on the toolbar.

3. Tap the diagram.

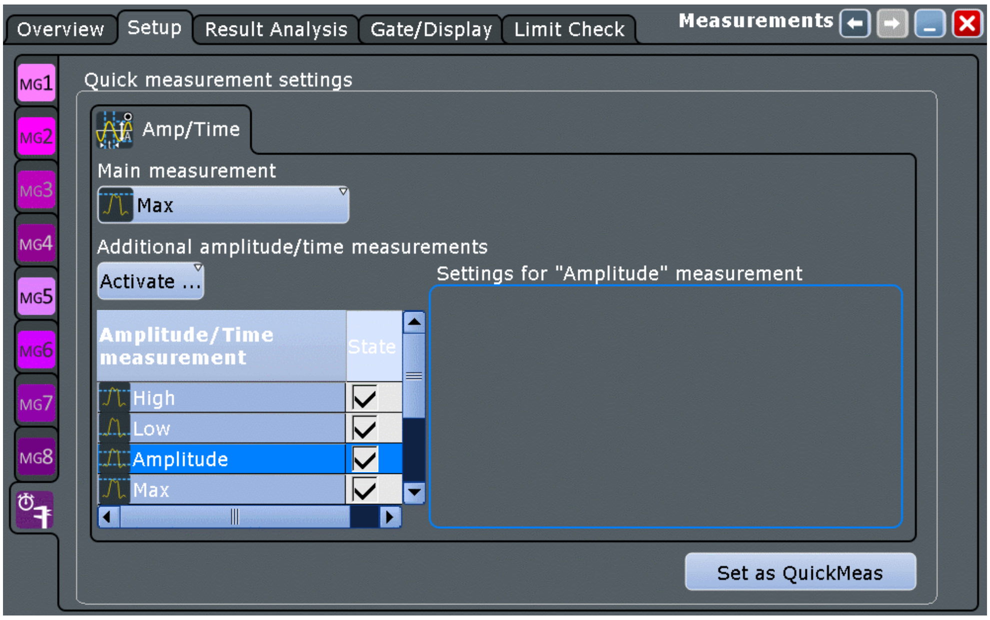
   The result box shows the results of the default quick measurement.



7.3.2 Configuring the Quick Measurement

The default configuration of the quick measurement includes already 8 amplitude measurements. If these measurements do not fit the measurement task, you can modify the selection.

1. On the "Meas" menu, select "Setup".
2. Under "Quick Measurement", tap "Setup".
3. In the list of active measurements, disable all measurements that you do not need.



4. Tap the "Main measurement" button.
5. Select the measurement that you want to use for long-term and histogram analysis.
6. Tap "Activate". Select a measurement that you want to add to the quick measurement.
   For details on the available measurements, see Chapter 7.2.5.1, "Overview of Amplitude/Time Measurements", on page 310.
7. Repeat the previous step until the setup is complete.
8. Tap each measurement in the list. If additional settings are needed, adjust them under "Settings".
9. Tap "Set as QuickMeas" to save the configuration.

   The saved configuration remains until you save another QuickMeas setup, there is no reset.

Set as QuickMeas
Saves the current QuickMeas setup. The saved QuickMeas configuration is used when you start a new quick measurement. It remains until you save another QuickMeas setup, or until you reset the instrument to "Factory defaults". The "Set as QuickMeas" is not available if the current configuration already has been saved.

8 Spectrum Analysis

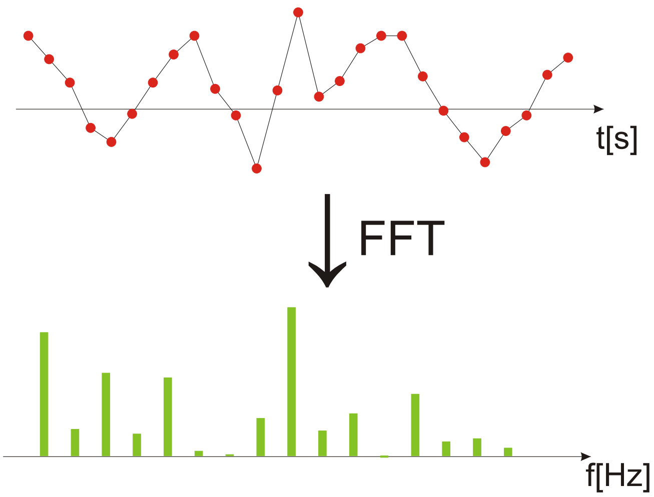
The R&S RTE provides two ways of spectrum analysis:

- Basic FFT calculation, which is included in the firmware
- Spectrum Analysis option R&S RTE-K18, which provides a wide range of analysis possibilities, for example, spectrogram, cursor and automatic measurements.

8.1 FFT Analysis

8.1.1 Fundamentals of FFT Analysis

During FFT analysis, a signal in the time domain is converted to a spectrum of frequencies. As a result, either the magnitude or the phase of the determined frequencies can be displayed. FFT analysis can be restricted to an extract of the original time base, and the results display can be restricted to a specified frequency range.

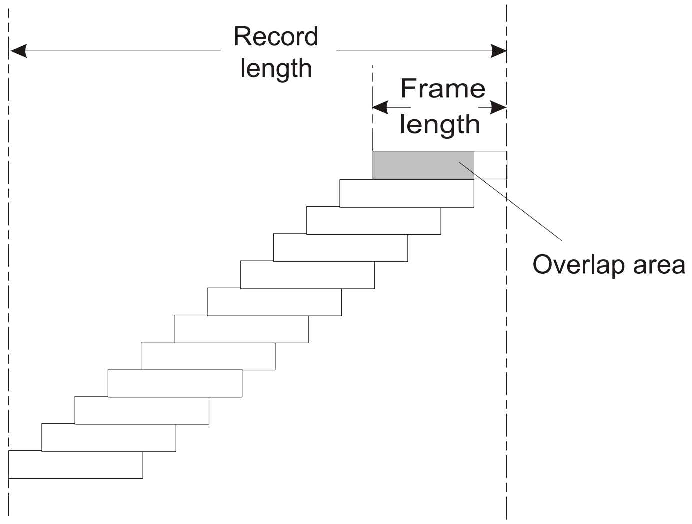


Frames/Segments

To convert the time domain signal to a frequency spectrum, an FFT (Fast Fourier Transformation) unit is used which converts a vector of input values into a discrete spectrum of frequencies.

Conventional oscilloscopes calculate one FFT per capture. The R&S RTE can calculate multiple FFTs per capture by dividing one capture into several *segments*, or *frames*. Thus, the R&S RTE can visualize how the frequency content of a signal changes over time which helps to detect intermittent or sporadic signal details. Furthermore, the R&S RTE allows consecutive frames to overlap. This is especially useful in conjunction with window functions since it enables a gap-free frequency analysis of the signal.

The overlapping factor can be set freely. The higher the overlap factor, the more frames are used. This leads to more individual results and improves detection of transient signal effects. However, it also extends the duration of the calculation. The size of the frame depends on the number of input signal values (record length), the overlap factor, and the FFT size (number of samples used for FFT calculation).



Window functions

Each frame is multiplied with a specific window function after sampling in the time domain. Windowing helps minimize the discontinuities at the end of the measured signal interval and thus reduces the effect of spectral leakage, increasing the frequency resolution.

There are several window functions that can be used in FFT analysis. Each of the window functions has specific characteristics, including some advantages and some trade-offs. Consider these characteristics carefully to find the optimum solution for the measurement task.

For details, see "Window type" on page 369.

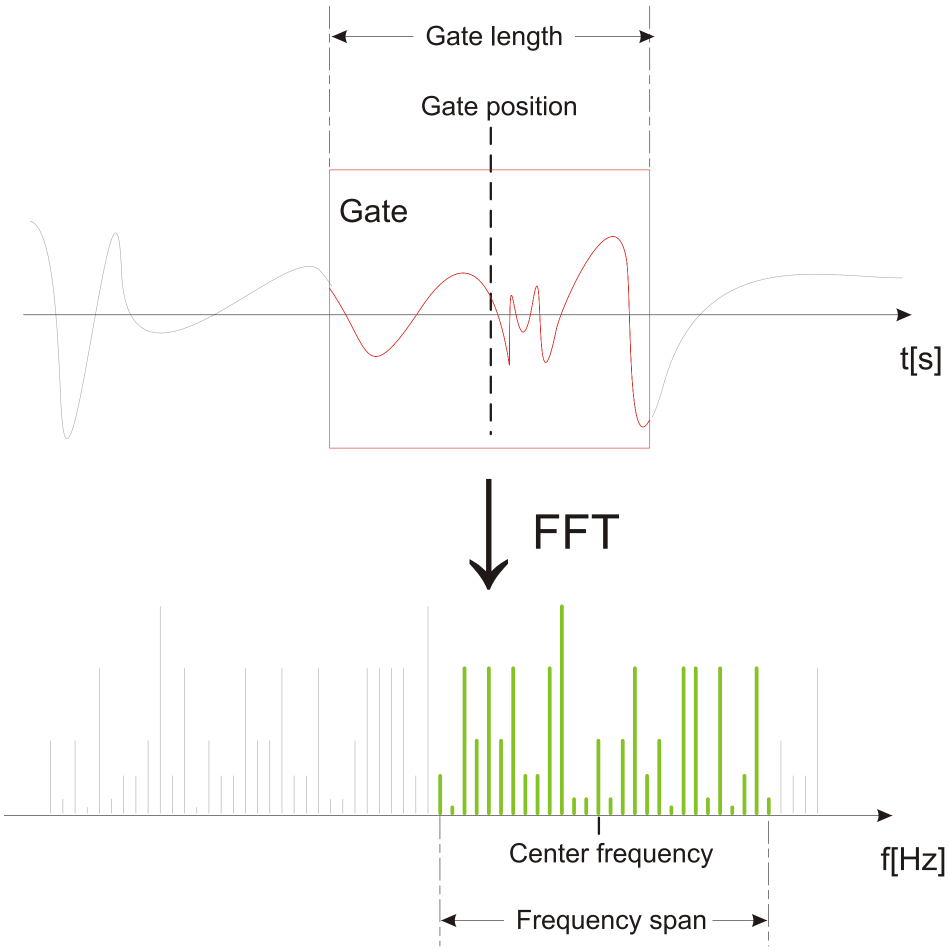
Combining FFT results

After the window function has been applied to the FFT results for each segment, the results for all segments of the data acquisition must be combined to obtain the final waveform. Various arithmetic functions are available for FFT segments, such as averaging, enveloping, or minimum and maximum calculation.

Gating functions

You can restrict the time base of the input signal for which FFT analysis is to be performed. There are various methods to do so:

- Define absolute start and stop times for the time base extract
- Define relative start and stop values that define a percentage of the original time base
- Couple the time base extract for FFT to an active zoom area.



Restricting the result range

You can restrict the results of the FFT analysis to a specified frequency range. The frequency range can be defined in two ways:

- Define a center frequency and frequency span
- Define start and stop frequencies

Using the new cursor functions for spectrum waveforms you can easily determine the results for the current center frequency by moving the cursor to that frequency ("C1 to Center"). If you detect a point of interest in the spectrum diagram, you can place the cursor on it and then move the center frequency to the position of the cursor automatically ("Center to C1").

See Chapter 7.1, "Cursor Measurements", on page 284.

Magnitude vs. phase display

The result of an FFT analysis is a spectrum of frequencies. Either the magnitudes or the phases of those frequencies are displayed, depending on the used FFT function. In "Basic" mode, and for the "Advanced" mode FFT functions |FFT|, FFT (re) and FFT (im), the magnitude is displayed. For the "Advanced" mode FFT (φ) function, the phase is displayed.

For magnitude display, you can select the scale and range of magnitudes to be displayed. For linear scaling, the vertical value range of the input signal is used. For logarithmic scaling, the logarithmic power of the frequency is displayed. In this case, the input signal must be given in either Volt or Watt. The resulting value range is defined by a maximum value and a range size. Logarithmic scaling can also be set in relation to a given reference value.

For phase display, you can select the unit and suppress phases beneath a threshold value which are most likely caused by noise. The value range [-π, + π] or [-180°, +180°] is used. Phase shifts due to a limitation of the value range can be eliminated using the "Unwrap" function.

Dependencies between FFT parameters

FFT analysis in the R&S RTE is highly configurable. Several parameters, including the resolution bandwidth, frequency span and center frequency, can be defined according to your requirements. Note, however, that several parameters are correlated and not all can be configured independently of the others.

The **resolution bandwidth** defines the minimum frequency separation at which the individual components of a spectrum can be distinguished. Small values result in high precision, as the distance between two distinguishable frequencies is small. Higher values decrease the precision, but increase measurement speed.

The minimum achievable RBW depends on the integration time which is equivalent to the number of samples available for FFT calculation. If a higher spectral resolution is required, the number of samples must be increased by using a higher sample rate or longer record length. To simplify operation some parameters are coupled and automatically calculated, such as record length and RBW.

The **frequency span** and **center frequency** define the start and stop frequency of the spectral diagram. By default, a suitable frequency range according to the resolution bandwidth is selected, in respect to performance and precision. Span and RBW settings are coupled, so that the parameters can be adjusted automatically as necessary.

With a **Span/RBW ratio** of 100 and a screen resolution of 1000 pixels, each frequency in the spectrum is displayed by 10 pixels. A span/RBW ratio of 1000 provides the highest resolution. For full flexibility, the span/RBW coupling can also be disabled. Note, however, that a higher span/RBW ratio (i.e. low RBW values and large frequency spans) result in large amounts of data and extend the duration of the calculation.

Advanced FFT functions

In "Advanced" math definition mode, other FFT results than the basic frequency magnitude can be displayed.

- **FFT (φ)**: phase display
- **FFT (im)**: imaginary part of FFT value (magnitude)
- **FFT (re)**: real part of FFT value (magnitude)

- **FFT -dφ*df** (group delay): the negative derivative of the phase with respect to frequency; useful to measure phase distortion

8.1.2 Configuring Spectrum Waveforms

During FFT analysis, a signal in the time domain is converted to a spectrum of frequencies. A basic spectrum waveform can be displayed quickly. By defining additional FFT parameters, the waveform can be configured in more detail.

As a result, either the magnitude or the phase of the determined frequencies can be displayed, or more complex FFT functions. Analysis can be restricted to an extract of the original time base, and the results display can be restricted to a specified frequency range.

To display a basic spectrum waveform using the toolbar

1. Tap the "FFT" icon on the toolbar.
2. Tap the waveform for which the FFT is performed.

   The first available math waveform is configured to use the selected waveform as a source and the "Mag(FFT(x))" operator, and is enabled. The spectrum waveform is displayed in a new diagram.

To display a basic spectrum waveform using the [MATH] key

1. Press the [MATH] key to open the "Math" dialog box.
2. In the "Setup" tab, in the "Basic" editor, select the input signal as "Source 1".
3. Select "Mag(FFT(x))" as the "Operator".
4. Select the "Enable math signal" icon.
5. If necessary, edit the spectrum waveform parameters as described in the following procedures.

To display advanced spectrum waveforms

In "Advanced" math definition mode, other FFT results than the basic frequency magnitude can be displayed.

1. In the "Setup" tab of the "Math" dialog box, select the "Advanced" expression editor.
2. Double-tap the edit area.

   The "FormulaEditor" is displayed.
3. Delete the contents of the edit field.
4. Tap the "More" key to display further functions in the editor.
5. Tap the required FFT function key.

6. Select the source channel.
7. Close the parenthesis.
8. Tap "Enter"

To configure the spectrum of FFT analysis

By default, a suitable frequency range for the expected horizontal values according to the resolution bandwidth is selected, in respect to performance and precision. Span and RBW settings are coupled. If a more precise evaluation is required, for example for postprocessing in a different application, disable the coupling and change the frequency ranges and resolution bandwidth values as required.

1. On the "Math", select "FFT Setup".
2. Tap the "Frequency axis" button to select the type of scaling you want to use: linear or logarithmic.
3. Disable the "Span/RBW coupling".
4. Specify the frequency range you want to display using one of the following methods:
   - Enter a "Center frequency" and a "Frequency span" that define the spectrum.
   - Enter a "Start frequency" and "Stop frequency" that define the spectrum.
   - Tap the "Full Span" button to display the complete spectrum resulting from the FFT analysis.
5. Define the resolution bandwidth for the FFT result.
   The resolution bandwidth defines how precise the results are, i.e. how close together the individual frequencies can be. Small values result in high precision, as the distance between two distinguishable frequencies is small. Higher values decrease the precision, but increase performance.

   You can define the RBW manually, or couple it to other FFT settings. Do one of the following:
   - To couple the RBW to the span, enable the "Span/RBW coupling" option and define the "Span/RBW ratio". The smaller the ratio, the higher the RBW becomes to display the same frequency span.
   - Enter the "Resolution BW" manually. The "Span/RBW coupling" option is automatically disabled.
   - To couple the RBW to the specified record length, in the "FFT Gating" tab of the "Math" dialog box, select the "Record length controlled" option. This option is only available if no gate is being used ("Use Gate" disabled).
6. Select the most suitable "Window type" for your source data. Window functions are multiplied with the input values and thus can improve the FFT display. For details, see "Window type" on page 369.
7. Optionally, select an arithmetic mode for the FFT segments. This mode defines how the individual segment results are combined to a final spectrum waveform. In the "FFT Overlap" tab of the "Math" dialog box, tap "FFT Segment Arithmetic" and select the required mode from the list.

8. If you use an arithmetic mode, increase the "Overlap factor" for neighboring segments to increase the accuracy of the results.

To restrict the input values (gating)

By default, the FFT is calculated for the entire record length as defined for the data acquisition. However, you can restrict the time range for which the FFT is calculated, resulting in a restricted spectrum. Alternatively, the record length can be determined automatically according to the selected RBW.

1. Select the "FFT Gating" tab of the "Math" dialog box.
2. Determine how the input length is configured by selecting one of the following options:
   - To ensure that the FFT is calculated for the full defined record length, select the "Record length controlled" option. This option is only available if no gate is being used ("Use Gate" disabled). The RBW is adapted so that the record length can be acquired in the specified acquisition time. However, the RBW is restricted, so that data acquisition may fail if the record length is too long for the specified acquisition time.
   - To couple the used record length to the required RBW, select the "RBW controlled" option. This option is only available if no gate is being used ("Use Gate" disabled).
     The required acquisition time for the defined RBW value is indicated.
   - To restrict the basis of the FFT calculation to a certain time base, configure a time gate, that is: an extract of the time base in the original diagram. To do so, enable the "Use Gate" option, then do one of the following:
     – Select the "Absolute" mode and enter the "Start" and "Stop" times that define the gate area.
     – Select the "Relative" mode and enter the percentages of the total time base that define the "Relative Start" and "Relative Stop" times.
     – If a zoom area has already been defined in the original diagram and you want to use the same time base for FFT analysis, select "Zoom coupling".

   The spectrum waveform displays the spectrum for the specified time span.

To configure magnitude results

1. Open the "FFT Y-Units" tab of the "Math" dialog box.
2. Select the scaling unit. Use logarithmic scaling only for input values in volt or watt.
3. Select if you want to configure the value range manually or use the automatic settings by tapping the corresponding icon.
4. In manual mode, define the size of the "Vertical range" and the "Vertical maximum" to be displayed.
   In automatic mode, define the size of the "Range" to be displayed.
   For logarithmic scaling in dB, also define the "Reference level" to be used.

To configure phase results

1. Open the "FFT Y-Units" tab of the "Math" dialog box.
2. Select the scaling unit.
3. To eliminate phase shifts due to a limitation of the value range, enable the "Unwrap" function.
4. To suppress small phase values due to noise, enable the "Suppression" function and enter a "Threshold" value.

To couple spectrum displays

The settings for one or more spectrums can be coupled. Thus, if any FFT setting for any of the coupled spectrums are changed, they are changed for all coupled spectrums.

1. Open the "FFT Coupling" tab of the "Math" dialog box.
2. Select the spectrums that you need to couple. You cannot select the spectrum for the currently selected math waveform. Its settings are applied to the selected spectrums.
3. If necessary, define an FFT function to be used for the coupled math waveforms so that a spectrum is displayed. See Chapter 6.3.1, "Displaying Math Waveforms", on page 257.

8.1.3 FFT Configuration Settings

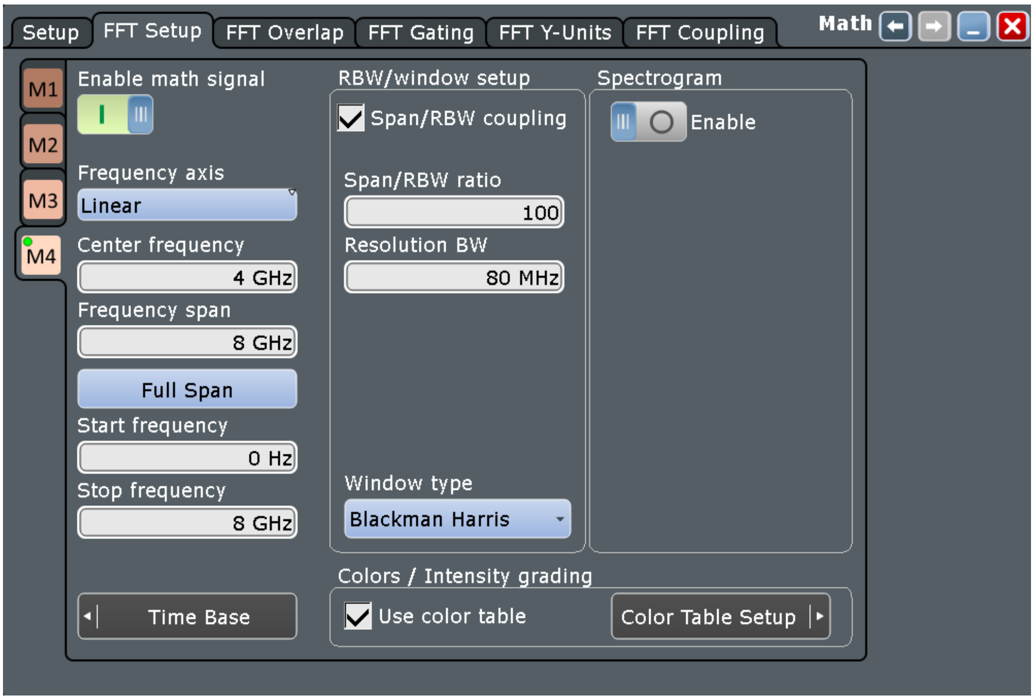
- FFT Setup....................366
- FFT Overlap....................370
- FFT Gating....................372
- FFT Y-Units....................374
- FFT Coupling....................377

8.1.3.1 FFT Setup

Access: [Math] > "FFT Setup"

In this tab, you define the settings for the FFT window. The display can be restricted to the results for a certain time base extract and to a specified frequency range.

Additional settings are available on this tab if the Spectrum Analysis option (R&S RTE-K18) is installed.
See Chapter 8.2.4, "Spectrogram Configuration Settings", on page 381.



Enable Math Signal....367
Frequency axis (R&S RTE-K18 only)....367
Center frequency....368
Frequency span....368
Full span....368
Start frequency....368
Stop frequency....368
Span/RBW Coupling....368
Span/RBW Ratio....368
Resolution BW....369
Window type....369
Use color table....369

Enable Math Signal

If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:

`CALCulate:MATH<m>:STATe` on page 1190

Frequency axis (R&S RTE-K18 only)

Defines the scaling method for the frequency (x-) axis of the spectrogram.

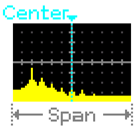
"Logarithmic" Logarithmic scaling

"Linear Unit" Linear scaling

Remote command:

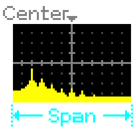
`CALCulate:MATH<m>:FFT:LOGScale` on page 1279

Center frequency
Defines the position of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The width of the range is defined using the "Frequency span" setting.



Remote command:
CALCulate:MATH<m>:FFT:CFRequency on page 1280

Frequency span
The span is specified in Hertz and defines the width of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The position of the span is defined using the "Center frequency" setting.



Remote command:
CALCulate:MATH<m>:FFT:SPAN on page 1280

Full span
Displays the full frequency span.

Remote command:
CALCulate:MATH<m>:FFT:FULLspan on page 1280

Start frequency
Defines the start frequency of the displayed frequency span.

Remote command:
CALCulate:MATH<m>:FFT:STARt on page 1279

Stop frequency
Defines the stop frequency of the displayed frequency span.

Remote command:
CALCulate:MATH<m>:FFT:STOP on page 1280

Span/RBW Coupling
Couples the frequency span to the "Resolution BW" setting.

Remote command:
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO on page 1281

Span/RBW Ratio
Defines the coupling ratio for Span/RBW. This setting is only available if CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO is ON.

Remote command:
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio on page 1281

Resolution BW

Defines the resolution bandwidth. Note that the resolution bandwidth is correlated with the span, record length and acquisition time. If a constant record length is to be used, the RBW may be adapted if the required number of samples cannot be acquired. If span and RBW values are coupled, changing the span will also change the RBW.

For details see Chapter 8.1.1, "Fundamentals of FFT Analysis", on page 359.

Remote command:
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue] on page 1282
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted? on page 1281

Window type

Windowing helps minimize the discontinuities at the end of the measured signal interval and thus reduces the effect of spectral leakage, increasing the frequency resolution.

Various different window functions are provided in the R&S RTE to suit different input signals. Each of the window functions has specific characteristics, including some advantages and some trade-offs. Consider these characteristics carefully, to find the optimum solution for the measurement task.

Window type	Frequency resolution	Magnitude resolution	Measurement recommendation
Rectangular	Best	Worst	Separation of two tones with almost equal amplitudes and a small frequency distance
Hamming Hann	Good	Poor	Frequency response measurements, sine waves, periodic signals and narrow-band noise
Blackman Harris (default)	Worst	Best	Mainly for signals with single frequencies to detect harmonics Accurate single-tone measurements
Gaussian	Good	Good	Weak signals and short duration
Flattop2	Poor	Best	Accurate single-tone measurements
Kaiser Bessel	Poor	Good	Separation of two tones with differing amplitudes and a small frequency distance

Remote command:
CALCulate:MATH<m>:FFT:WINDow:TYPE on page 1282

Use color table

If enabled, the spectrum waveform (and a spectrogram, if available) is displayed according to the assigned color table. For information on the available color tables, see Chapter 3.4.2.2, "Color Tables", on page 108.

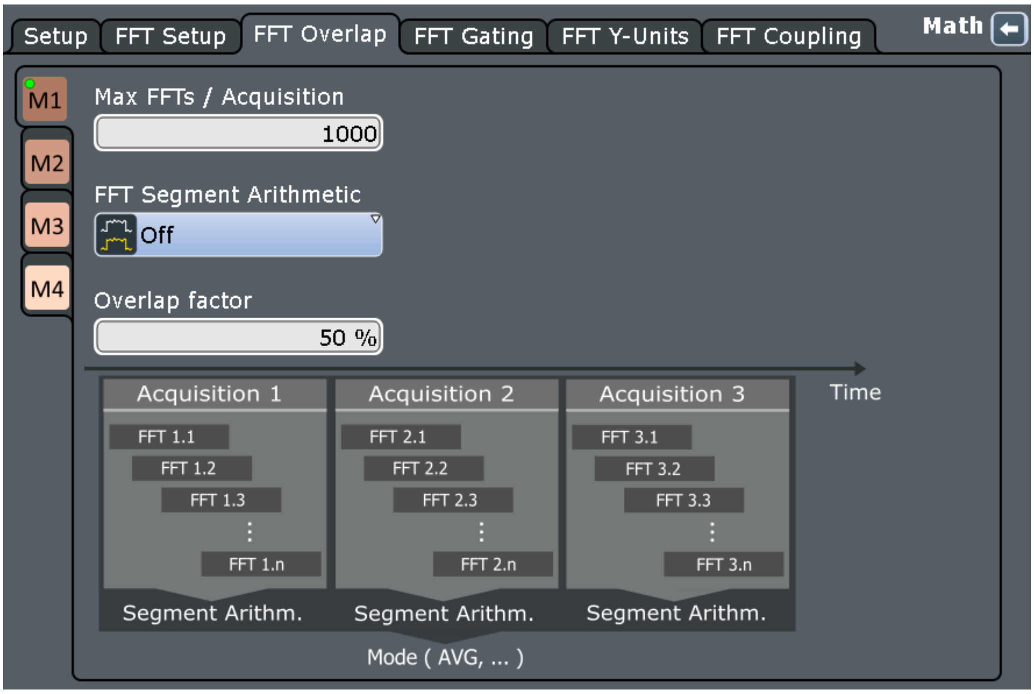
If this option is disabled, the preset color of the selected channel source is displayed, and the intensity of the specific signal color varies according to the cumulative occurrence of the values. For spectrum diagrams, this setting corresponds to the common waveform display. The spectrogram, on the other hand, is then displayed in gray tones, which is not useful.

Remote command:
`CALCulate:MATH<m>:FFT:USEColtab` on page 1293

8.1.3.2 FFT Overlap

Access: [Math] > "FFT Overlap"

In this tab, you define the settings for the magnitude and phase of the frequencies.



Enable Math Signal....370
Max FFTs / Acquisition....370
FFT Segment Arithmetics....371
Overlap Factor....371

Enable Math Signal
If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:
`CALCulate:MATH<m>:STATe` on page 1190

Max FFTs / Acquisition
Restricts the maximum number of FFTs to be calculated for each data acquisition. Due to the other parameter settings, the required number of FFTs may become very high, thus slowing performance. By restricting the number of FFTs, you can avoid performance loss without changing the other parameters.

However, if the maximum number of FFTs is lower than the required number to cover the entire waveform, the waveform may only be analyzed partially. In this case, the "Frame coverage" indicates the percentage of the waveform that was analyzed, i.e. which part of the data was included in the FFT calculation.

Remote command:
`CALCulate:MATH<m>:FFT:FRAMe:MAXCount` on page 1284
`CALCulate:MATH<m>:FFT:FRAMe:COVerage?` on page 1284

FFT Segment Arithmetics
FFT analysis can only be performed on a maximum number of values at once. If more values must be calculated, the input signal is divided into segments, each of which is calculated separately. The segments need not be disjunct, that is: they may overlap, so that some values have several FFT results. In this case, the arithmetic mode defines how the final result is calculated from the individual results.

The following methods are available:

"Off"	The data of only one segment is considered. In effect, no arithmetics are processed.
"Envelope"	Detects the minimum and maximum values for FFT calculation over all segments. The resulting diagram shows two envelope waveforms: the minimums (floor) and maximums (roof). These envelopes indicate the range of all FFT values that occurred.
"Average"	The average is calculated over all segments.
"RMS"	The root mean square is calculated over all segments. The result is the average power spectrum. If you measure the channel power on this RMS spectrum, you get the same result as for the average channel power measurement on segments.
"MinHold"	Determines the minimum result for each input value from the data of the current acquisition and the acquisitions before. Only available if option R&S RTE-K18 is installed.
"MaxHold"	Determines the maximum result for each input value from the data of the current acquisition and the acquisitions before. Only available if option R&S RTE-K18 is installed.

Remote command:
`CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics` on page 1283

Overlap Factor
Defines the minimum factor by which two neighboring segments overlap. If the required number of segments to cover the input values allows for more overlap, the factor is increased.

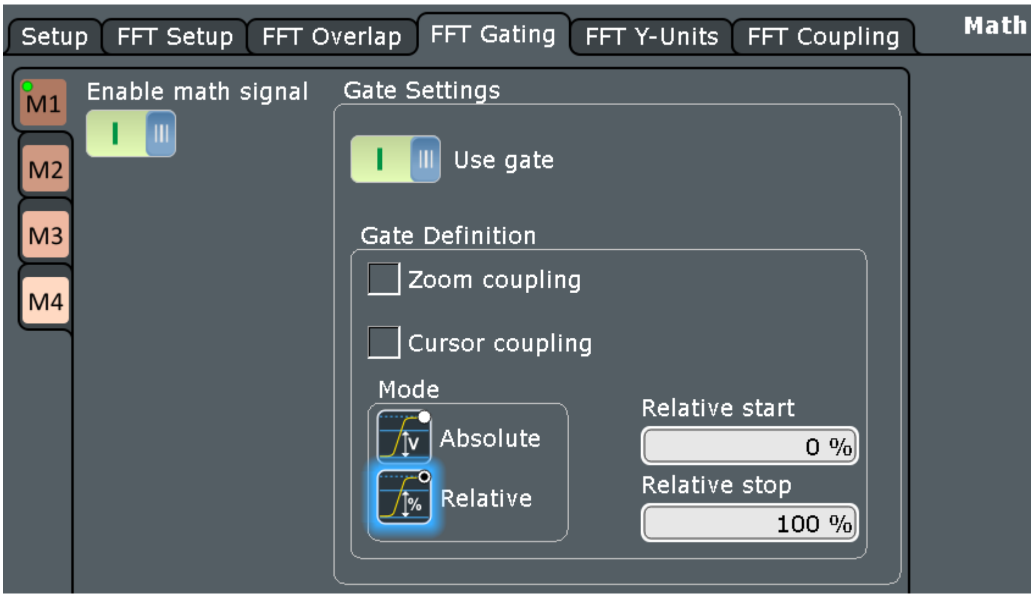
The higher the overlap factor, the more segments are used. This leads to more individual results and improves detection of transient signal effects. However, it also extends the duration of the calculation.

Remote command:
`CALCulate:MATH<m>:FFT:FRAMe:OFACtor` on page 1284

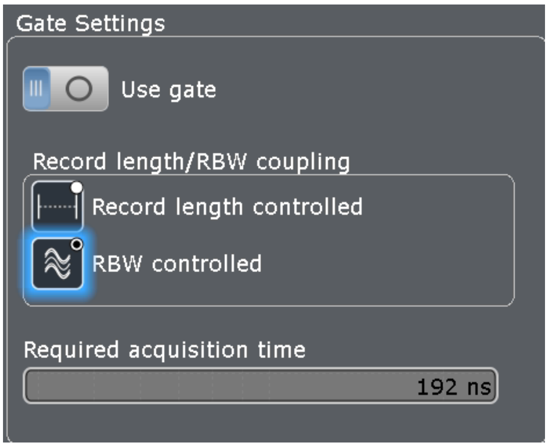
8.1.3.3 FFT Gating

Access: [Math] > "FFT Gating"

FFT gating allows you to restrict FFT analysis to a certain time base of the input signal.



If no gate is used, you can define the record length as dependent on the RBW, or the RBW as dependent on the record length (which is defined by the acquisition time).



Enable Math Signal....373
Use Gate....373
Gate Definition....373
└ Zoom coupling....373
└ Mode....373
└ (Relative) Start....373
└ (Relative) Stop....374
Record Length/RBW Coupling....374
Required acquisition time....374

Enable Math Signal
If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:
CALCulate:MATH<m>:STATe on page 1190

Use Gate
Enables FFT gating and shows the gate.

If enabled, the "Gate Definition" settings are displayed.

If disabled, the relation between the record length and the RBW can be defined manually instead.

When a gate is used, the RBW is adapted, if necessary. The smaller the gate, the higher the RBW.

For details, see Chapter 8.1.1, "Fundamentals of FFT Analysis", on page 359.

Gate Definition
Defines the gate settings for FFT gating.

Zoom coupling ← Gate Definition
Zoom coupling is available if a zoom is defined. As long as "Zoom coupling" is enabled, the gate area is defined identically to the zoom area - if you change the zoom, the gate changes as well.

If several zoom diagrams are defined, select the zoom diagram to be used for gating. The "Start" and "Stop" values of the gate are adjusted accordingly.

Zoom coupling can be set for measurement gates, FFT gates, and search gates. The zoom must be defined on the diagram that contains the signal source of the measurement, FFT, or search.

Remote command:
MEASurement<m>:GATE:ZCOupling on page 1267
MEASurement<m>:GATE:ZDIagram on page 1268
CALCulate:MATH<m>:FFT:GATE:ZCOupling on page 1287
SEARch:GATE:ZCOupling on page 1344
SEARch:GATE:ZDIagram on page 1344

Mode ← Gate Definition
Defines whether the gate settings are configured using absolute or relative values.

"Absolute" The gate is defined by absolute start and stop values.

"Relative" The gate's start and stop values are defined by a percentage of the value range.

Remote command:
CALCulate:MATH<m>:FFT:GATE:MODE on page 1286
MEASurement<m>:GATE:MODE on page 1266
SEARch:GATE:MODE on page 1342

(Relative) Start ← Gate Definition
Defines the starting value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt` on page 1286
`CALCulate:MATH<m>:FFT:GATE:RELative:STARt` on page 1286
`MEASurement<m>:GATE:ABSolute:STARt` on page 1266
`MEASurement<m>:GATE:RELative:STARt` on page 1266
`SEARch:GATE:ABSolute:STARt` on page 1343
`SEARch:GATE:RELative:STARt` on page 1343

(Relative) Stop ← Gate Definition
Defines the end value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP` on page 1286
`CALCulate:MATH<m>:FFT:GATE:RELative:STOP` on page 1287
`MEASurement<m>:GATE:ABSolute:STOP` on page 1266
`MEASurement<m>:GATE:RELative:STOP` on page 1266
`SEARch:GATE:ABSolute:STOP` on page 1343
`SEARch:GATE:RELative:STOP` on page 1344

Record Length/RBW Coupling
The record length and resolution bandwidth are coupled during FFT analysis. If you change one value, the other must be adapted accordingly. You can keep either value constant, thus preventing automatic adaptation when the other parameter is changed. However, this may cause the FFT analysis to fail.

This setting is only available if gating is not enabled (otherwise the gate determines the RBW automatically).

For details, see Chapter 8.1.1, "Fundamentals of FFT Analysis", on page 359.

"Record length controlled"	The record length remains constant. If not enough samples are available for the selected RBW, the RBW is decreased.
"RBW controlled"	The RBW is not adapted, i.e. remains as defined. The required acquisition time for this RBW is indicated. If necessary and possible, the record length is extended to acquire the required number of samples.

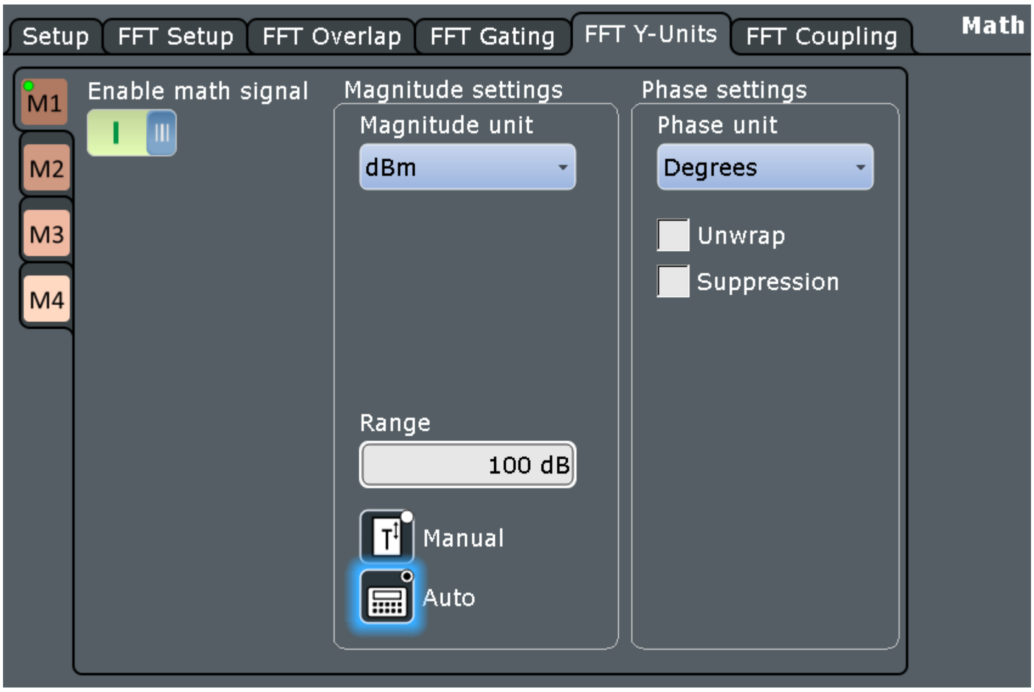
Remote command:
`CALCulate:MATH<m>:FFT:GATE:COUPling` on page 1285

Required acquisition time
The required acquisition time is calculated for the defined RBW value if "RBW constant" is selected, and is displayed for information only. If the required acquisition time is not available (e.g. because acquisition has already been stopped), an error message is displayed in the FFT Setup tab indicating that not enough samples are available for the defined RBW.

Remote command:
`TIMebase:RACTime?` on page 1285

8.1.3.4 FFT Y-Units

Access: [MATH] > "FFT Y-Units"



Enable Math Signal....375
Magnitude unit....375
Reference level....376
Vertical scaling mode (Manual/Auto)....376
Vertical maximum....376
Vertical range....376
Range....376
Phase unit....376
Unwrap....377
Suppression....377
Threshold....377

Enable Math Signal

If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:

`CALCulate:MATH<m>:STATe` on page 1190

Magnitude unit

Defines the scaling of the y-axis. The display values are valid for 50Ω termination impedance.

"Linear" Linear scaling; displays the RMS value of the voltage.

"dBm" Logarithmic scaling; related to 1 mW.

"dB" Logarithmic scaling; related to reference level.

"dBµV, dBmV, dBV"

Logarithmic scaling; related to voltage 1 µV, 1 mV, 1 V, respectively

"dBps, dBns, dBµs, dBms, dBs"
Logarithmic scaling; related to time 1 ps, 1 ns, 1 µs, 1 ms, 1 s, respectively.

"dBHz, dBkHz, dBMHz, dBGz"
Logarithmic scaling; related to frequency.

"dBA, dBmA, µdBA"
Logarithmic scaling; related to current.

Remote command:
`CALCulate:MATH<m>:FFT:MAGNitude:SCALe` on page 1288

Reference level

Defines the reference level for dB scaling.

Remote command:
`CALCulate:MATH<m>:FFT:MAGNitude:LEVel` on page 1288

Vertical scaling mode (Manual/Auto)

By default, the vertical scale is adapted to the current measurement results automatically to provide an optimal display. However, if necessary, you can define scaling values manually to suit your requirements.

Note: When you change the scaling values manually using the "Scale" rotary knob, the scale mode is set to "Manual" temporarily. When you edit the math function, scaling is automatically set back to "Auto" mode. "Manual" mode is only maintained during math function changes if you select it yourself.

"Manual" Enter the required values for "Vertical scale" and "Vertical offset". For FFT, set "Vertical range" and "Vertical maximum".

"Auto" "Vertical scale" and "Vertical offset" are read-only. For FFT, only the "Vertical maximum" is read-only.

Vertical maximum

Defines the maximum value on y-axis for spectrum displays. Only available for "Manual" scale mode.

Vertical range

Defines the range of FFT values to be displayed.

Remote command:
`CALCulate:MATH<m>:VERTical:RANGe` on page 1192

Range

Defines the vertical value range in spectrum mode.

Remote command:
`CALCulate:MATH<m>:FFT:MAGNitude:RANGe` on page 1288

Phase unit

Defines the scaling unit for phase display.

- Radians
- Degrees

Remote command:
CALCulate:MATH<m>:FFT:PHASe:SCALe on page 1289

Unwrap
If enabled, phase shifts due to a limitation of the value range are eliminated.

Remote command:
CALCulate:MATH<m>:FFT:PHASe:UNWRap on page 1290

Suppression
Enables noise suppression. Phase calculation is restricted to frequencies with a minimum magnitude, the threshold value.

Remote command:
CALCulate:MATH<m>:FFT:PHASe:SUPPression on page 1289

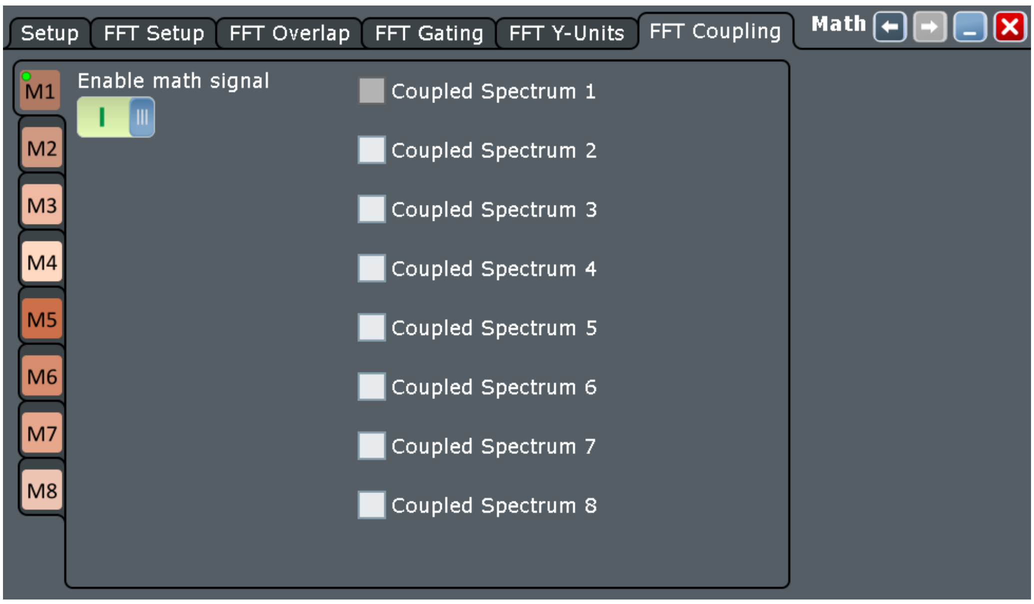
Threshold
Defines the minimum frequency magnitude for which phases are calculated. This setting is only available if "Suppression" is enabled.

Remote command:
CALCulate:MATH<m>:FFT:PHASe:THReshold on page 1290

8.1.3.5 FFT Coupling

Access: [Math] > "FFT Coupling"

Up to four spectrum displays can be shown simultaneously, one for each math waveform. The settings for one or more spectrums can be coupled. Thus, if any FFT setting for any of the coupled spectrums are changed, they are changed for all coupled spectrums.



Enable Math Signal

If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:
`CALCulate:MATH<m>:STATe` on page 1190

Coupled Spectrum 1/2/3/4/5/6/7/8

Copies the current FFT settings of the selected math waveform (M1...M8) to the other selected math waveforms, and couples those waveforms. Thus, if any FFT setting for any of the coupled spectrums are changed, they are changed for all coupled spectrums.

Two different sets of spectrums can be coupled at the same time, for instance "Spectrum 1" can be coupled to "Spectrum 2", while "Spectrum 3" is coupled to "Spectrum 4".

Note that the formula of the coupled math waveforms is not changed. If necessary, you must select an FFT function for the math waveform manually before the FFT settings of the coupled waveform are applied. See Chapter 6.3.1, "Displaying Math Waveforms", on page 257.

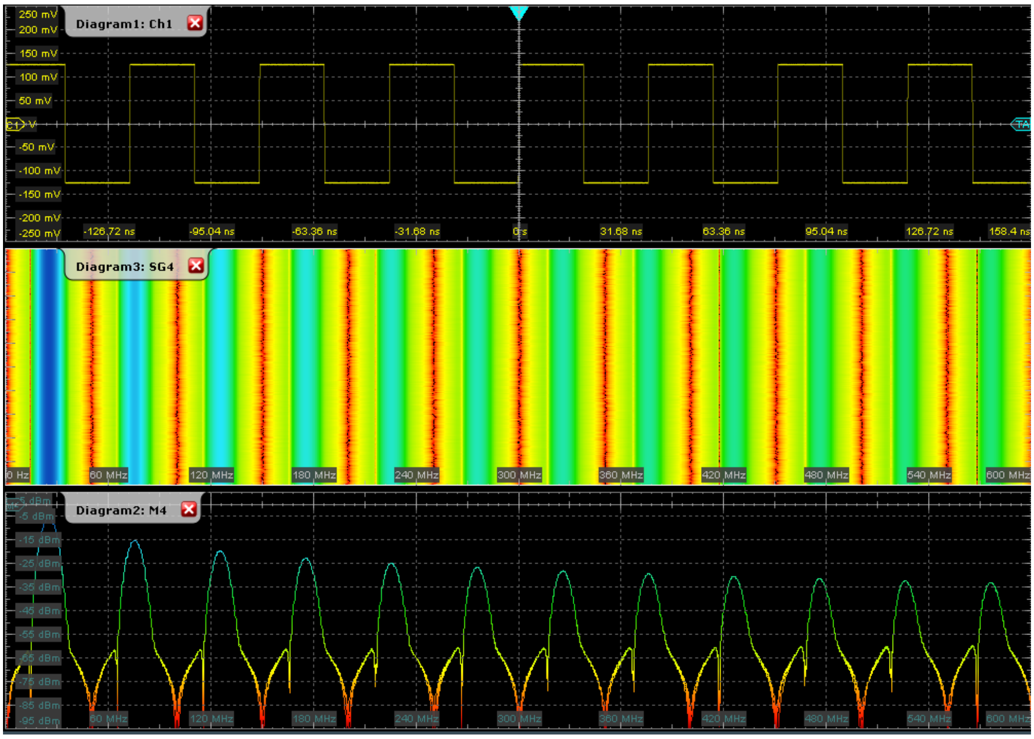
Remote command:
`CALCulate:MATH<m>:FFT:COUPled:WITH<1..8>` on page 1290

8.2 Spectrum Analysis (Option R&S RTE-K18)

This chapter describes the Spectrum Analysis option R&S RTE-K18.

8.2.1 Spectrogram Display

The Spectrum Analysis option provides a new diagram for spectrum waveforms: a spectrogram. When you enable a spectrogram, three windows are displayed: the power vs. time diagram at the top, the spectrogram in the middle (labeled "SG") and the power vs. frequency (=spectrum) diagram at the bottom.



A spectrogram shows how the spectral density of a signal varies over time. The x-axis shows the frequency, the y-axis shows the time. A third dimension, the power level, is indicated by different colors. Thus you can see how the strength of the signal varies over time for different frequencies.

The spectrogram is updated with each data acquisition, from top to bottom, so that the most recent trace is at the bottom. Up to two time lines can be shown at a specified position so that you can analyze the spectrum at a specific point in time.

The spectrum diagram indicates the power vs. frequency values for a single data acquisition. If a time line is enabled, the spectrum shows the results at the selected time. Otherwise, the spectrum shows the results of the most recent data acquisition.

8.2.2 Spectrum Analysis Functions

In addition to spectrograms, the Spectrum Analysis option also provides some new automatic measurements based on spectrum waveforms.

- A peak list measurement detects all peaks above a user-definable threshold and optionally indicates the peaks in the spectrum diagram.

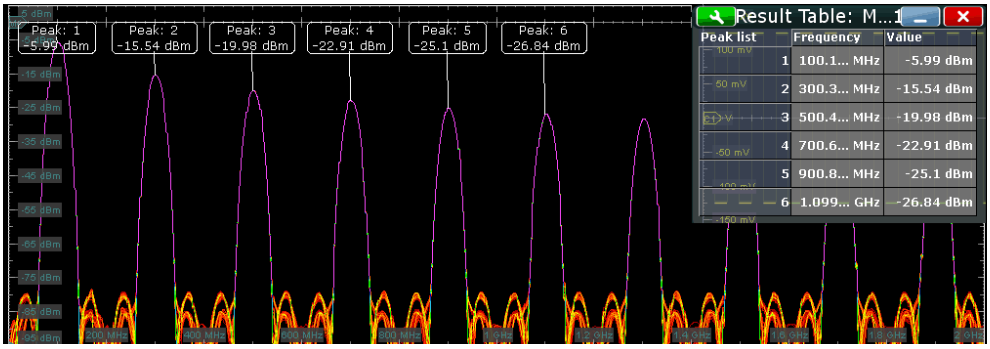


Figure 8-1: Peak list with labels for spectrum waveform

For a description of the measurement settings, see Chapter 7.2.7.2, "Settings for Spectrum Measurements ", on page 326.

- The THD measurements are an extension to the basic THD measurement. See Chapter 7.2.7, "Spectrum Measurements", on page 324 for details.
- Cursor measurements on spectrum waveforms provide easy center definition and peak search functions, see Chapter 7.1.1.2, "Cursor Measurements on Spectrum Waveforms", on page 285.

8.2.3 Configuring Spectrograms

Spectrograms are only available if the Spectrum Analysis option R&S RTE-K18 is activated.

1. Tap the "Spectrogram" icon on the toolbar.
2. On the signal bar, select the source of the spectrogram.

   A spectrogram diagram is displayed. A new signal icon for the spectrogram is displayed on the signal bar ("SGx").
   If the selected source is a channel waveform, an FFT is started, on which the spectrogram is created.
   Additional settings for time lines become available in the dialog box.
3. Optionally, to display a time line and thus mark a specific waveform in the spectrogram, select "Enable" for one of the two time lines.

   A small arrow icon labeled "T1" / "T2" indicates the position of the time line in the spectrogram.
   The spectrum diagram displays the results for the selected waveforms. A new signal icon is displayed on the signal bar for each time line ("SGxTL1|2").
4. To view the spectrum for each time line in a separate diagram, drag the signal icon for one time line to the diagram area and drop it.

   A new window is displayed for the selected time line, and the original diagram displays the other time line.

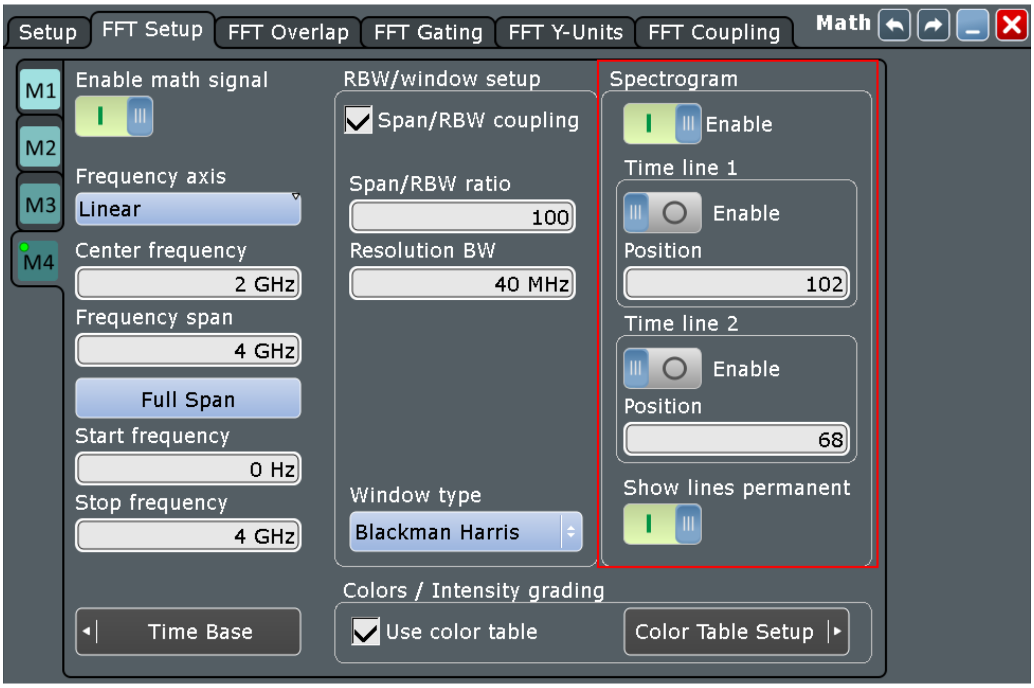
5. To view a different waveform from the spectrogram, move the time lines in the spectrogram.

8.2.4 Spectrogram Configuration Settings

Access: [Math] > "FFT Setup"

Spectrograms are only available if the option R&S RTE-K18 is activated. Furthermore, a math (FFT) waveform must be configured and enabled.

See Chapter 8.1.2, "Configuring Spectrum Waveforms", on page 363.



Enable....................381
Time line 1/2....................381
Show lines permanently....................382

Enable

Enables the spectrogram display.

If enabled, a new signal icon for the spectrogram is displayed on the signal bar ("SGx").

Remote command:
`CALCulate:MATH<m>:FFT:SPECtrogram:STATe` on page 1293

Time line 1/2

A time line marks a single spectrum in the spectrogram, that is: the power vs frequency results for the data acquired at a specific time. After enabling a time line, the results for that time are displayed in the spectrum diagram. A small arrow icon labeled "T1" / "T2" indicates the position of the time line in the spectrogram. A new signal icon is displayed on the signal bar for each time line ("SGxTL1|2").

You can enable and display two time lines at the same time. This allows you to compare the results at different times.

The position of the time line is defined by the index of the data acquisition in the history. How many acquisitions are available depends on the history settings.

See "Max. acquisition count" on page 277 and Chapter 6.4.2.1, "Viewer", on page 275.

Remote command:
`CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe` on page 1294
`CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition` on page 1293

Show lines permanently
Displays the spectrogram time lines in the diagrams until you disable this option.

If disabled, only the small arrow icons are permanently visible. The line is only displayed temporarily when you touch the arrow.

9 Mask Testing

9.1 About Mask Testing

Masks are used to determine whether the signal remains within specified limits, e.g. to uncover signal anomalies or test compliance and stability of digital signals. The limits are specified as "mask", which is laid over the input signal in the display. Thus you can easily detect where the signal violates the mask.

Mask testing with R&S RTE has only a minor impact on the acquisition rate, thus mask violations are detected fast and reliably.

With R&S RTE, you can define own masks easily. Specific actions can be executed when mask violations occur. For error analysis, you can stop the acquisition on a failed test and use the history view to look at the previous waveforms.

Mask test

A mask test consists of:

- Mask definition
- Waveform to be tested
- Fail criteria for test
- Actions to be taken on violation or successful completion

Mask Definition

A mask can be created in several ways:

- The individual mask points are defined, either on the touch screen or as numerical values. This mask type is called *user mask*.
  For details, see Chapter 9.2.2.1, "Mask Definition: User Mask", on page 387.
- The mask is derived from an existing waveform. This mask type is called *waveform mask*.
  For details, see Chapter 9.2.2.2, "Mask Definition: Waveform Mask", on page 390.

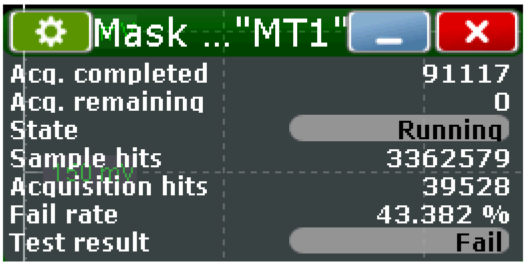
Fail Criteria for Testing

The fail criteria for a mask test is set by two parameters: "Fail condition" and "Violation tolerance". Fail condition defines if sample hits or the number of acquisitions with sample hits are considered. Violation tolerance sets the number of tolerable sample hits or acquisition hits. A test has failed if the number of sample hits or acquisition hits exceeds the limit of violation tolerance hits.

See also: "Fail condition, Violation tolerance" on page 386.

9.1.1 Results of a Mask Test

The result box of a mask test shows the following test results:



Acq. completed

Number of tested acquisitions.

Remote command:

MTESt:RESult:COUNt:WAVeforms? on page 1312

Acq. remaining

Remaining acquisitions until "Average count / Nx Single count" is reached.

The value is useful if you test a specified number of acquisitions with action "Stop acquisition" on violation. Also if the acquisition has been stopped manually before the required number of acquisitions has been acquired.

See also: Chapter 9.3.4, "Running a Mask Test", on page 403.

Remote command:

MTESt:RESult:COUNt:REMaining? on page 1313

State

Shows if the test has been completed. The state is set to "Finished" when "Nx Single count" acquisitions are tested and the number of "Acq. remaining" is 0. as long as the number of tested acquisitions is less the "Nx Single count" number, the state is "Running".

If you run the acquisition with [RUN CONT], or the number of played history acquisitions exceeds "Nx Single count", the mask testing is performed according to fail criteria settings independently of the test state. The testing is not stopped when the state is set to "Finished".

Remote command:

MTESt:RESult:STATe? on page 1312

Sample hits

Number of samples that hit the mask.

Remote command:

MTESt:RESult:COUNt:FAILures? on page 1313

Acquisition hits

Number of acquisitions that contained at least one sample hit.

Remote command:

MTESt:RESult:COUNt:FWAVeforms? on page 1313

Fail rate

Ratio of acquisition hits to the number of tested acquisitions.

Remote command:
`MTESt:RESult:FRATe?` on page 1314

Test result
A test has failed if the number of sample hits or acquisition hits exceeds the limit of "Violation tolerance" hits.

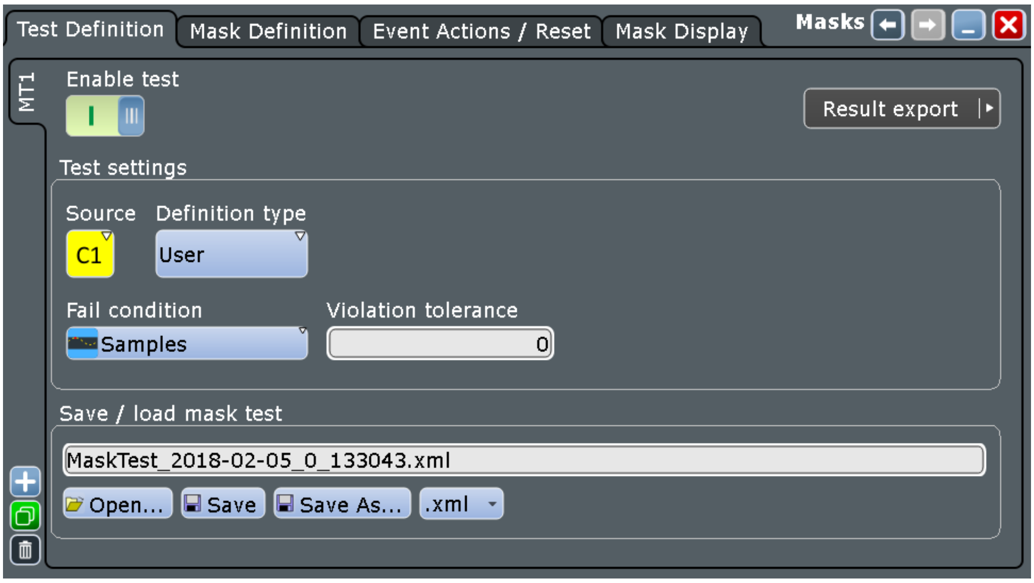
Remote command:
`MTESt:RESult[:RESult]?` on page 1312

9.2 Mask Test Settings

9.2.1 Test Definition

Access: [MASKS] > "Test Definition"

The "Test Definition" tab provides all settings for the mask test itself: the waveform to be tested, pass/fail conditions, and saving/loading the mask definition.



The content of the "Test Definition" tab depends on the selected definition type. If "Waveform" is selected, the main mask settings can be set directly on the "Test Definition" tab. For a description of these settings, see Chapter 9.2.2.2, "Mask Definition: Waveform Mask", on page 390.

Make sure that the correct "Mask Test" tab is selected on the left side before you enter the settings.

Remote commands:

MTESt:ADD on page 1295

MTESt:REMove on page 1295

Enable test

Activates and deactivates the mask test. If the acquisition is running, the test starts immediately. Otherwise, the test starts when acquisition is started.

The testing is stopped when acquisition is stopped, or if a stop action is configured with Stop acq..

Closing the result box also disables the mask test.

Remote command:
`MTESt[:STATe]` on page 1295

Source

Selects the waveform to be tested against the mask. All channel waveforms can be tested.

Remote command:
`MTESt:SOURce` on page 1296

Definition type

Sets the method of mask definition.

"User" The mask is created manually by tapping the mask points on the touch screen and/or by entering the numerical x- and y-values of the mask points.

"Waveform" The mask is created from an existing waveform. The waveform builds the upper and lower limit line of the mask, and the limits are moved and stretched. The result is a tolerance tube around the waveform that is used as mask.

Remote command:
`MTESt:CTYPe` on page 1297

Fail condition, Violation tolerance

The fail criteria for a mask test is set by two parameters: "Fail condition" and "Violation tolerance".

"Fail condition" defines the kind of hits to be considered for test evaluation:
- "Samples": Considers the number of samples that hit the mask.
- "Acquisitions": Considers the number of acquisitions that contain at least one sample hit. It is not relevant how many samples hit the mask in that acquisition.

"Violation tolerance" sets the number of tolerable sample hits or acquisition hits.

A test has failed if the number of sample hits or acquisition hits exceeds the limit of violation tolerance hits.

Example:

The example test has failed when the sixth acquisition violated the mask.

Remote command:
`MTESt:CONDition` on page 1296
`MTESt:TOLerance` on page 1297

Save / load mask test
Provides all functions to store and recall a mask test. The mask definition, defined actions and fail conditions are stored in an R&S RTE-specific .xml file.

"Load, Save"	Recalls or stores the specified file.
"Open, Save As"	Opens a dialog box where you can select the directory the file name. See also: Chapter 11.7, "File Selection Dialog", on page 471.
"Delete"	Opens a dialog box where you can select the file to be deleted.

Remote command:
`MTESt:FILE:NAME` on page 1298
`MTESt:FILE:SAVE` on page 1298
`MTESt:FILE:OPEN` on page 1298
`MTESt:FILE:DELete` on page 1298

9.2.2 Mask Definition

Access: [MASKS] > "Mask Definition"

With mask definition, you define the shape of the mask - the form and position of its limit lines. The content of the "Mask Definition" tab depends on the selected "Definition type": "User" or "Waveform".

The "Definition type" is a common setting on the top of the tab, see "Definition type" on page 386.

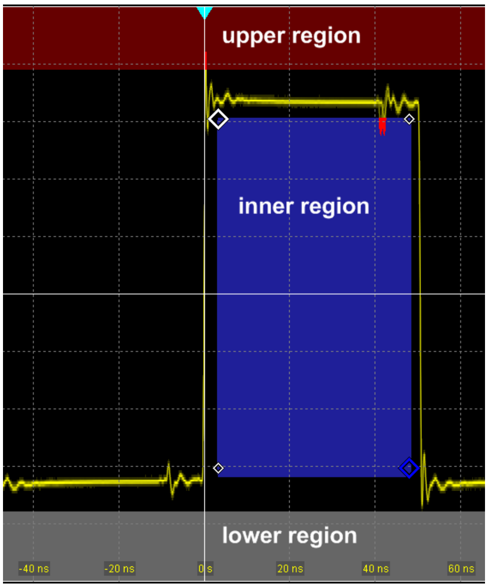
Below, you find the specific settings:

9.2.2.1 Mask Definition: User Mask

Access: [MASKS] > "Mask Definition" > "Definition type" = "User"

A user mask is defined by entering the time and voltage values for all corner points of the mask segments. A user mask has at least one segment. Complex masks can have up to 16 segments.

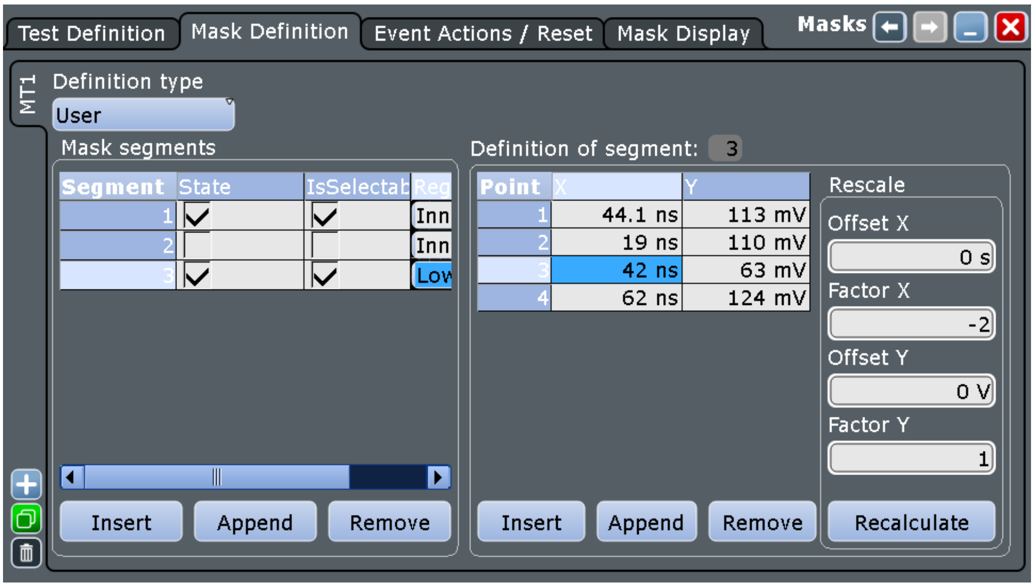
An inner segment is an area defined by three or more points. Upper and lower segments limit the signal on top and bottom of the screen. They are defined by a line, the region above or below the line is set automatically as mask segment.



Alternatively, you can set the corner points on the touch screen and adjust the values in the "Mask Definition" tab.

To save the mask, select the "Test Definition" tab and save the mask test.

Settings overview:



Make sure that the correct "Mask Test" tab is selected on the left side before you enter the settings.

Mask segments

Defines the number and state of mask segments for the selected mask test. Here you can:

- Insert a new segment before the selected segment.

- Append a new segment at the end of the list.
- Remove the selected mask segment from the list.
- Select the region that builds the mask.
  – Inner region: the segment points form a closed geometrical shape, which is the mask segment.
  – Upper region: the segment points are connected to a line, the display area above this line is the mask segment.
  – Lower region: the segment points are connected to a line, the display area below this line is the mask segment.
- Enable and disable the mask segments individually. Disabled segments are not considered by running tests.

Remote command:
`MTESt:SEGMent:STATe` on page 1299
`MTESt:SEGMent:ADD` on page 1299
`MTESt:SEGMent:REMove` on page 1300
`MTESt:SEGMent:INSert` on page 1300
`MTESt:SEGMent:REGion` on page 1300
`MTESt:SEGMent:COUNt?` on page 1300

Definition of segment
The number of the selected segment is shown above the table. In the definition table, the individual points of the selected mask segment are listed with exact horizontal and vertical numerical coordinates. Here you can:

- Insert a new point before the selected point.
- Append a new point at the end of the list.
- Remove the selected point from the list.
- Change the x- and y-values of each point. To scale or move the complete segment, use offset and factor values, see Rescale.

Remote command:
`MTESt:SEGMent:POINt:ADD` on page 1301
`MTESt:SEGMent:POINt:REMove` on page 1301
`MTESt:SEGMent:POINt:INSert` on page 1301
`MTESt:SEGMent:POINt:X` on page 1302
`MTESt:SEGMent:POINt:Y` on page 1302
`MTESt:SEGMent:POINt:COUNt?` on page 1302

Rescale
You can rescale and move mask segments by numerical input of factors and offsets. The values change the selected mask segment and take effect on "Recalculate".

Offset X ← Rescale
Moves the mask segment horizontally. The specified offset is added to the x-values of all points of the selected mask segment.

To take effect, tap "Recalculate".

Remote command:
`MTESt:SEGMent:RESCale:XOFFset` on page 1303

Factor X ← Rescale
Stretches or compresses the selected mask segment in horizontal direction. The x-values of all points of the selected mask segment are multiplied with this factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.

To take effect, tap "Recalculate".

Remote command:
`MTESt:SEGMent:RESCale:XFACtor` on page 1303

Offset Y ← Rescale
Moves the mask segment vertically. The specified offset is added to the y-values of all points of the selected mask segment.

To take effect, tap "Recalculate".

Remote command:
`MTESt:SEGMent:RESCale:YOFFset` on page 1304

Factor Y ← Rescale
Stretches or compresses the selected mask segment in vertical direction. The y-values of all points of the selected mask segment are multiplied with this factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.

To take effect, tap "Recalculate".

Remote command:
`MTESt:SEGMent:RESCale:YFACtor` on page 1303

Recalculate ← Rescale
Multiplies and adds the given x- and y-factors and offsets to the coordinates of all points of the selected mask segment.

Remote command:
`MTESt:SEGMent:RESCale:RECalculate` on page 1303

9.2.2.2 Mask Definition: Waveform Mask

Access: [MASKS] > "Mask Definition" > "Definition type" = "Waveform"

A waveform mask is created from an existing waveform. The waveform builds the upper and lower limit line of the mask, and the limits are moved and stretched. The result is a tolerance tube around the waveform that is used as mask.

During mask testing using a waveform mask, the record length is limited to 1 MSample.

The source for a waveform mask is a reference waveform. The reference waveform can be defined before mask definition, or loaded from a file, or it is created from the waveform to be tested.

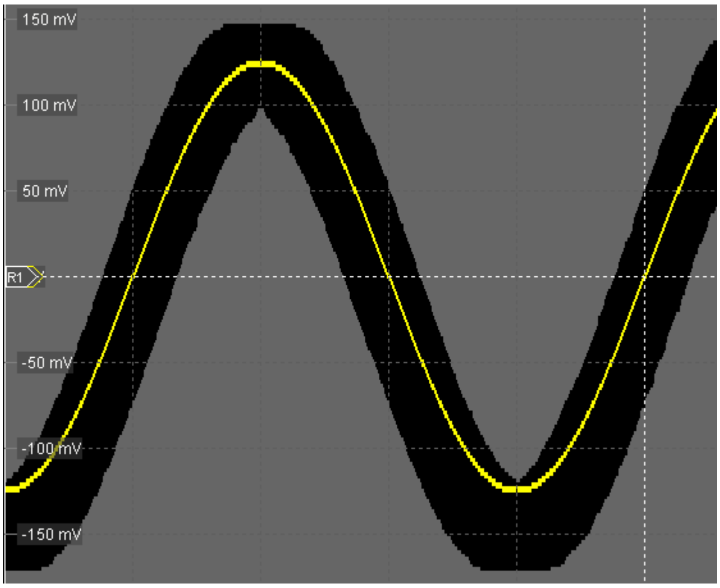
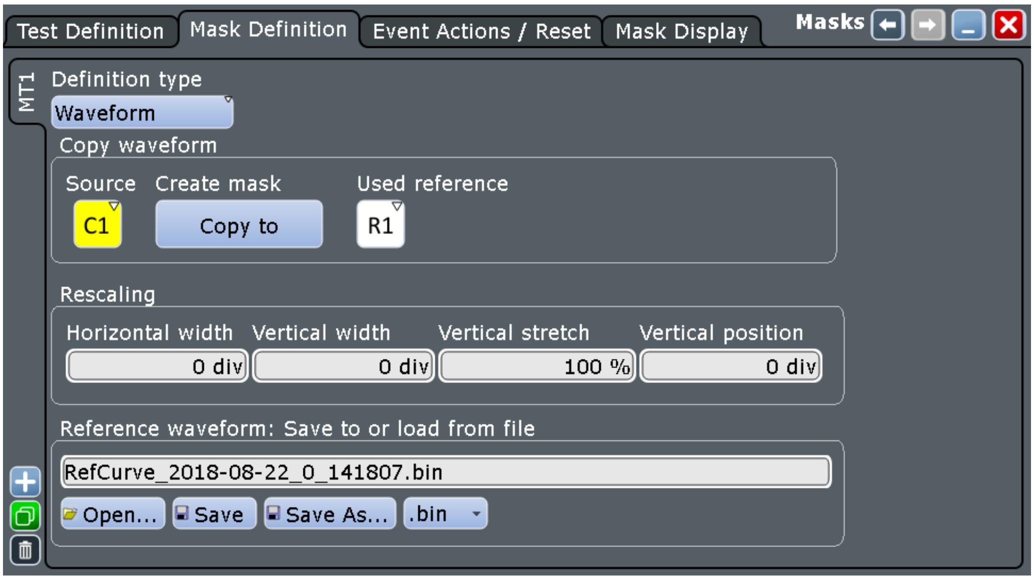


Figure 9-1: Waveform mask

Settings overview:



Common settings:

- "Definition type" on page 386
- "Source" on page 386

Create mask

Creates the upper and lower mask limit from the selected reference waveform. If the reference waveform was not defined before, it is created automatically from the mask test "Source" waveform which is selected in the "Test Defintion" tab.

Remote command:

`MTESt:WFMLupdate` on page 1305

Used reference

Sets the reference waveform from which the mask is created.

The reference waveform can be created before with "Reference Waveform Setup", or loaded from a file in the lower part of the dialog box. If the reference waveform was not defined before mask definition, it is created automatically from the mask test "Source" waveform.

Remote command:
`MTESt:REFWfm` on page 1304

Horizontal width

Sets the width of the mask in horizontal direction. The specified number of divisions is added to the positive x-values and subtracted from the negative x-values of the mask limits in relation to the source waveform of the mask. The overall mask width is twice the specified horizontal width.

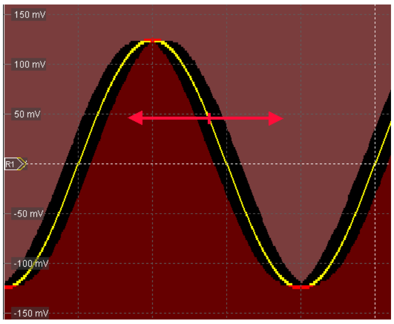


Figure 9-2: Waveform mask with horizontal width = 0.2 div

Remote command:
`MTESt:WFMRescale:XWIDth` on page 1305

Vertical width

Sets the width of the waveform mask in vertical direction. The specified number of divisions is added to the y-values of the upper mask limit and subtracted from the y-values of the lower mask limit. Thus, the upper half of the mask is pulled upwards, the lower half is pulled down, and the overall height of the mask is twice the vertical width.

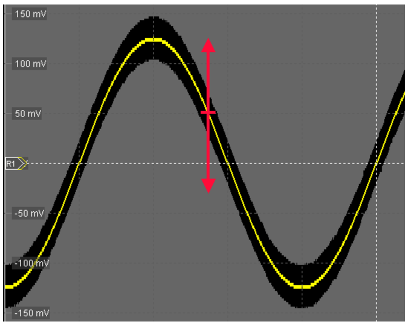


Figure 9-3: Waveform mask with vertical width = 0.5 div

Remote command:
`MTESt:WFMRescale:YWIDth` on page 1305

Vertical stretch

Sets the vertical scaling to stretch the mask in y-direction. The scaling axis is the horizontal line through the lowest value of the lower mask limit. Values > 100% stretch the mask, and values < 100% compress it.

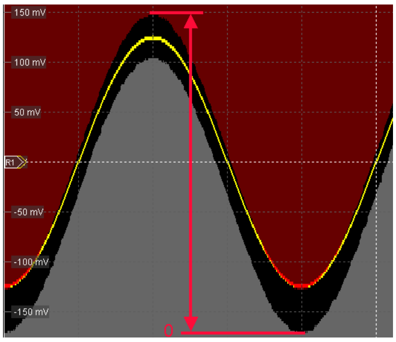


Figure 9-4: Waveform mask with vertical width = 0.5 div, vertical position = -0.5 div, vertical stretch = 110%

Remote command:
MTESt:WFMRescale:YSTRetch on page 1306

Vertical position

Moves the mask vertically within the display.

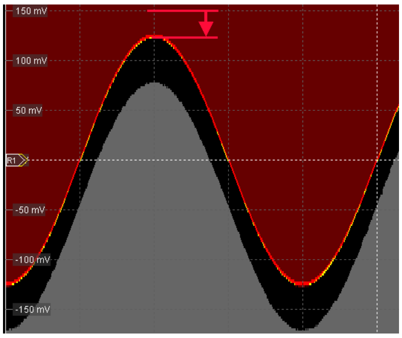


Figure 9-5: Waveform mask with vertical width = 0.5 div and vertical position = -0.5 div

Remote command:
MTESt:WFMRescale:YPOSition on page 1306

Reference waveform: save to or load from file

Loads the waveform from the selected file to the "Reference" and creates the mask immediately.

See also: "Save to or load from file" on page 255.

9.2.3 Event Actions /Reset

Access: [MASKS] > "Event Actions /Reset "

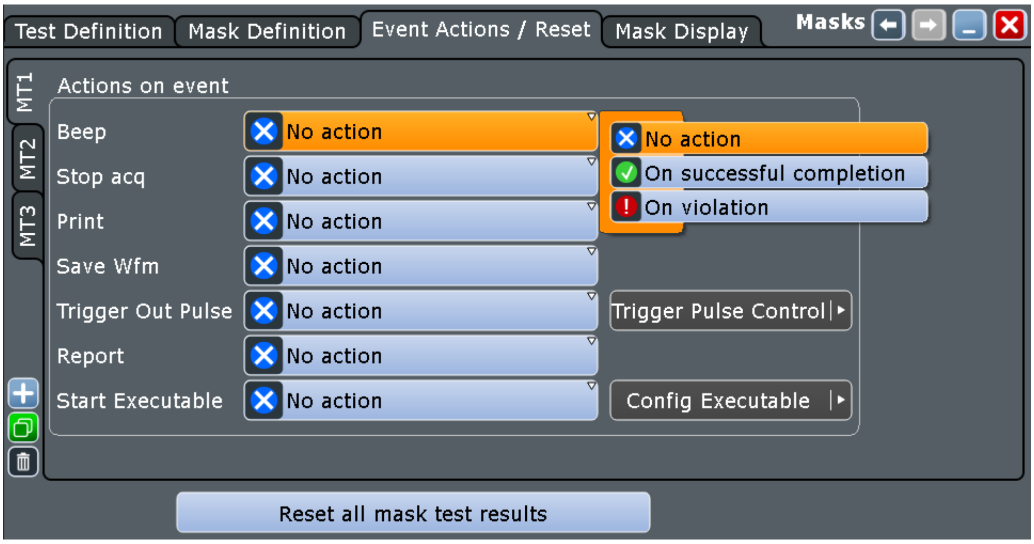
The settings in this tab define what happens when the mask test has failed or when it has passed successfully. Furthermore, you can reset all totals and results in the "Mask Test" result boxes.

Most actions can be initiated either on failure or on success:

- On violation
  The action is initiated when the fail criteria is fulfilled.
- On successful completion
  The action is initiated when the [RUN N× SINGLE] acquisition has finished and the fail criteria is not fulfilled - the fail condition and violation tolerance limit have not been reached.

There are two usual test practices:

- Testing a defined number of waveforms against the mask and initiate an action when the acquisition cycle has been completed without failure:
  – Set the number of acquisitions to be tested: "Average count (N-single count)"
  – Start [RUN N× SINGLE]
- Testing a continuous acquisition or a defined number of waveforms against the mask and initiate an action when the fail criteria is fulfilled



Make sure that the correct "Mask Test" tab is selected on the left side before you enter the settings.

Beep

Generates a beep sound.

Remote command:

MTESt:ONViolation:BEEP on page 1307

Stop acq.

Stops the waveform acquisition on mask violation.

Remote command:
MTESt:ONViolation:STOP on page 1307

Print
Prints a screenshot including the mask test results to the printer defined in the "Print" dialog box (see Printer).

Remote command:
MTESt:ONViolation:PRINt on page 1307

Save Wfm
Saves the failed waveform as a reference waveform to the file specified in [FILE] > "Save/Recall" > "Waveform".

Remote command:
MTESt:ONViolation:SAVewaveform on page 1308

Trigger Out Pulse
Creates a pulse on the [EXT TRIGGER OUT] connector on mask violation or successful completion of the test cycle. The minimum time difference between two trigger out pulses is 30 ms because the instrument detects mask violation at display update. Events with a higher frequency are not captured completely.

If this event is enabled and the mask test is running, the trigger control option "Enable trigger out" is disabled. Thus, the trigger out pulse is provided only on mask test result but not when a trigger occurs. The pulse is provided always with the minimum delay of 800 ns, the "Delay" cannot be set.

Remote command:
MTESt:ONViolation:TRIGgerout on page 1308

Report
Creates and saves a report using the settings defined in "File" menu > "Report Setup".

Remote command:
MTESt:ONViolation:REPort on page 1308

Start Executable
Starts an external application. Tap "Config Executable" to set the application path and parameters.

See: Chapter 3.5, "External Application", on page 120.

Remote command:
MTESt:ONViolation:RUNexec on page 1308

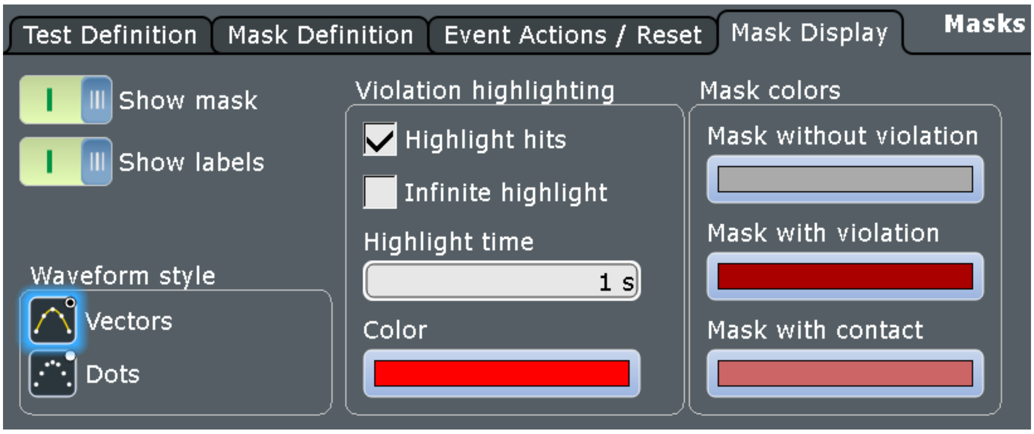
Reset
Clears all totals and results in all "Mask Test" result boxes.

Remote command:
MTESt:RST on page 1296

9.2.4 Mask Display

Access: [MASKS] > "Mask Display"

The "Mask Display" tab contains all settings for mask and hit display.



Show mask
Switches the display of all mask segments on or off.

Show labels
Switches the display of the mask test name on or off.

To change the name of the mask test, open the "Test Definition" tab, double-tap the mask test subtab and enter the new mask test name.

Remote command:
`MTESt:LABel` on page 1309
`MTESt:REName` on page 1309

Waveform style
See: "Style" on page 106.

Highlight hits
If selected, the mask hits are highlighted on the screen. You can define the color and the time of the hit display.

Remote command:
`MTESt:HIGHlight:STATe` on page 1310

Infinite highlight
If selected, the mask hits are highlighted for an unlimited period of time.

Remote command:
`MTESt:HIGHlight:INFinite` on page 1309

Highlight time
Sets the time how long the mask hits are highlighted.

Remote command:
`MTESt:HIGHlight:TIME` on page 1310

Color
Sets the color of samples that violated the mask.

Remote command:
`MTESt:COLor:MATCh` on page 1310

Mask without violation
Sets the color of masks segments that were not hit.

Remote command:
`MTESt:COLor:UNMatch` on page 1310

Mask with violation
Sets the color of mask segments the signal has entered into.

Remote command:
`MTESt:COLor:INTerior` on page 1311

Mask with contact
Sets the color of masks segments that were touched at the border. In this case, the resolution is not sufficient to detect if the mask was really hit or not. Zoom into the concerned area to see the actual result.

Remote command:
`MTESt:COLor:BORDer` on page 1311

9.3 Working with Masks

This chapter explains step-by-step how mask tests are setup and preformed. For the explanation of the individual settings, see Chapter 9.2, "Mask Test Settings", on page 385.

- Setting Up User Masks....................397
- Setting Up a Mask Test....................401
- Configuring the Mask and Hit Display....................402
- Running a Mask Test....................403
- Saving and Loading Masks....................404
- Mask Testing on History Acquisitions....................404

9.3.1 Setting Up User Masks

9.3.1.1 Creating User Masks

There are two ways to create a new mask:

- Graphical way by tapping the mask points on the touchscreen,
- Numerical entry of the x- and y-values of the mask points.

You can combine both methods. For example, at first you enter the mask quickly on the touchscreen, and then modify the point coordinates with precise values.

To create a mask graphically on the touch screen

1. Tap the "Masks" icon on the toolbar.

2. Tap the corner points of the mask segment on the touch screen.

   Tip: To create an exact rectangle, draw the diagonal of the rectangle on the screen.

3. To finish the segment and the mask definition, double-tap the last point.
4. Tap outside the mask to deselect the mask segment.

To create a mask numerically in the dialog box

The settings mentioned here are described in detail in Chapter 9.2.2.1, "Mask Definition: User Mask", on page 387.

1. Press the [MASKS] key on the front panel.
2. Select the "Mask Definition" tab.
3. Create a mask test:
   a) Tap the "+"-icon in the lower left corner.
   b) Enter a name for the new mask test.

   A new, empty tab for the mask test appears.
4. Adjust the horizontal and vertical units if necessary.
5. In the "Mask segments" area, tap "Insert" to create a new mask segment.
6. Set the corner points of the mask segment:
   a) In the "Definition of segment" area, tap "Insert".
      Point 1 appears.
   b) Tap the X-cell and enter the X-value of the point.
   c) Tap the Y-cell and enter the Y-value of the point.
   d) To insert the next point:
      - Tap "Insert" to add a point before the selected point.
      - Tap "Append" to add a point at the end of the list.
   e) Set the X- and Y-values for this point.
   f) Repeat the last two steps until you define all points.

9.3.1.2 Modifying User Masks

To change an existing mask definition, you can also use the graphical method on the touch screen, or the numerical way, or combine both.

With the graphical method, you can:

- Move, add, and delete segments
- Move and delete points

Adding points to an existing segment graphically is not possible.

With the numerical method, in the "Mask Definition" tab, you have all modification possibilities. You can delete and add points and segments, change the coordinates, and also stretch a segment, or move it by adding an offset.

To add a mask segment on the touch screen

1. Tap a mask segment of the mask test that you want to complement.
2. Tap the "Masks" icon on the toolbar.

3. Tap the corner points of the new mask segment on the touch screen.
4. To finish the segment and mask definition, double-tap the last point, or tap the "Select" icon on the toolbar.

To delete a mask segment on the touch screen

1. On the toolbar, tap the "Delete" icon.

2. Tap the mask segment you want to delete.

To delete a point on the touch screen

1. Tap the mask segment from which you want to delete a point.

   The selected segment is now in definition mode, shown with blue color.
2. On the toolbar, tap the "Delete" icon.

3. Tap the point you want to delete.

To move a segment on the touch screen

1. Drag&drop the segment to the new position.
2. Tap outside the mask to deselect the mask segment.

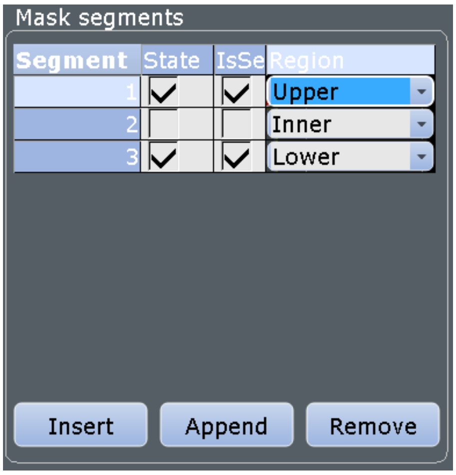
To move a point on the touch screen

1. Tap the mask segment that you want to change.
2. Drag&drop the point to the new position.
3. Tap outside the mask to deselect the mask segment.

To change the mask definition numerically

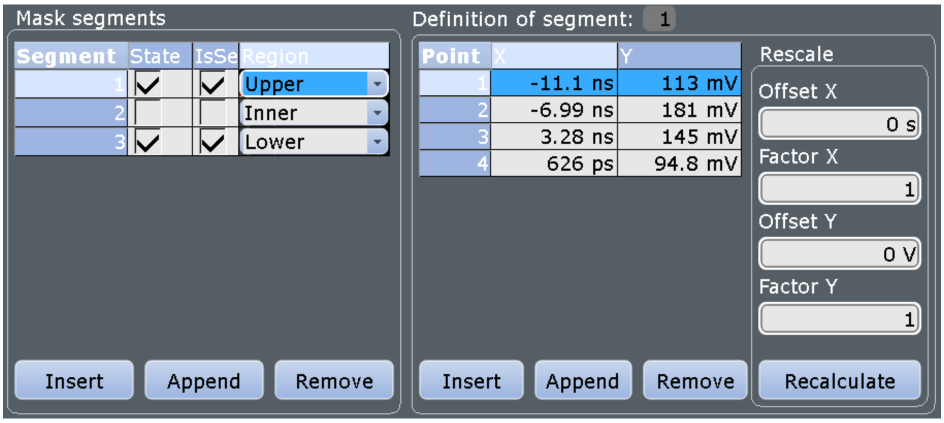
The settings mentioned here are described in detail in Chapter 9.2.2.1, "Mask Definition: User Mask", on page 387.

1. Press the [MASKS] key on the front panel.
2. Select the "Mask Definition" tab.
3. On the left, select the mask test for which you want to change the mask definition.
4. To add or delete a mask segment, tap the segment's row in the "Mask segments" table and tap the required button below:
   - "Insert": to add a new segment before the selected segment.
   - "Append": to add a new segment at the end of the list.
   - "Remove": to delete the selected mask segment from the list.



5. To add, delete, or move a point of a segment:
   a) Select the segment in the "Mask segments" table.
   b) Select the point in the "Definition of segment" table.
   c) To add or delete the selected point, use the buttons below the table.
      - "Insert": to add a new point before the selected point.
      - "Append": to add a new point at the end of the list.
      - "Remove": to delete the selected point from the list.

d) To move the selected point, change the X- and Y-values.



To rescale and move a mask segment

The settings mentioned here are described in detail in Chapter 9.2.2.1, "Mask Definition: User Mask", on page 387.

1. Press the [MASKS] key on the front panel.
2. Select the "Mask Definition" tab.
3. On the left, select the mask test for which you want to change the mask definition.
4. Select the required segment in the "Mask segments" table.
5. To stretch or compress the selected mask segment, enter the "X-Factor" for horizontal scaling and the "Y-Factor" for vertical scaling. The x-values and y-values of all points are multiplied with the corresponding factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.
6. To move the selected mask segment, enter the "X-Offset" for horizontal direction and the "Y-Offset" for vertical direction. The specified offset is added to the corresponding values of all points.
7. Tap "Recalculate" to perform the scaling and/or move.

9.3.2 Setting Up a Mask Test

In addition to the mask definition, the mask test contains further settings:

- the waveform to be tested
- the criteria for a failed test
- the actions to be taken if a test has failed or has been completed successfully

1. Press the [MASKS] key on the front panel.
2. Select the "Test Definition" tab.
3. Select the "Source" to be tested.

4. Set the conditions for a failed test:
   a) Fail condition: select if sample hits or the number of acquisitions with sample hits are considered.
   b) Violation tolerance: number of tolerable sample hits or acquisition hits.

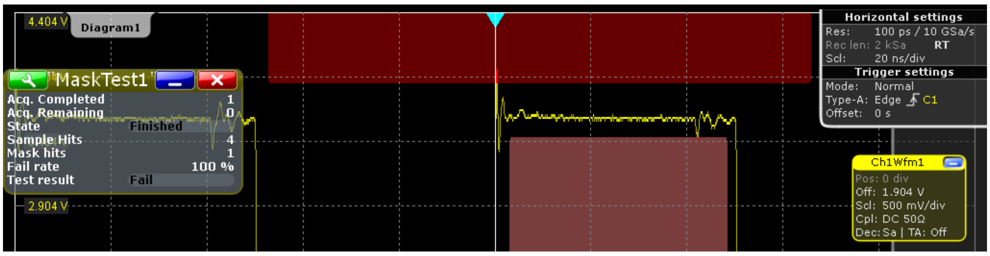
   A test has failed if the number of sample hits or acquisition hits exceeds the limit of violation tolerance hits.
5. Select the "Event Actions / Reset" tab.
6. For each action, select when the action will be executed:
   - "On violation" if the mask test has failed
   - "On successful completion"

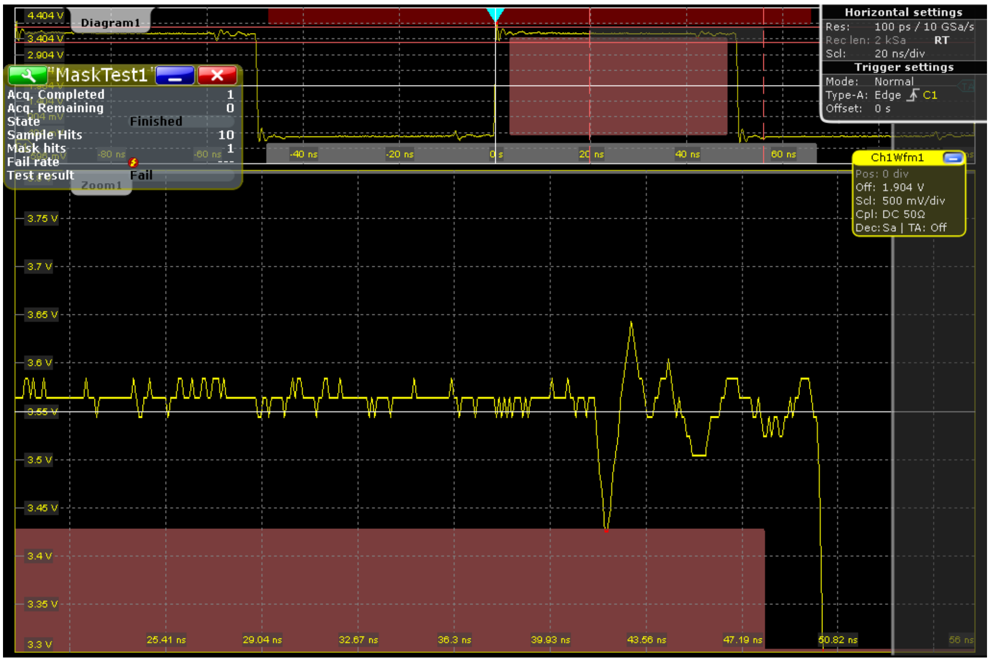
9.3.3 Configuring the Mask and Hit Display

The display of masks and mask violation is the same for all mask tests.

The settings mentioned here are described in detail in Chapter 9.2.4, "Mask Display", on page 396.

1. Press the [MASKS] key on the front panel.
2. Select the "Mask Display" tab.
3. Select "Show mask" to display the masks of all enabled mask tests on the screen.
4. Define how the sample hits are displayed:
   a) Select "Highlight hits" to display the sample hits.
   b) Set the "Highlight time" or "Infinite highlight".
      Set the "Color" of the sample hits.
5. Define the color of the masks segments depending on the violation state:
   - Mask without violation
   - Mask with violation
   - Mask with contact: This color shows that the edge of the mask segment was touched. In this case, the resolution is not sufficient to detect if the mask was really hit or not. Zoom into the concerned area to see the correct result.





9.3.4 Running a Mask Test

Before you can start a mask test, make sure that the mask setup is complete:

- The mask is defined, see Chapter 9.3.1.1, "Creating User Masks", on page 397 and Chapter 9.3.1.2, "Modifying User Masks", on page 398.
- The mask test is defined, see Chapter 9.3.2, "Setting Up a Mask Test", on page 401
- The mask display is configured, see Chapter 9.3.3, "Configuring the Mask and Hit Display", on page 402.

You can perform continuous testing or test a specified number of acquisitions.

1. Press the [MASKS] key on the front panel.
2. Select the "Test Definition" tab.
3. Select "Enable test".

   If the acquisition is running, the test starts immediately.
4. If the acquisition is not running, press [RUN CONT].

   The test starts and runs until you stop the acquisition or the stop action is executed if defined.
5. To test a specified number of acquisitions:
   a) Press the [ACQUISITION] key.
   b) Set the "Average count" to the number of acquisitions.
   See also:"Average count (N-single count)" on page 144
   c) Press [RUN N× SINGLE].

Note: If you run the acquisition with [RUN CONT], the state of the mask test is set to "Finished" when this number of acquisitions has been captured but the mask testing continues until the acquisition is stopped.

9.3.5 Saving and Loading Masks

Mask test definitions remain on the instrument until they are changed or deleted, or [PRESET] is performed. If you want to keep a mask test, you can save and reload them.

To save a mask

1. Press the [MASKS] key on the front panel.
2. Select the "Test Definition" tab.
3. To save the mask file in the current directory, change the file name if needed, and tap "Save".
   You can use the automatic file name generation, see Chapter 11.3, "Autonaming", on page 459.
4. To select the directory and enter the file name, tap "Save As".

To load a mask

1. To load the specified mask file, tap "Load."
2. To load the mask from a different file, tap "Open". Select the file from the file selection dialog box.

9.3.6 Mask Testing on History Acquisitions

In the same way as for running acquisitions, you can set up and perform the mask testing also on history waveforms.

The requirements for mask testing on history waveforms are also the same:
- The mask is defined, see Chapter 9.3.1.1, "Creating User Masks", on page 397 and Chapter 9.3.1.2, "Modifying User Masks", on page 398.
- The mask test is defined, see Chapter 9.3.2, "Setting Up a Mask Test", on page 401
- The mask display is configured, see Chapter 9.3.3, "Configuring the Mask and Hit Display", on page 402.

1. Perform and finish the acquisition.
2. Press [HISTORY].

3. In the quick-access "History" dialog box, tap "Play".

   The mask testing is performed on the complete history memory, starting with the oldest acquisition. The state of the mask test is set to "Finished" when "Nx Single count" acquisitions are tested.

For details on history, see Chapter 6.4, "History", on page 273.

10 Search Functions

Search functions allow you to detect and analyze specific events in the acquired data quickly and simply. You can search in various waveforms for several events at once. The search area can be limited by a gate.

The events that can be searched for are basically the same events you can trigger on. Thus, the search parameters are defined in the same way as the trigger conditions. The results are displayed in a result box and optionally shown in a zoom window.

10.1 Overview: Search Definition and Results

10.1.1 Search Definition

You can define up to 4 different searches and let them run simultaneously. For each search, you define the criteria, the parameters of each criterion, the gate, and the result display.

The instrument keeps the settings until the next preset. If you save a user-defined preset, the search settings are included in the preset.

Each search is configured in a separate tab and contains:

- *Search control*
  If you enable a search and run an acquisition, the search is performed continuously on the acquired data until acquisition is stopped.
  If acquisition is stopped and you enable a search, the data of the last acquisition is searched.
  Enabling the search zoom window disables the search, stops a running acquisition, and displays the search results of the last acquisition in the zoom window.
- *Source*
  Waveform that is searched for one or more events. You can search in analog and digital signals, math or reference waveforms, and tracks. Furthermore, search in decoded data of serial buses is possible.
- *Search criteria and parameters*
  Various search criteria are available, depending on the source. Most parameters known from trigger event definition can also be configured as search conditions. Unlike triggering, you can configure several event types to be searched for simultaneously.
  If the source is an FFT spectrum, you can perform a frequency marker search by using the cursor measurement and defining the peak excursion. See Chapter 7.1.3.3, "Peak Search ", on page 292.
- *Search gate*
  Searches can be performed on the entire waveform, or only on a defined area (gate). The gate can be coupled to an existing zoom.
  Gating is not available for searches on digital signals and serial buses.

- *Result presentation*
  For each search, you define how the search results are displayed: in a result table and/or in a search zoom window.
- *Noise rejection*
  Hysteresis for the selected source is defined for each search separately, in absolute or relative values.
  Noise rejection is not available for searches on serial buses.

Remote commands:
- SEARch:ADD on page 1315
- SEARch:REMove on page 1315

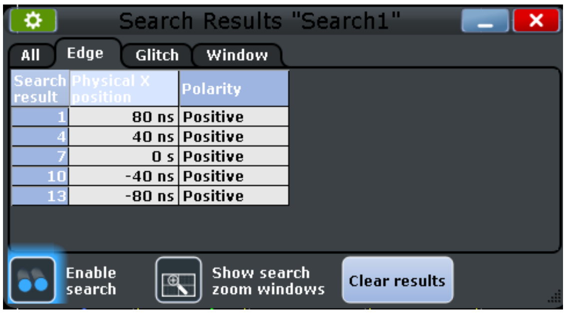
10.1.2 Search Results

The results are displayed in a "Search Results" box and optionally in a zoom window.

Search Results box

The results of each search are tabulated in a "Search Results" box.

If you search for several event types in parallel, the results are presented in several tabs - one for each search event and one for the combined results. Each tab contains a table with the position and, if available, further parameters for each result. The tables row can be sorted, and you can define a maximum number of table entries in the "Result Presentation" dialog box. As with all result boxes, you can minimize it, display it like a diagram, and define the default position.



If "Auto clear" is enabled in the "Result Presentation" dialog box, the instrument displays the search results of the last acquisition. If "Auto clear" is disabled, the first result of each acquisition is listed until the maximum number of entries in the table is reached.

Remote commands for result query:
- SEARch:RESult[:ALL]? on page 1352

Search zoom windows

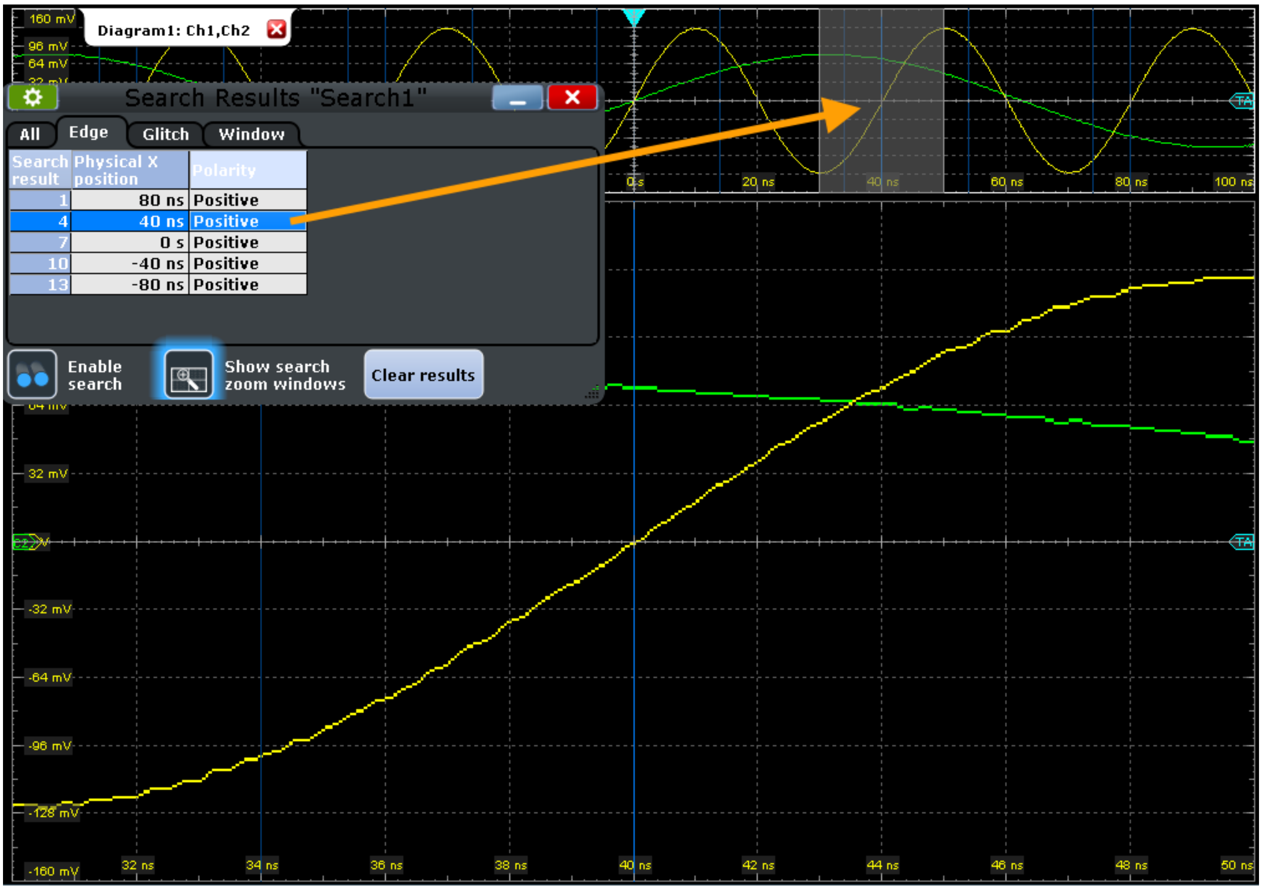
Search zoom windows allow you to analyze the search results in more detail. By default, the zoom is displayed for the selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.

Enabling the search zoom window disables the search, stops a running acquisition, and displays the search results of the last acquisition in the zoom window.

Navigating search results

If a search zoom window has been opened, it shows the first result that was found.

► To display the zoom of a specific search result, tap the result line in the result table to set the zoom to this event.



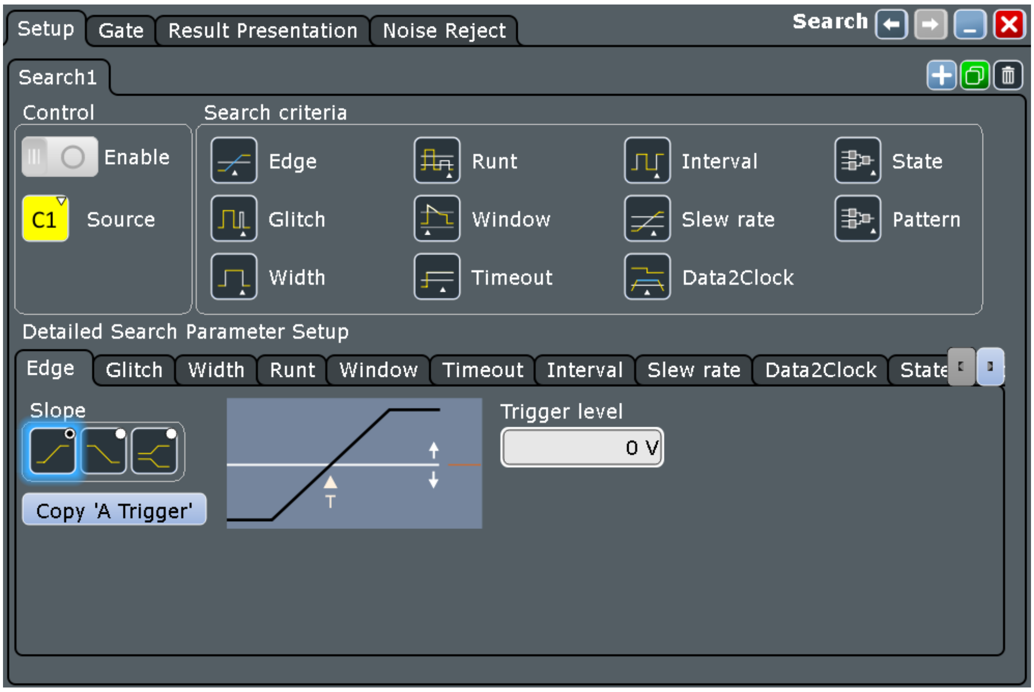
The zoom area in the source diagram moves to the selected result and the zoom is displayed.

You can change the size and the position of the search zoom area in the same way as a usual zoom. If you move the zoom area in the source diagram, the nearest search result is marked in the results table. See also: Chapter 6.1.1, "Methods of Zooming", on page 240.

10.2 Search Setup

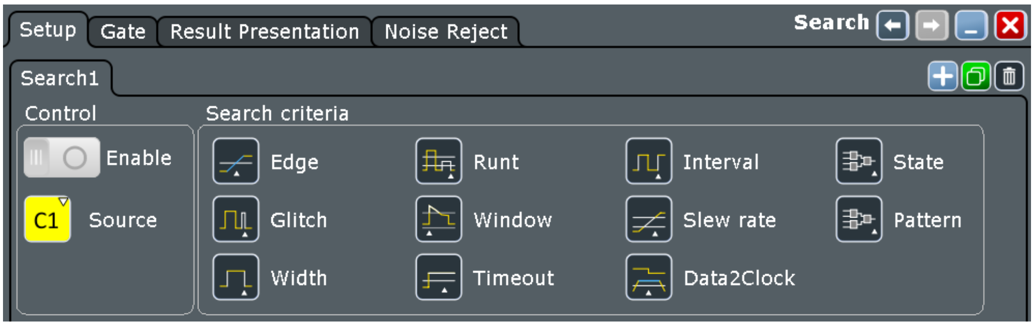
Access: [SEARCH] > "Setup" tab

The search setup includes the source selection, the selection of search events (criteria), event-specific search conditions, and search control.



10.2.1 Search Criteria

Access: [SEARCH] > "Setup" tab



Enable
If you enable a search and run an acquisition, the search is performed continuously on the acquired data until acquisition is stopped.

If acquisition is stopped and you enable a search, the data of the last acquisition is searched.

Remote command:
`SEARch:ONLine` on page 1316
`SEARch:ALL` on page 1316

Source
Defines the waveform to be searched. The source can be any analog and digital input signal, math or reference waveform, or track. While the instrument triggers only on real input signals, it can search also calculated and restored waveforms.

If the source is an FFT spectrum, you can perform a frequency marker search by using the cursor measurement and defining the peak excursion. See Chapter 7.1.3.3, "Peak Search ", on page 292.

For some serial protocol options, search on a serial bus is available. For details, see the relevant chapters of the "Protocol Analysis" chapter.

Depending on the selected source, different search criteria are available.

Remote command:
`SEARch:SOURce` on page 1315

Edge, Glitch, Width, Runt, Window, Timeout, Interval, Slew rate, Data2Clock, State, Pattern
Search criteria for analog and digital input signals, math and reference waveforms, and tracks. For searching on digital channels, only "Edge", "Width", "Timeout" and "Data2Clock" criteria are available.

Tap the icon to include or exclude the search criteria in the next search. You can enable several event types for simultaneous search.

Remote command:
`SEARch:TRIGger:EDGE[:STATe]` on page 1317
`SEARch:TRIGger:GLITch[:STATe]` on page 1317
`SEARch:TRIGger:WIDTh[:STATe]` on page 1317
`SEARch:TRIGger:RUNT[:STATe]` on page 1317
`SEARch:TRIGger:WINDow[:STATe]` on page 1317
`SEARch:TRIGger:TIMeout[:STATe]` on page 1317
`SEARch:TRIGger:INTerval[:STATe]` on page 1317
`SEARch:TRIGger:SLEWrate[:STATe]` on page 1317
`SEARch:TRIGger:DATatoclock[:STATe]` on page 1317
`SEARch:TRIGger:STATe[:STATe]` on page 1317
`SEARch:TRIGger:PATTern[:STATe]` on page 1317

Copy 'A Trigger'
Copies the trigger type-specific settings from the A-trigger configuration to the search settings. The source itself is not copied.

Remote command:
`SEARch:TRIGger:EDGE:ACOPy` on page 1318
`SEARch:TRIGger:GLITch:ACOPy` on page 1318
`SEARch:TRIGger:WINDow:ACOPy` on page 1318
`SEARch:TRIGger:WIDTh:ACOPy` on page 1318
`SEARch:TRIGger:RUNT:ACOPy` on page 1318
`SEARch:TRIGger:WINDow:ACOPy` on page 1318
`SEARch:TRIGger:TIMeout:ACOPy` on page 1318
`SEARch:TRIGger:INTerval:ACOPy` on page 1318
`SEARch:TRIGger:SLEWrate:ACOPy` on page 1318

SEARch:TRIGger:DATatoclock:ACOPy on page 1318
SEARch:TRIGger:STATe:ACOPy on page 1318
SEARch:TRIGger:PATTern:ACOPy on page 1318

10.2.2 Search Parameters

Most parameters available for trigger event definition can also be configured as search conditions. Each event type is defined in a separate subtab.

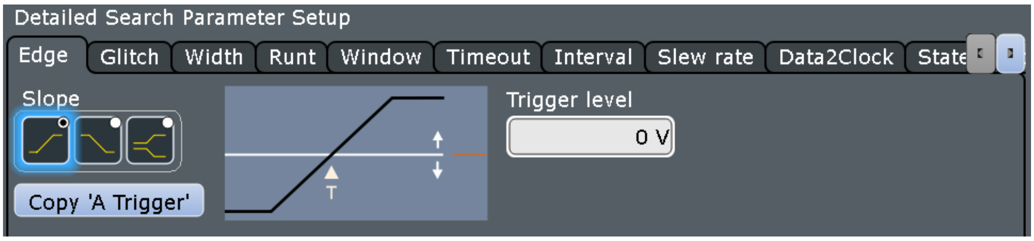
If the source is a spectrum, the instrument performs a frequency marker search.

For serial protocol options, search on a serial bus is available. These searches have protocol-specific search criteria. For details, see the relevant chapters of the "Protocol Analysis" chapter.

- Edge......411
- Glitch......412
- Width......412
- Runt......413
- Window......414
- Timeout......415
- Interval......416
- Slew Rate......417
- Data2Clock......417
- State......418
- Pattern......419

10.2.2.1 Edge

The edge search works the same way as the edge trigger.



Slope
Sets the edge type: rising edge ("Positive"), falling edge ("Negative"), or both.

Remote command:
SEARch:TRIGger:EDGE:SLOPe on page 1319

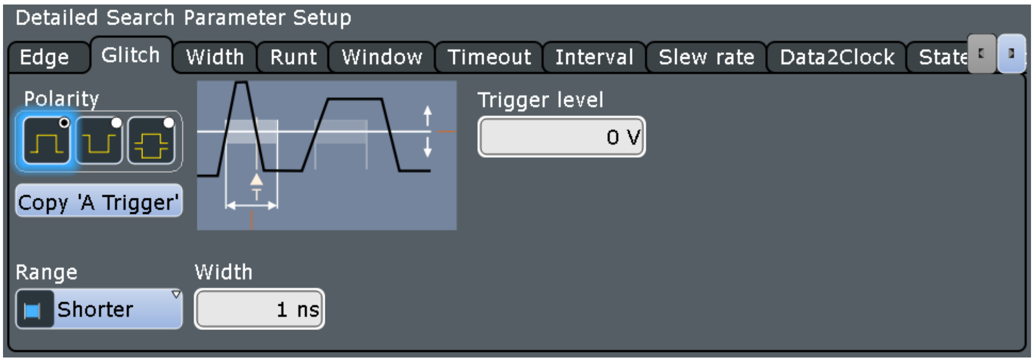
Trigger level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
SEARch:TRIGger:LEVel[:VALue] on page 1318

10.2.2.2 Glitch

The glitch search works the same way as the glitch trigger. To apply the trigger settings to search, tap Copy 'A Trigger'.

The glitch search is not available if the search source is a digital channel.



Polarity, Range, Width
See trigger settings:
- "Range" on page 207
- "Width" on page 207
- "Polarity" on page 207

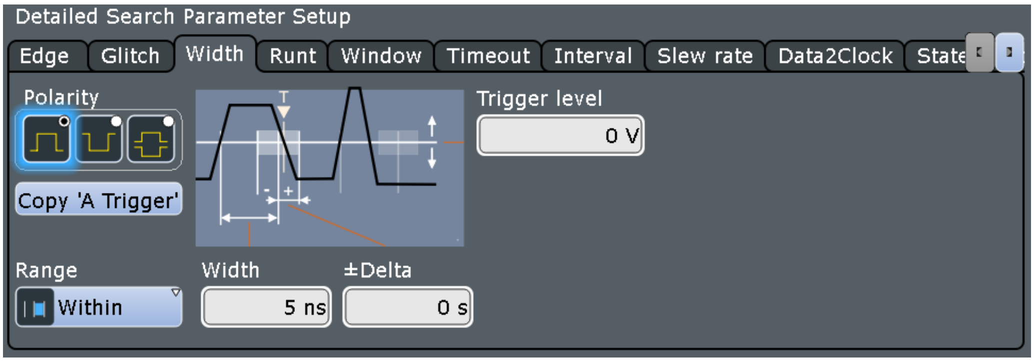
Remote command:
SEARch:TRIGger:GLITch:POLarity on page 1319
SEARch:TRIGger:GLITch:RANGe on page 1320
SEARch:TRIGger:GLITch:WIDTh on page 1320

Trigger level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
SEARch:TRIGger:LEVel[:VALue] on page 1318

10.2.2.3 Width

The width search works the same way as the width trigger. To apply the trigger settings to search, tap Copy 'A Trigger'.



Polarity, Range, Width, ±Delta

See trigger settings:

- "Polarity" on page 208
  While the width trigger can only analyze positive *or* negative polarity, searching for a width is also possible for both polarities at the same time ("Either").
- "Range" on page 208
- "Width" on page 208
- "±Delta" on page 209

Remote command:

`SEARch:TRIGger:WIDTh:POLarity` on page 1328
`SEARch:TRIGger:WIDTh:RANGe` on page 1328
`SEARch:TRIGger:WIDTh:WIDTh` on page 1329
`SEARch:TRIGger:WIDTh:DELTa` on page 1328

Trigger level

Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

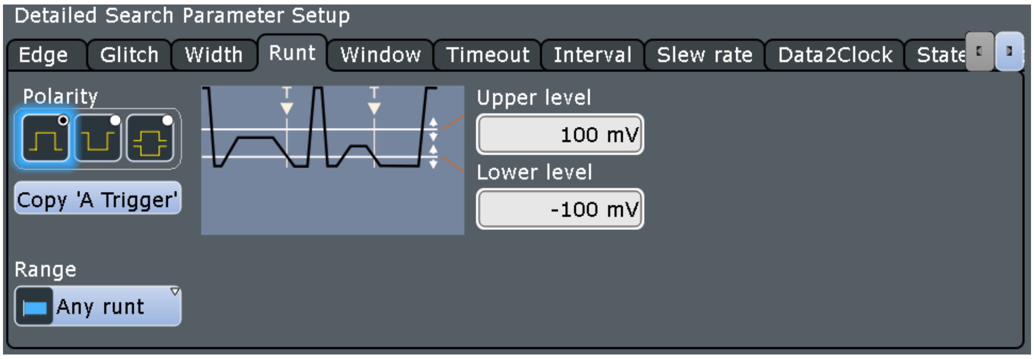
Remote command:

`SEARch:TRIGger:LEVel[:VALue]` on page 1318

10.2.2.4 Runt

The runt search settings are the same as the runt trigger settings. To apply the trigger settings to search, tap Copy 'A Trigger'.

The runt search is not available, if the search source is a digital channel.



Polarity, Range, Runt width, ±Delta
Time limit for the runt, see trigger settings:
- "Polarity" on page 207
- "Range" on page 210
- "Runt width" on page 210
- "±Delta" on page 210

Remote command:
`SEARch:TRIGger:RUNT:POLarity` on page 1322
`SEARch:TRIGger:RUNT:RANGe` on page 1323
`SEARch:TRIGger:RUNT:WIDTh` on page 1323
`SEARch:TRIGger:RUNT:DELTa` on page 1322

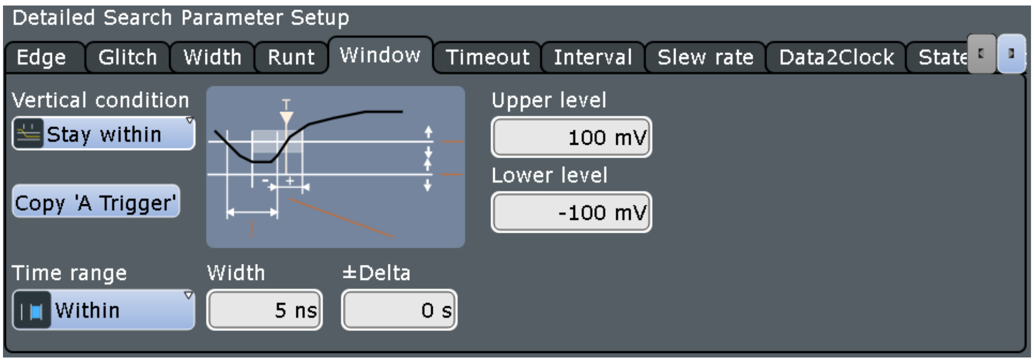
Upper level, Lower level
Set the upper and lower voltage thresholds. The amplitude of a runt crosses the first threshold twice in succession without crossing the second one.

Remote command:
`SEARch:TRIGger:LEVel:RUNT:UPPer` on page 1324
`SEARch:TRIGger:LEVel:RUNT:LOWer` on page 1324

10.2.2.5 Window

The window search settings are the same as the window trigger settings. This search type is not available if the search source is a digital channel. To apply the trigger settings to search, tap Copy 'A Trigger'.

The window search is not available if the search source is a digital channel.



Vertical condition
Defines the run of the signal relative to the window, see "Vertical condition" on page 211.

Remote command:
SEARch:TRIGger:WINDow:RANGe on page 1330

Time condition, Width, ±Delta
Set the time limit for the vertical condition, see:
- "Time condition" on page 211
- "Width" on page 212
- "±Delta" on page 212

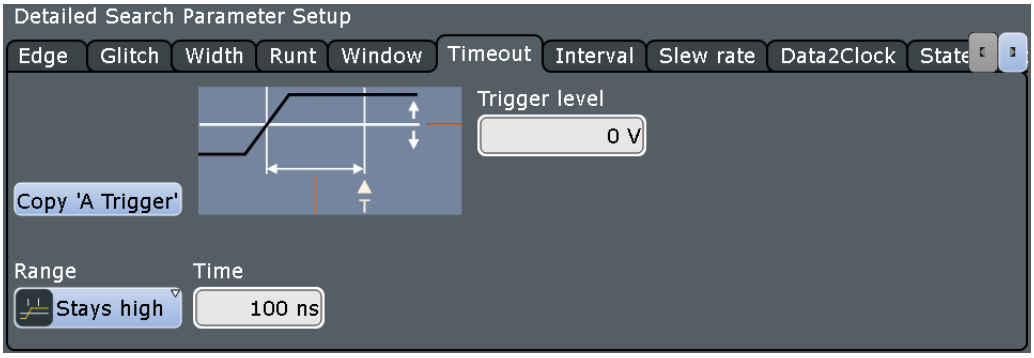
Remote command:
SEARch:TRIGger:WINDow:TIMerange on page 1331
SEARch:TRIGger:WINDow:WIDTh on page 1331
SEARch:TRIGger:WINDow:DELTa on page 1330

Upper level, Lower level
Set the upper and lower voltage thresholds. The amplitude of a runt crosses the first threshold twice in succession without crossing the second one.

Remote command:
SEARch:TRIGger:LEVel:WINDow:UPPer on page 1332
SEARch:TRIGger:LEVel:WINDow:LOWer on page 1332

10.2.2.6 Timeout

The timeout search settings are the same as the timeout trigger settings. To apply the trigger settings to search, tap Copy 'A Trigger'.



Range, Time
Set the timeout condition, see:
- "Range" on page 213
- "Time" on page 213

Remote command:
SEARch:TRIGger:TIMeout:RANGe on page 1327
SEARch:TRIGger:TIMeout:TIME on page 1327

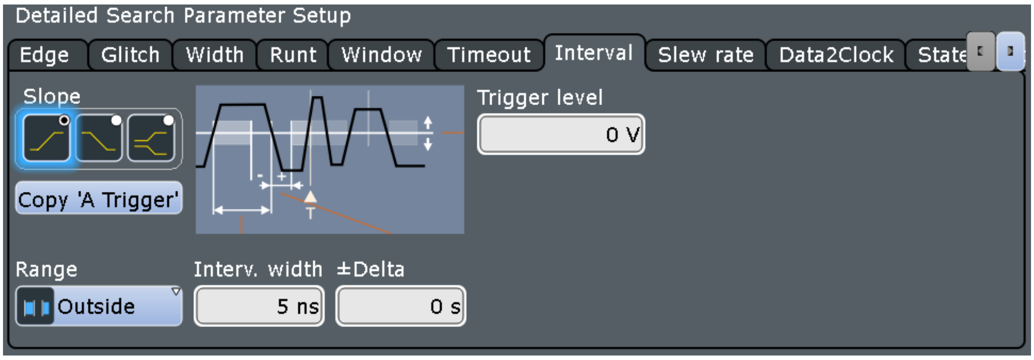
Trigger level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1318

10.2.2.7 Interval

The interval search settings are the same as the interval trigger settings. To apply the trigger settings to search, tap Copy 'A Trigger'.

The interval search is not available if the search source is a digital channel.



Slope, Range, Interv. width, ±Delta
Set the interval condition, see:
- "Slope" on page 213
  While the interval trigger can only analyze rising or falling edges, searching for a width is possible for both edges at the same time ("Either").
- "Range" on page 214
- "Interv. width" on page 214
- "±Delta" on page 214

Remote command:
`SEARch:TRIGger:INTerval:SLOPe` on page 1320
`SEARch:TRIGger:INTerval:RANGe` on page 1321
`SEARch:TRIGger:INTerval:WIDTh` on page 1322
`SEARch:TRIGger:INTerval:DELTa` on page 1321

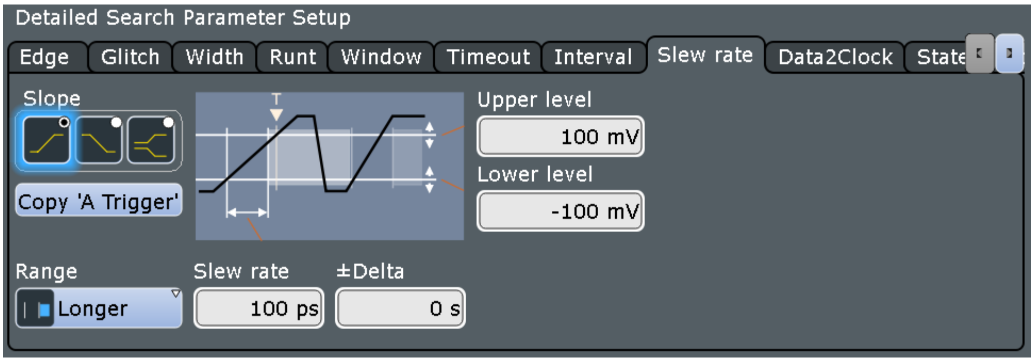
Trigger level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1318

10.2.2.8 Slew Rate

The slew rate search settings are the same as the slew rate trigger settings. To apply the trigger settings to search, tap Copy 'A Trigger'.

The slew rate search is not available if the search source is a digital channel.



Polarity, Range, Slew rate, ±Delta

- "Slope" on page 206
- "Range" on page 215
- "Slew rate" on page 215
- "±Delta" on page 215

Remote command:
`SEARch:TRIGger:SLEWrate:SLOPe` on page 1325
`SEARch:TRIGger:SLEWrate:RANGe` on page 1325
`SEARch:TRIGger:SLEWrate:TIME` on page 1326
`SEARch:TRIGger:SLEWrate:DELTa` on page 1324

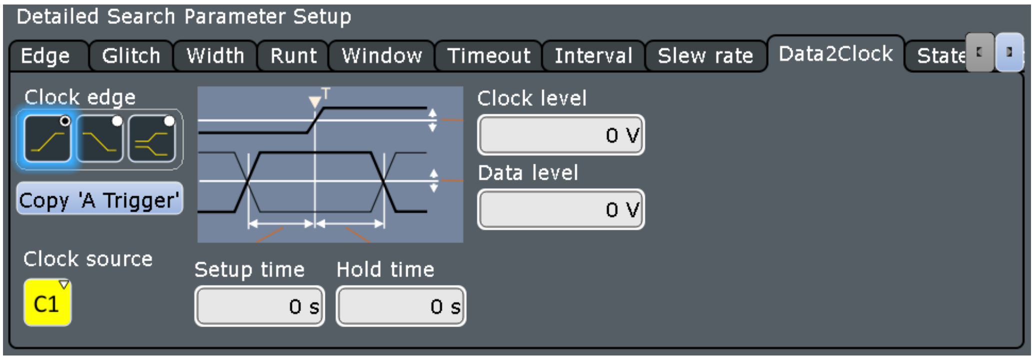
Upper level, Lower level

Set the upper and lower voltage thresholds. When the signal crosses a level, the slew rate measurement starts or stops depending on the selected slope.

Remote command:
`SEARch:TRIGger:LEVel:TRANsition:UPPer` on page 1326
`SEARch:TRIGger:LEVel:TRANsition:LOWer` on page 1326

10.2.2.9 Data2Clock

The Data2Clock search settings are the same as the Data2Clock trigger settings. To apply the trigger settings to search, tap Copy 'A Trigger'.



Clock source, Clock edge, Clock level

Set the clock settings. Both "Clock level" and "Clock edge" define the starting point for calculation of the setup and hold time.

Remote command:

`SEARch:TRIGger:DATatoclock:CSOurce` on page 1333

`SEARch:TRIGger:DATatoclock:CEDGe` on page 1332

`SEARch:TRIGger:DATatoclock:CLEVel` on page 1333

Data level

Sets the voltage level for the data signal. At this level, the setup and hold time are measured.

Remote command:

`SEARch:TRIGger:LEVel[:VALue]` on page 1318

Trigger level

Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:

`SEARch:TRIGger:LEVel[:VALue]` on page 1318

Setup time, Hold time

Sets the minimum time **before** (Setup) and **after** (Hold) the clock edge while the data signal must stay steady above or below the data level.

See also: "Setup time" on page 217 and "Hold time" on page 217.

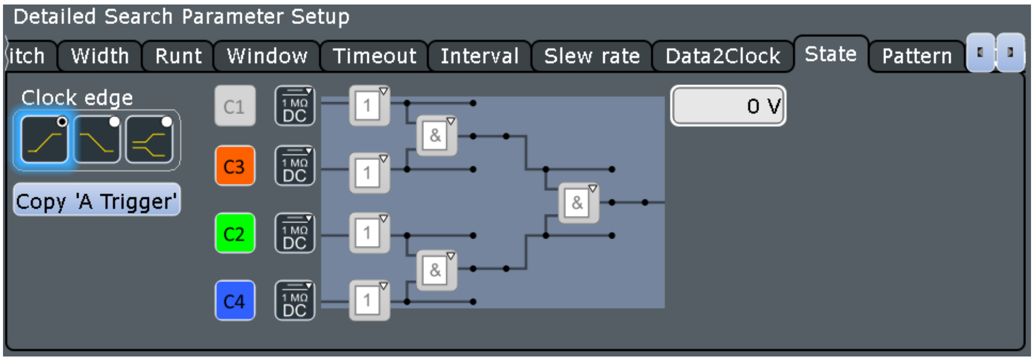
Remote command:

`SEARch:TRIGger:DATatoclock:STIMe` on page 1334

`SEARch:TRIGger:DATatoclock:HTIMe` on page 1333

10.2.2.10 State

The state search is a qualified edge search. The state search is only available for analog channel sources (Ch1 to Ch4).



Clock source, Clock edge

Define the clock settings. The clock signal is the waveform to be searched.

Remote command:

SEARch:TRIGger:STATe:CSOurce on page 1339

SEARch:TRIGger:STATe:CEDGe on page 1339

SEARch:TRIGger:STATe:CLEVel on page 1339

State pattern

State settings are the same as for the state trigger. See also "Pattern" on page 218.

Remote command:

SEARch:TRIGger:STATe:A[:ENABle] on page 1340

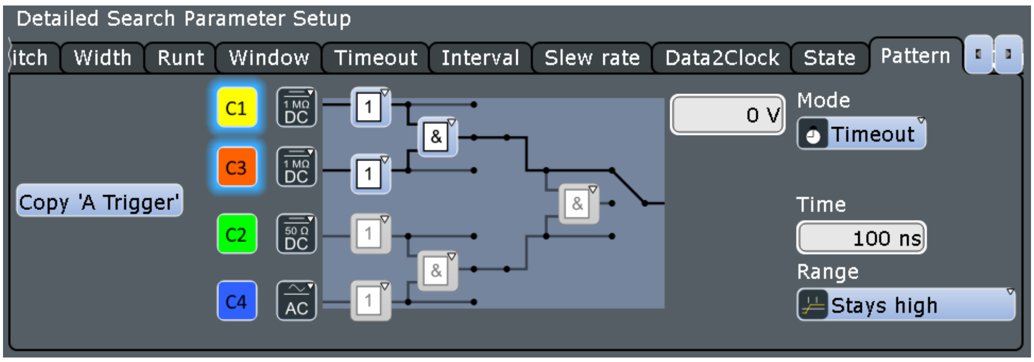
SEARch:TRIGger:STATe:A:LOGic on page 1340

SEARch:TRIGger:STATe:AB:LOGic on page 1341

SEARch:TRIGger:STATe:ABCD:LOGic on page 1341

10.2.2.11 Pattern

The pattern search combines a logical combination of the input channels with a timing condition. The pattern search is only available for analog channel sources (Ch1 to Ch4).



Pattern

Pattern search settings are the same as for the pattern trigger. See also "Pattern" on page 218.

Remote command:
`SEARch:TRIGger:PATTern:A[:ENABle]` on page 1335
`SEARch:TRIGger:PATTern:A:LOGic` on page 1335
`SEARch:TRIGger:PATTern:AB:LOGic` on page 1336
`SEARch:TRIGger:PATTern:ABCD:LOGic` on page 1336

Timing condition: Mode, Range, Time, Width, ±Delta
Additional time limitation to the pattern, see "Additional settings: Timing" on page 219.

Remote command:
`SEARch:TRIGger:PATTern:MODE` on page 1336
`SEARch:TRIGger:PATTern:TIMeout:MODE` on page 1337
`SEARch:TRIGger:PATTern:TIMeout[:TIME]` on page 1337
`SEARch:TRIGger:PATTern:WIDTh:RANGe` on page 1337
`SEARch:TRIGger:PATTern:WIDTh[:WIDTh]` on page 1338
`SEARch:TRIGger:PATTern:WIDTh:DELTa` on page 1338

10.2.3 Frequency Marker Search

When you start a search on a spectrum, a frequency marker search is performed to detect peaks in a spectrum. You can define which peaks the instrument finds by defining the noise reject settings.

Threshold
See "Threshold" on page 292.

Peak excursion
See "Peak excursion" on page 293.

10.2.4 Configuring the Search Setup

There are two ways to create a search:

- Creating a simple default search using the toolbar icon. This method is not available for search on serial buses.
- Setting up a search using the dialog box.

To perform a simple search

1. If more than one waveform is in the diagram, select the waveform to be searched for by tapping it in the diagram.
2. Select the "Search" icon on the toolbar.

3. Tap the diagram with the waveform to be searched, or drag a rectangle on the diagram to define the search area.

   The default edge search is configured as "Search<x>" and performed. The "Search Results" box is displayed.

To create a user-defined search

1. Press the [SEARCH] key.
2. There are two ways to create a search:
   - If you want to create a new, unconfigured search, tap the "Add" icon.
     

   - If you want to create a new search based on an existing one, tap the "Copy" icon.
     
3. Enter a name for the search using the on-screen keyboard.

To configure a user-defined search

1. Select the "Setup" tab and the search you want to configure.
2. Select the "Source" on which you want to perform the search.
3. Select the events you want to include in the search.
4. Define the settings of the first search event.
   To use the same conditions as defined in the trigger configuration of the A-event, tap "Copy 'A-Trigger". The selected trigger settings are applied to the search settings.
5. Repeat the previous steps to define further events for the same search.
6. To perform the search only on a part of the waveform, configure the gate in the "Gate" tab as described in Chapter 10.3.2, "Defining the Search Gate", on page 424.
7. To filter out noise from the search results, configure noise rejection as described in Chapter 10.5.2, "Defining Noise Rejection for Searches", on page 429.

   Note: A-event copy, gating and noise reject are not available for search on serial buses.

10.3 Search Gate

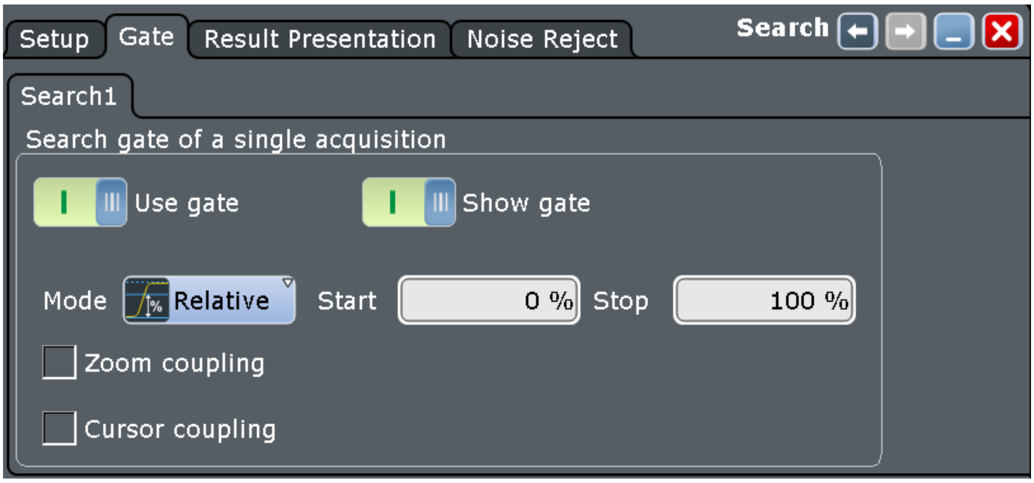
The gate defines the search area within the source waveform. You can use absolute or relative values to define the gate, or couple it to a previously defined zoom area.

10.3.1 Gate Settings

Access: [SEARCH] > "Gate" tab

The search gate settings are identical to those for gate areas for measurements or FFT analysis.

Gating is not available if the search source is a digital channel or a serial bus.



Use Gate
Enables the gate settings and shows the gate. Search is only performed on the defined gate area of the source waveform.

Remote command:
`SEARch:GATE[:STATe]` on page 1342

Show gate
Displays the gate area in the source diagram.

Remote command:
`SEARch:GATE:SHOW` on page 1343

Mode
Defines whether the gate settings are configured using absolute or relative values.

"Absolute"	The gate is defined by absolute start and stop values.
"Relative"	The gate's start and stop values are defined by a percentage of the value range.

Remote command:
CALCulate:MATH<m>:FFT:GATE:MODE on page 1286
MEASurement<m>:GATE:MODE on page 1266
SEARch:GATE:MODE on page 1342

(Relative) Start
Defines the starting value for the gate.

Remote command:
CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt on page 1286
CALCulate:MATH<m>:FFT:GATE:RELative:STARt on page 1286
MEASurement<m>:GATE:ABSolute:STARt on page 1266
MEASurement<m>:GATE:RELative:STARt on page 1266
SEARch:GATE:ABSolute:STARt on page 1343
SEARch:GATE:RELative:STARt on page 1343

(Relative) Stop
Defines the end value for the gate.

Remote command:
CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP on page 1286
CALCulate:MATH<m>:FFT:GATE:RELative:STOP on page 1287
MEASurement<m>:GATE:ABSolute:STOP on page 1266
MEASurement<m>:GATE:RELative:STOP on page 1266
SEARch:GATE:ABSolute:STOP on page 1343
SEARch:GATE:RELative:STOP on page 1344

Zoom coupling
Zoom coupling is available if a zoom is defined. As long as "Zoom coupling" is enabled, the gate area is defined identically to the zoom area - if you change the zoom, the gate changes as well.

If several zoom diagrams are defined, select the zoom diagram to be used for gating. The "Start" and "Stop" values of the gate are adjusted accordingly.

Zoom coupling can be set for measurement gates, FFT gates, and search gates. The zoom must be defined on the diagram that contains the signal source of the measurement, FFT, or search.

Remote command:
MEASurement<m>:GATE:ZCOupling on page 1267
MEASurement<m>:GATE:ZDIagram on page 1268
CALCulate:MATH<m>:FFT:GATE:ZCOupling on page 1287
SEARch:GATE:ZCOupling on page 1344
SEARch:GATE:ZDIagram on page 1344

Cursor coupling
If enabled, the gate area is defined by the cursor lines of an active cursor measurement. If several cursor measurements are enabled, select the cursor set to be used for gating. The "Start" and "Stop" values of the gate are adjusted to the values of the cursor line positions. The measurement is limited to the part of the waveform between the cursor lines.

Remote command:
MEASurement<m>:GATE:CCOupling on page 1267
MEASurement<m>:GATE:CURSor on page 1267

10.3.2 Defining the Search Gate

If you create a search using the "Search" toolbar icon, you can directly define the gate by dragging a rectangle on the diagram. Otherwise, you define the gate in the "Gate" tab of the "Search" dialog box.

1. Press the [SEARCH] key and select the "Gate" tab.
2. Select the search for which you want to define the gate.
3. Use one of the following methods:
   - Set the absolute or relative "Mode" and enter the start and stop values of the gate area.
   - If a zoom area has already been defined for the waveform, couple the gate area to the zoom area by selecting the "Zoom coupling" option. If several zoom diagrams are defined, select the zoom diagram you want to use for gating.
4. Tap "Use gate" to enable the gate.
5. Optionally, tap "Show gate" to display the gate area in the diagram.

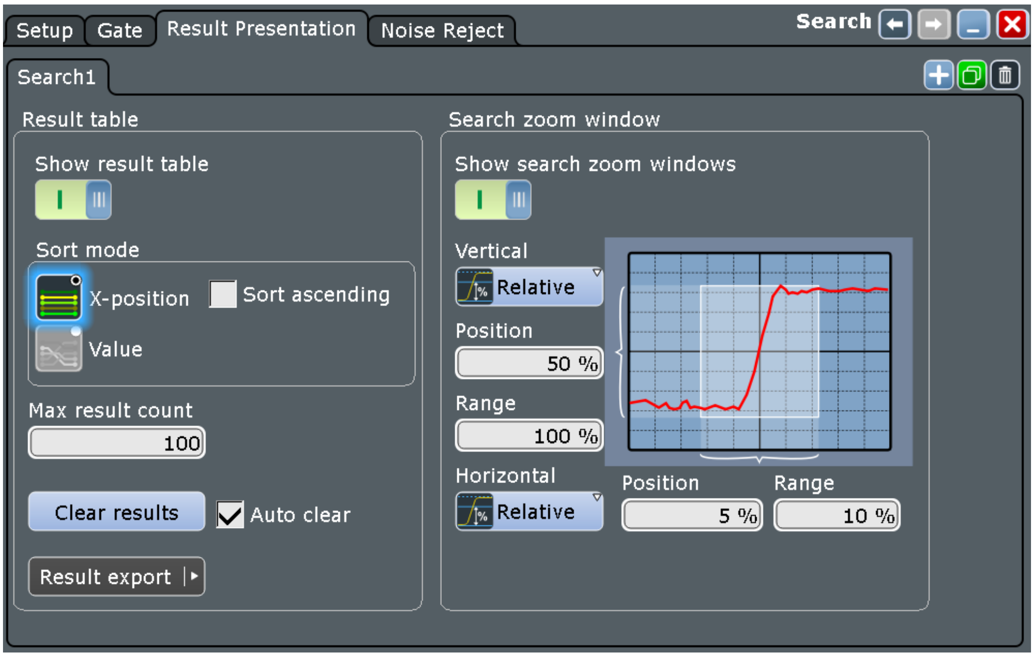
10.4 Result Presentation

Search results are displayed in a table in the "Search Results" box. In addition, a zoom window for a selected search result can be displayed so that you can analyze the result in more detail.

10.4.1 Result Presentation Settings

Access: [SEARCH] > "Result Presentation" tab

The following settings configure the layout of the result table in the "Search Results" box and the size and position of the search zoom window. The result tables can be sorted by x-position or value. You can define a maximum number of table entries.



Result table

These settings refer to the search result table.

Show result table ← Result table

Displays or hides the search result table.

Remote command:

`SEARch:RESult:SHOW` on page 1351

Sort mode ← Result table

Sorts the search results by x-value position or value of the result.

Remote command:

`SEARch:RESult:SORT[:MODE]` on page 1351

Sort ascending ← Result table

By default, the results are listed in descending order, i.e. the largest value at the top. To change the sorting direction, enable "Sort ascending".

Remote command:

`SEARch:RESult:SORT:ASCending` on page 1351

Max result count

Defines the maximum number of entries in the search result table.

Remote command:

`SEARch:RESult:LIMit` on page 1350

Auto clear

If "Auto clear" is enabled, the instrument displays the search results of the last acquisition.

If "Auto clear" is disabled, the first result of each acquisition is listed until the maximum number of entries in the table is reached.

Clear results
Clears the search results once and starts a new search.

Remote command:
`SEARch:CLEar` on page 1315

Search zoom window
The search zoom window allows you to analyze the search results in more detail.

You can change the size and the position of the search zoom area in the same way as a usual zoom. If you move the zoom area in the source diagram, the nearest search result is marked in the results table. See also: Chapter 6.1.1, "Methods of Zooming", on page 240

The search zoom area is marked in the waveform diagram. You can change the color of the area with: "Display" menu > "Diagram layout" > "Search result gate symbol color" on page 112.

Show search zoom windows ← Search zoom window
If enabled, a zoom window is displayed for the currently selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.

Remote command:
`SEARch:RESDiagram:SHOW` on page 1349

Vertical ← Search zoom window
Defines whether absolute or relative values are used to specify the y-axis values.

Remote command:
`LAYout:ZOOM:VERTical:MODE` on page 1179
`SEARch:RESDiagram:VERT:MODE` on page 1349

Position / Relative position (vertical) ← Search zoom window
Defines the y-value of the centerpoint of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:ABSolute:POSition` on page 1179
`LAYout:ZOOM:VERTical:RELative:POSition` on page 1181
`SEARch:RESDiagram:VERT:ABSolute:POSition` on page 1349
`SEARch:RESDiagram:VERT:RELative:POSition` on page 1350

Range / Relative Range (vertical) ← Search zoom window
Defines the height of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:RELative:SPAN` on page 1181
`LAYout:ZOOM:VERTical:ABSolute:SPAN` on page 1180
`SEARch:RESDiagram:VERT:ABSolute:SPAN` on page 1349
`SEARch:RESDiagram:VERT:RELative:SPAN` on page 1350

Horizontal ← Search zoom window
Defines whether absolute or relative values are used to specify the x-axis values.

Remote command:
`LAYout:ZOOM:HORZ:MODE` on page 1176
`SEARch:RESDiagram:HORZ:MODE` on page 1348

Position / Relative position (horizontal) ← Search zoom window
Defines the x-value of the centerpoint of the zoom area.

Remote command:
`LAYout:ZOOM:HORZ:ABSolute:POSition` on page 1176
`LAYout:ZOOM:HORZ:RELative:POSition` on page 1178
`SEARch:RESDiagram:HORZ:ABSolute:POSition` on page 1347
`SEARch:RESDiagram:HORZ:RELative:POSition` on page 1348

Range / Relative Range (horizontal) ← Search zoom window
Defines the width of the zoom area.

Remote command:
`LAYout:ZOOM:HORZ:ABSolute:SPAN` on page 1176
`LAYout:ZOOM:HORZ:RELative:SPAN` on page 1178
`SEARch:RESDiagram:HORZ:ABSolute:SPAN` on page 1347
`SEARch:RESDiagram:HORZ:RELative:SPAN` on page 1348

10.4.2 Configuring the Search Results Presentation

Initially, the "Search Results" box is displayed in front of the other diagrams or as result icon on the signal bar. This depends on the default setting in the "Diagram Layout" tab. Alternatively, you can display it in its own area on the screen, like any other diagram.

For details, see Chapter 2.4.8, "Displaying Results", on page 85.

To configure the result tables

1. Press the [SEARCH] key to open the "Search" dialog box.
2. Select the tab for the search you want to configure.
3. Select the "Result Presentation" tab.
4. Select "Show result table" to display the "Search Results" box.
5. Select the sort mode of the result table.
6. By default, the results are listed in descending order, i.e. the largest value at the top. To change the sorting direction, enable "Sort ascending".
7. Define a maximum number of results to be displayed in the result table in the "Max result count" field.

To display search zoom windows

1. In the "Search Results" box, select "Show search zoom windows".

This stops a running search and a running acquisition.
The zoom area is indicated in the diagram that displays the source waveform of the search. The zoom window is displayed for the first result that was found.

2. If you need to adjust the search zoom area, you can drag the area or their edges on the screen. You can also enter the limits of the search zoom window in the "Search > Results Presentation" tab.
Be aware, that the zoom window size is valid for all results of a search definition. If you change the settings drastically for one result, they may not be correct for the next search result you switch to.
See also:
   - Chapter 6.1.3, "Zooming for Details", on page 246
   - "Navigating search results" on page 408

10.5 Noise Reject

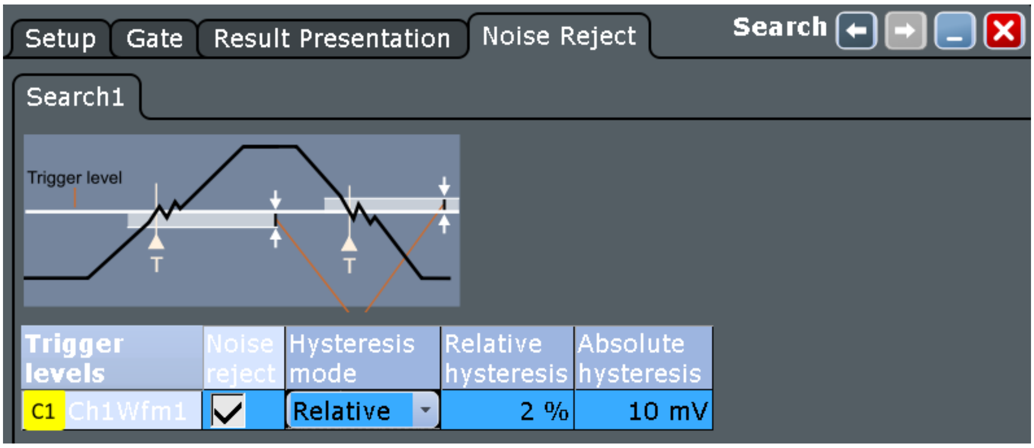
Noise rejection for searches is similar to noise rejection for triggers. You can reject noise by setting a hysteresis to avoid finding events caused by noise oscillation around the trigger level.

10.5.1 Noise Reject Settings

Access: [SEARCH] > "Noise Reject" tab

You can select the hysteresis mode and value for each analog and digital input channel, math and reference waveform.

The noise reject settings are similar to those for triggers, see also Chapter 5.5, "Noise Reject", on page 227.



Noise reject
If enabled, the hysteresis is considered for the search.

Remote command:
SEARch:TRIGger:LEVel:NOISe[:STATe] on page 1346

Hysteresis mode
Defines whether values absolute or relative to the vertical scaling are used.

Remote command:
SEARch:TRIGger:LEVel:NOISe:MODE on page 1345

Relative / Absolute hysteresis
Defines a range in absolute or relative values around the search level. If the signal jitters inside this range and crosses the level, no search event is detected.

Absolute hysteresis values are adapted when the relative hysteresis is changed , and vice versa.

If you change the vertical scaling, either the relative or the absolute value is adjusted automatically.

Remote command:
SEARch:TRIGger:LEVel:NOISe:ABSolute on page 1345
SEARch:TRIGger:LEVel:NOISe:RELative on page 1346

10.5.2 Defining Noise Rejection for Searches

1. Press the [SEARCH] key to open the "Search" dialog box.
2. Select the "Noise reject" tab.
3. Select the tab for the search you want to configure.
4. Define the absolute or relative hysteresis. If you change one value, the other is automatically calculated.

11 Data and File Management

This chapter describes how to manage instrument settings, and measurement results like waveform data, numeric results and screenshots.

The [FILE] key provides functions for saving and restoring data on the instrument. A naming pattern is available and can be adjusted to simplify a clear data storage.

The effect of the [PRINT] key can be configured to save or print screenshots or reports.

- Instrument Settings....................430
- Waveform Data and Results....................438
- Autonaming....................459
- Screenshots....................461
- Reports....................467
- Preset Setup....................469
- File Selection Dialog....................471

11.1 Instrument Settings

To repeat measurements at different times or perform similar measurements with different test data, it is useful to save the used instrument settings and load them again later. Furthermore, it can be helpful to refer to the instrument settings of a particular measurement when analyzing the results. Therefore, you can easily save the instrument settings of a measurement. In addition to the measurement-related settings, user-specific display settings and active reference waveforms can also be saved and loaded.

Access to save instrument settings: [FILE] key > "Save" tab > "User settings"

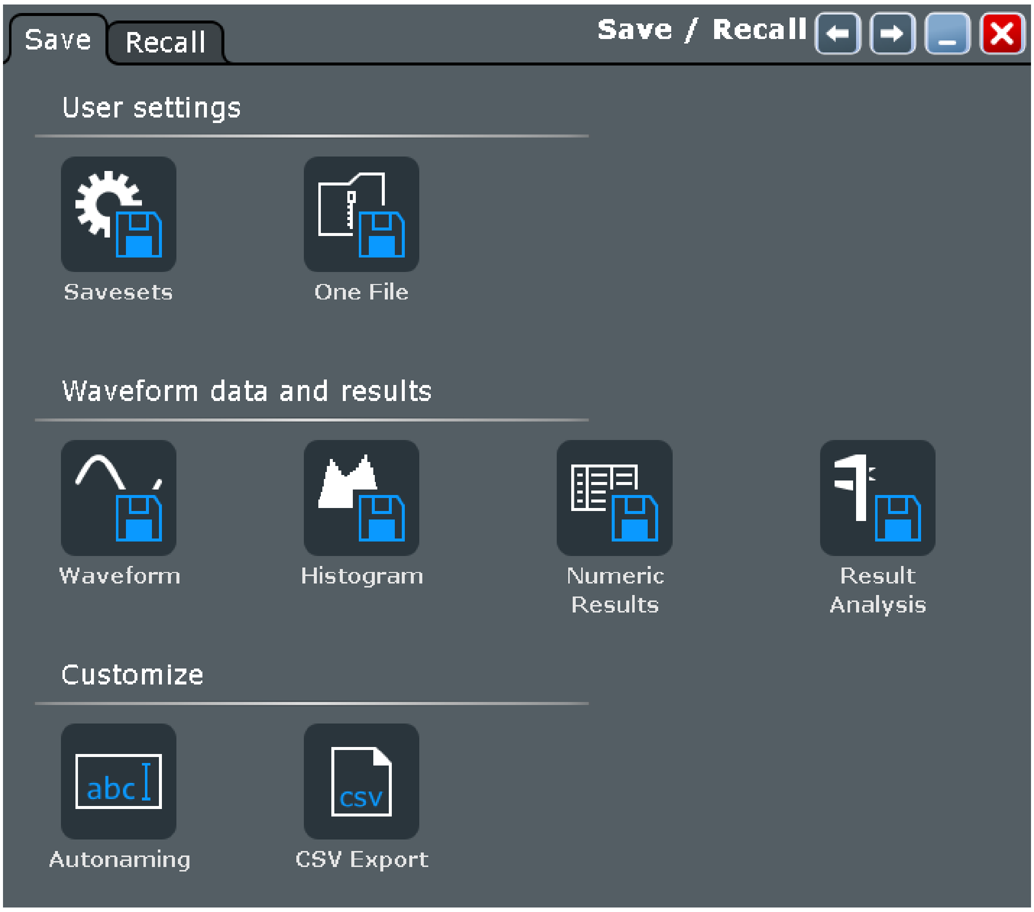


Figure 11-1: Save tab

Access to load instrument settings: [FILE] key > "Recall" tab > "User settings"

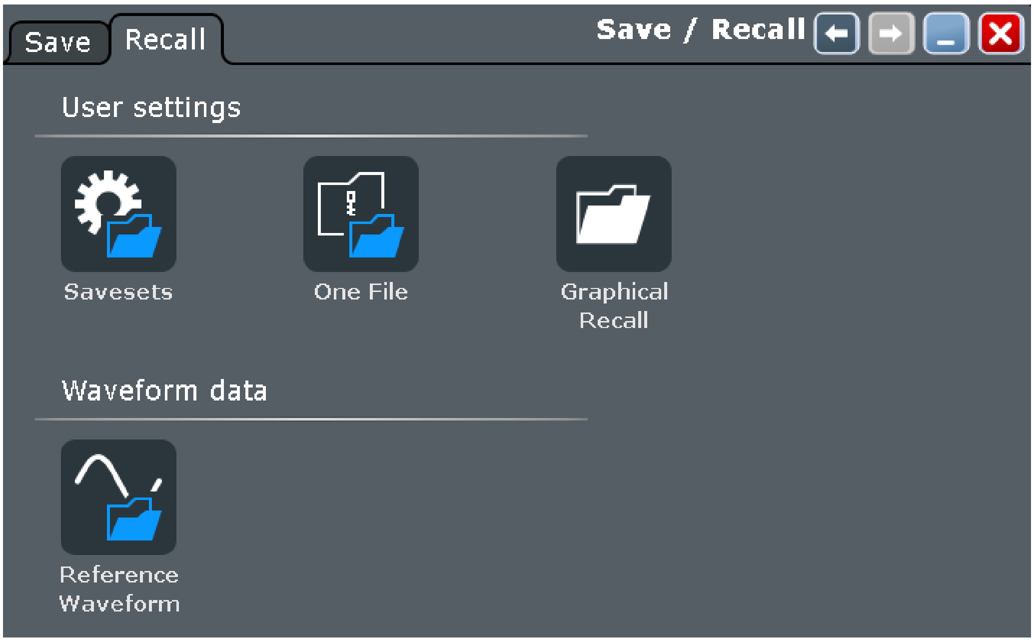


Figure 11-2: Recall tab

The R&S RTE provides three types of saving and restoring settings:

- **Savesets** contain the complete instrument and measurement configuration except for user-specific display settings.
- **One File** contains the saveset and active reference waveform files in a ZIP file.
- **User-defined presets** contain the complete instrument setup including display settings, except for transparency and intensity. These settings can be restored by pressing the [PRESET] key. See Chapter 11.6, "Preset Setup", on page 469.

Access: [FILE] key

- Savesets..432
- One File..434
- Graphical Recall Function..435
- Saving and Loading Settings..437

11.1.1 Savesets

Savesets contain the complete instrument and measurement configuration including a screenshot of the current display, but except for user-specific display settings stored as user preferences. You can save an unlimited number of setting files.

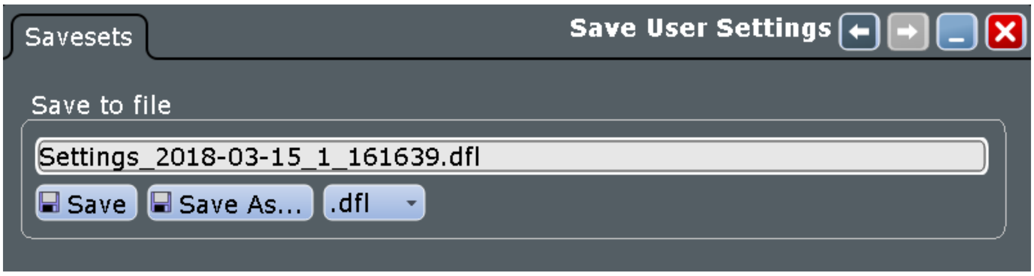
The waveform generator/pattern generator and the pulse source state is saved as "Off".

To load a saveset quickly, you can use the "Load saveset" function on the toolbar. A graphical preview helps you to find the required settings file.

If you need to store the instrument settings often, you can add the "Save settings" icon to the toolbar and use the icon to store the saveset file.

11.1.1.1 Save Saveset Settings

Access: [FILE] key > "Save" tab > "Savesets"



Save to file

Enter the file name to save the setting data to, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box. See also: Chapter 11.7, "File Selection Dialog", on page 471.

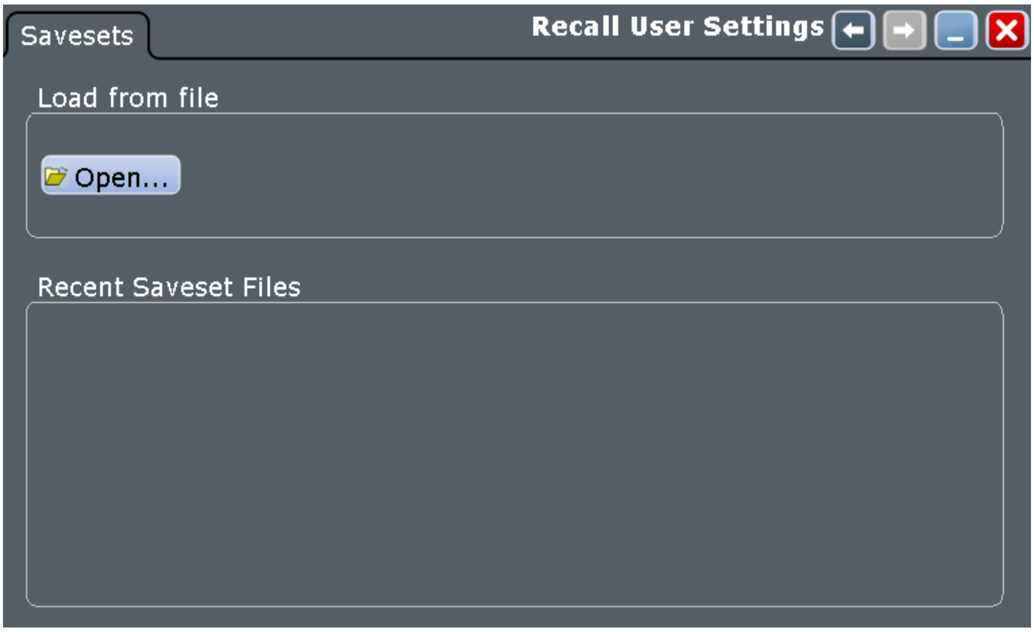
By default, saveset file names have the prefix "Settings_".

"Save"	Saves the data to the selected file.
"Save As..."	Opens the file selection dialog box and saves the data to the selected file.
".dfl/.xml"	Selects the file format.

Remote command:
`MMEMory:SAV` on page 1358

11.1.1.2 Load Saveset Settings

Access: [FILE] key > "Recall" tab > "Savesets"



Load from file
Enter the file name to load the setting data from, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box.
See also: Chapter 11.7, "File Selection Dialog", on page 471.

By default, saveset file names have the prefix "Settings_".

"Open" Opens a file selection dialog box and loads the selected file.

Remote command:
MMEMory:RCL on page 1359

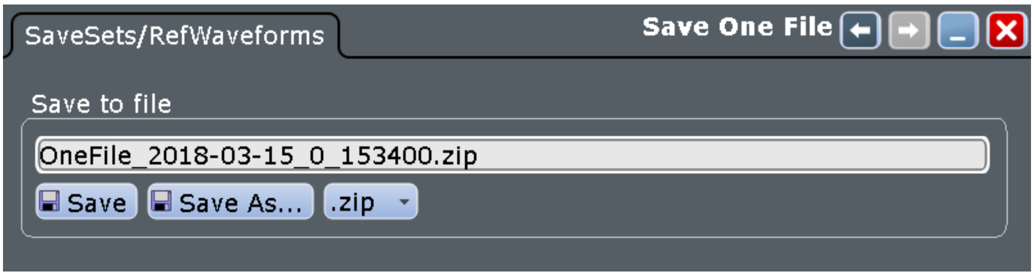
11.1.2 One File

The One File function stores the savesets file and the data of active reference waveforms together in a ZIP file. If you want to keep channel waveform data together with the instrument settings, make sure to create a reference waveform from the channel waveform before you save the One File.

The waveform generator/pattern generator and the pulse source state is saved as "Off".

11.1.2.1 Save One File Settings

Access: [FILE] key > "Save" tab > "One File"



Save to file
Enter the file name to save the One File data to, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box.
See also: Chapter 11.7, "File Selection Dialog", on page 471.

By default, saveset file names have the prefix "OneFile_".

"Save" Saves the data to the selected file.

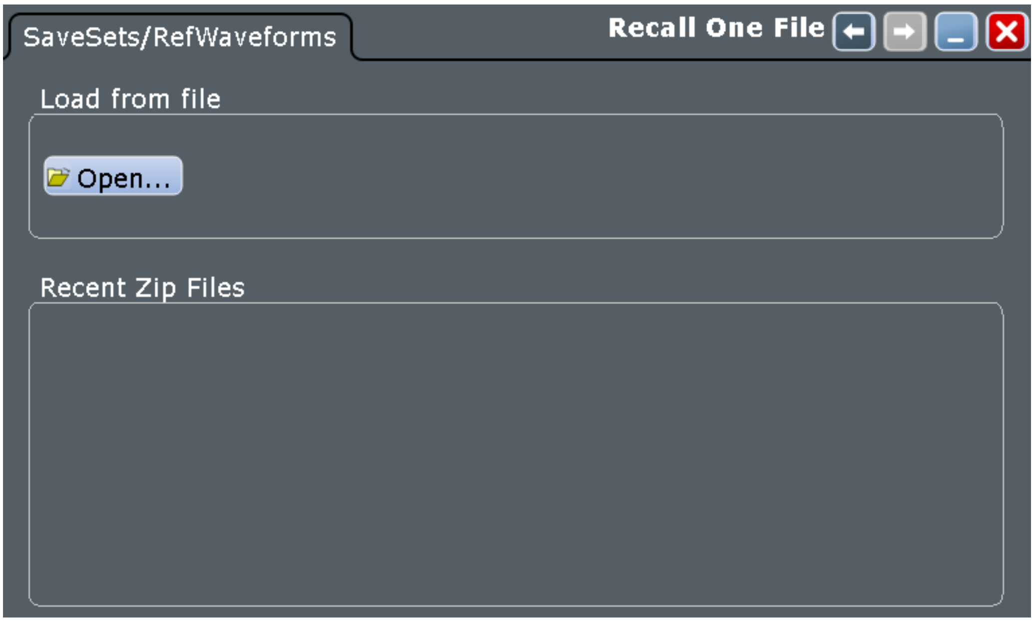
"Save As..." Opens the file selection dialog box and saves the data to the selected file.

".zip" Shows the file format.

Remote command:
SAVeset:ONEFile:NAME on page 1360
SAVeset:ONEFile:SAVE on page 1361

11.1.2.2 Load One File Settings

Access: [FILE] key > "Recall" tab > "One File"



Load from file
Select "Open" to open the file selection dialog box, and to select the required file. See also: Chapter 11.7, "File Selection Dialog", on page 471.

By default, saveset file names have the prefix "OneFile_".

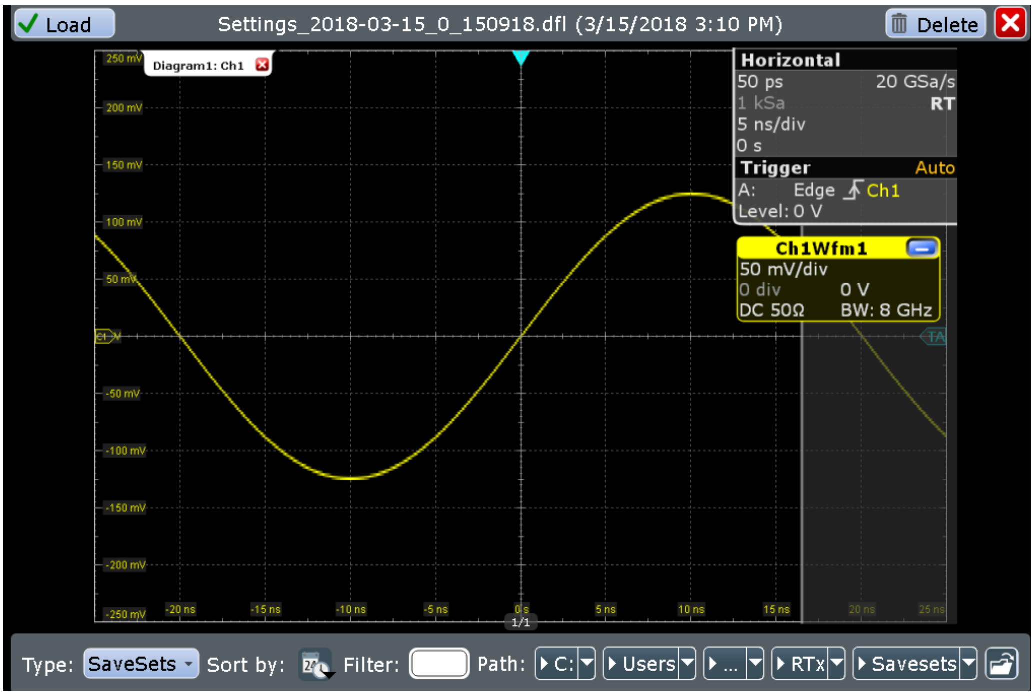
Remote command:
`SAVeset:ONEFile:NAME` on page 1360
`SAVeset:ONEFile:OPEN` on page 1360

11.1.3 Graphical Recall Function

Access: [FILE] key > "Recall" tab > "Graphical Recall"

The graphical preview helps you to find the required settings file or a reference wave-form.

The "Graphical Recall" function is also available on the toolbar, see Chapter 2.4.7.3, "Toolbar Functions", on page 80.



1. On the toolbar, tap the "Graphical Recall " icon.

   A window opens and shows the screenshot of the first measurement configuration that is stored in the default directory.

2. Select if you want to display a "SaveSets" file or a reference waveform ("Ref Wfm").

3. Find the required file using one of the following methods:

   - Tap the "Next" icon on the right or the "Previous" icon on the left to scroll the savesets of the directory.

     The file name is shown on the top, and the screenshot helps to identify the settings.
   - If the saveset was stored in another directory, use the path buttons at the bottom, or tap "Open" to open the required directory.
   - Tap "Sort by" to sort the files according to the "Last modified" date or the "Filename".
   - Specify a "Filter" name to show only the files containing the filter string.

4. Tap "Load" in the upper left corner to recall the settings of the selected file.

11.1.4 Saving and Loading Settings

Settings can be stored in a file with user-defined name and location, or in a quick saveset. The settings in a saveset can be saved and retrieved quickly at the touch of a button, so savesets are ideal for frequently used measurements.

For details on save/recall instrument settings and associated remote commands, see Chapter 11.1.1, "Savesets", on page 432.

To save settings to a saveset file

Alternatively, you can add the "Save settings" icon to the toolbar and use the icon to store the saveset file to the folder and file specified in the "Settings" tab. See also Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.

1. Press the [FILE] key.
2. In the "Save" tab, press the "Saveset" button .
3. Tap "Save" to save the settings to the specified file.
   Tap "Save As" to save the settings to a different file. Select the file and directory from the file selection dialog box.

   The current settings are saved to the selected file.

To load settings from a saveset file

Alternatively, you can use the "Load saveset" function on the toolbar, see Chapter 11.1.3, "Graphical Recall Function", on page 435.

1. Press the [FILE] key.
2. Select the "Recall" tab.
3. Press the "Saveset" button.
4. Tap "Load" to load the settings from the specified file.
   Tap "Open" to navigate to a different file. Select the file from the file selection dialog box and tap "Select".

   The saved settings are loaded to the R&S RTE.

To save a One File

1. If you want to save reference waveforms in the One File, create and display the reference waveforms.
2. Press the [FILE] key.
3. In the "Save" tab, tap the "One File".
4. Tap "Save" to save the settings to the specified file.
   Tap "Save As" to save the settings to a different file. Select the file and directory from the file selection dialog box.

   The current settings and the active reference waveforms are saved to the selected file.

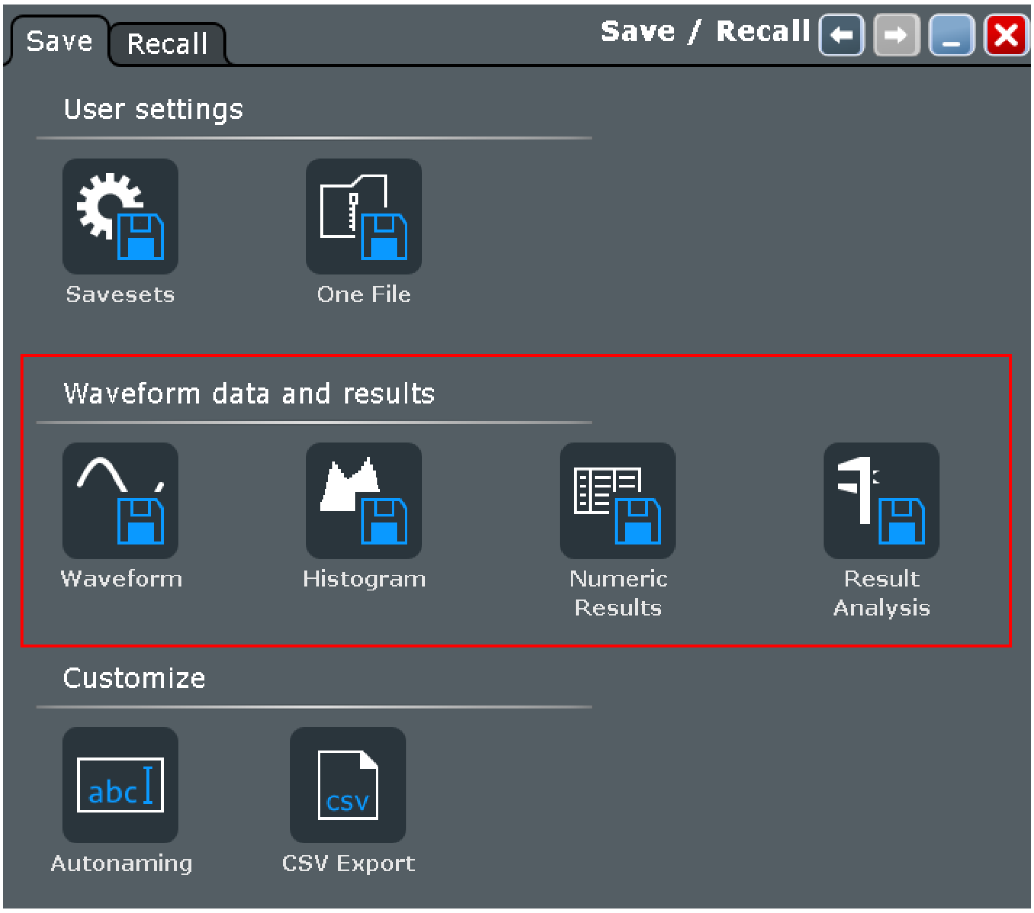
To load a One File

1. Press the [FILE] key.
2. Select the "Recall" tab.
3. Press the "One File" button.
4. Tap "Load" to load the settings from the specified file.
   Tap "Open" to navigate to a different file. Select the file from the file selection dialog box and tap "Select".

   The saved settings and reference waveforms are loaded to the R&S RTE.

11.2 Waveform Data and Results

Access: [FILE] > "Save" tab



You can export various data to file: waveform data, histograms, and measurement results.

- Waveform Export Files....................439
- Waveforms - Export Settings....................445
- Waveform Histogram....................450
- Numeric Results....................452
- Result Analysis....................454
- CSV Export....................456
- Saving and Loading Waveform Data....................457

11.2.1 Waveform Export Files

Waveforms can be stored in XML, CSV, or BIN format.

Reloading waveforms: Restrictions

To reload waveform data as a reference waveform, it must be stored in BIN format.

If multiple acquisitions of one waveform are exported (Data logging or Multiple waveforms), only the first acquisition can be reloaded.

If the signal is a spectrum, reloading is only possible for waveforms with "Magnitude unit" = Linear. Waveforms with logarithmic unit cannot be reloaded.

Data of all waveforms is saved in two files. One file contains the waveform data values and is indicated by `*Wfm.*` in the filename. The second file contains the header data, for example, time scale, vertical scale, vertical and horizontal positions, interpolation mode and much more. Header data is required to restore the waveform from data, or to analyze the data values of the data file.

11.2.1.1 Header Files

The header files of XML and BIN waveform files are written in XML format. The header files of CSV waveform files are written in CSV format. You can open the header files and use their information for data analysis.

CSV header files only contain the property names and values, one property per row.

```
Resolution:1e-010:
RecordLength:1000:
```

XML header files contain more information than CSV header files. The additional information is required to reload the stored waveforms with their correct settings.

```
<Prop Avail="0" ValueKey="" Name="Resolution" Value="1e-010" UserValue="0"
Step="1e-011" Default="0" Min="0" Max="1e+026" StepDefault="1e-011"
StepFactor="10" Resolution="0" UnitId="55" UnitName="s" UnitPowerProduct=""
BitGroupSize="0" Format="0"></Prop>

<Prop Avail="0" ValueKey="" Name="RecordLength" Value="1000" UserValue="1000"
Step="1" Default="1000" Min="0" Max="4294967295" StepDefault="1" StepFactor="10"
Resolution="1" UnitId="93" UnitName="Sa" UnitPowerProduct="" BitGroupSize="0"
Format="0"></Prop>
```

Header files contain the following properties:

Table 11-1: Header file properties

Value	Description
General	
FirmwareVersion	Firmware version that is installed on the R&S RTE (last entry in the header file)
Source	Name of the exported waveform
Resolution	Time between two samples *Resolution = 1 / Sample Rate*
SignalResolution	Time between two samples in this waveform. The value can differ from Resolution if the source is, for example, a spectrum, a bus signal, a correlation or a measurement. The value is determined automatically considering the waveform parameters and their dependencies. If the signal is a spectrum, the value indicates the frequency range of FFT bins.
EnhancementMode	Method to increase the sample rate if the required sample rate is higher than the ADC sample rate.
InterpolationMode	Interpolation method. The value is relevant when the enhancement mode is interpolated time.
DecimationMode	Method to reduce the number of data samples to achieve the required sample rate
DecimationFactor	Factor to the number of data samples to achieve the required sample rate *Decimation factor = ADC sample rate / Sample rate*
TraceArithmetics	Off, Envelope, or Average
InterleavedTraceCount	Number of y-values saved at each sampling time. The value is usually 1. The value is 2, if min and max values are saved for each sample, for example, for envelope waveforms.
SignalFormat	Format of the data values: ● FLOAT: floating point numbers, general export format ● INT (8 Bit): Integer 8 bit, used for "Raw (ADC sample)" data export. ● INT (16 Bit): Integer 16 bit, used for "Raw (ADC sample)" data export in high definition mode.
Timestamp	Absolute time of the waveform recording
ByteOrder	Endianness, only relevant for raw data export in high definition mode (SignalFormat = INT (16 Bit)). ● LSB first: little endian, least significant byte first ● MSB first: big endian, most significant byte first
NumericFormat	Number format of bus values and digital channel data (bit pattern format)
Record length	
RecordLength	Number of samples in a waveform record of one acquisition
HWRecordLength	Equivalent to the RecordLength

Value	Description
SignalRecordLength	Number of required samples in the waveform. The value can differ from RecordLength and HWRecordLength if the source is, for example, a spectrum, a bus signal, a correlation or a measurement. The value is determined automatically considering the waveform parameters and their dependencies. If the signal is a spectrum, the value indicates the number of FFT bins.
SignalHardwareRecordLength	Number of samples actually available in this waveform, including the number of required samples in the waveform and the additional samples needed for further computation
LeadingSettlingSamples	Relevant only for BIN files. In XML and CSV files, the value is 0. Number of additional samples before the beginning of waveform samples. These additional samples are needed for further computation, for example, for filters.
Horizontal system	
TimeScale	Horizontal scale in seconds per division
HorizontalDivisionCount	Number of horizontal divisions
RescaleCenterTime	Horizontal position, the time distance between the reference point and the zero point of the diagram
RescaleCenterPoint	Position of the reference point in % of the screen
ReferencePoint	Position of the zero point in % of the screen
TriggerOffset	Time distance from the trigger point to the zero point of the diagram
XStart	Horizontal start value of the waveform (time or frequency) *)
XStop	Horizontal stop value of the waveform (time or frequency)
HardwareXStart	Actual horizontal start value of data, including the settling time for further computation *)
HardwareXStop	Actual horizontal stop value of data, including the settling time for further computation
	*) If the waveform is a spectrum, the XStart and HardwareXStart values may be slightly smaller than the specified start frequency, or even get negative. The spectrum is centered on the center frequency, and the frequency range covered by one spectral bin is given by the SignalResolution. Hence, the spectral bin in the center of the spectrum always covers the range [CenterFrequency; CenterFrequency + SignalResolution[. As a result, the range covered by the first spectral bin in the spectrum may reach further than the specified start frequency. It is ensured that the specified start frequency is included in the frequency range.
Vertical system In case of multi channel export, the values of channel1 are delivered, no matter if channel 1 is exported or not.	
VerticalScale	Vertical scale of the waveform in Volts per division, or other unit / division
VerticalDivisionCount	Number of vertical divisions
VerticalPosition	Vertical position of the waveform in divisions
VerticalOffset	Vertical offset of the waveform in Volts, or other unit

Value	Description
NofQuantisationLevels	Theoretical number of quantization levels in the signal. This value depends on the waveform format (8 bit, 16 bit, …). In case of a math waveform, it depends on the quantization levels of the operands and on the operator type.
BaseYStart	Vertical start value of the waveform
BaseYStop	Vertical stop value of the waveform
Multi channel export The header files contain strings like this: `MultiChannelVerticalOffset: 4:1.63:1.96:0:0:1e-005:0:-1e+026:1e+026:1e-005:10:0:V:`. Only the first 5 values and the unit at the end of the string are relevant for data analysis. All other values are for internal use and not explained here. Examples are in csv format.	
MultiChannelExport	Indication whether multiple channels are exported simultaneously: On | Off
MultiChannelExportState	Number of channels and export status of the individual channels, for example, 4:On:Off:On:On...: channels 1, 3 and 4 are exported.
MultiChannelVerticalOffset	Number of channels and vertical offset of the individual channels, for example, 4:0:0:0.02:0...: channel 3 has an offset of 20 mV.
MultiChannelVerticalPosition	Number of channels and vertical position of the individual channels, for example, 4:0:0:0:2...: the position of channel 4 is 2 divisions.
MultiChannelVerticalScale	Number of channels and vertical scale of the individual channels, for example, 4:0.05:0:0.03:0.04...: scale of channel is 50 mV/div, channel 3 has 30 mV/div and channel 4 has 40 mV/div.
MultiChannelBaseYStart	Number of channels and minimum value of the vertical range for each individual channel, for example, 4:-0.25:0:-0.13:-0.28...
MultiChannelBaseYStop	Number of channels and maximum value of the vertical range for each individual channel, for example, 4:0.25:0:0.17:0.12...: The range of channel 1 is -250 mV to 250 mV. The range of channel 3 is -130 mV to 170 mV. The range of channel 4 is -280 mV to 120 mV.
History	
TimestampState	State of the timestamps export. If on, the timestamps of each history waveform are written to the waveform data file.
Math waveform	
BaseUnit	Base unit of a mathematic waveform, for example, linear unit
ViewUnit	User-selected unit of a mathematic waveform, for example, logarithmic unit for a spectrum. The value is only valid if the exported waveform is a math waveform.
ViewUnitRelative	Indication of a relative unit. It is true if the math waveform has the ViewUnit "dB", for example. The value is only valid if the exported waveform is a math waveform.
ViewReferenceLevel	Reference level for a relative unit. The value is only valid if the exported waveform is a math waveform, and the unit is relative.
FFT	
CenterFreq	Center frequency of the spectrum

Value	Description
FreqSpan	Frequency span of the spectrum
FrequencyStart	Start frequency of the spectrum
FrequencyStop	Stop frequency of the spectrum
WindowType	Window used for the spectrum computation
ResolutionBW	Resolution bandwidth of the spectrum
AdjustedResolutionBW	Actual resolution bandwidth of a spectrum waveform. The value is only valid if the exported waveform is a spectrum.
GateRBWCoupling	Indication whether the record length or the resolution bandwidth is a constant for the spectrum computation
Parameters for power calculation	
Impedance	Impedance used for power calculation
NoiseBandwidth	Noise bandwidth of a spectrum waveform, required for power calculation. The value is only valid if the exported waveform is a spectrum.
Parameters for internal use	
SourceType	Source qualifier
TraceType	Waveform qualifier
ValueType	
TOADone	
BaseUnitRelative	Base unit indication
UseInterSampleTriggerOffset	
ISO_TRG SC_POST SC_TRG	

11.2.1.2 Waveform Data Files

The waveform data files - indicated by `*Wfm.*` in the filename - contain the actual waveform data. Usually only Y-values - mostly voltage values - are written subsequently. If the signal is a spectrum, the data of the last frame is written.

If the waveform consists of minimum and maximum values, two Y-values per sample are written, and the property `InterleavedTraceCount` in the header file is >1. This applies to envelope waveforms, for example.

The option "Interleaved X/Y" allows you to include horizontal values into the file.

If multi-channel export is enabled, the Y-values of the selected channels are written in interleaved order.

- One channel, single acquisition export
  - Normal waveform:
    Y_0; Y_1; Y_2; Y_3; ...

  - Envelope waveform:
    $Ymin_0$; $Ymax_0$; $Ymin_1$; $Ymax_1$; $Ymin_2$; $Ymax_2$; $Ymin_3$; $Ymax_3$; ...
  - Normal waveform, interleaved x/y data:
    X_0; Y_0; X_1; Y_1; X_2; Y_2; X_3; Y_3; ...
  - Envelope waveform, interleaved x/y data:
    X_0;$Ymin_0$; $Ymax_0$; X_1; $Ymin_1$; $Ymax_1$; X_2; $Ymin_2$; $Ymax_2$; X_3; $Ymin_3$; $Ymax_3$; ...
- Multi-channel, single acquisition export
  In the example, two channels are exported.
  - Normal waveforms:
    $YCh1_0$; $YCh2_0$; $YCh1_1$; $YCh2_1$; $YCh1_2$; $YCh2_2$; $YCh1_3$; $YCh2_3$; ...
  - Envelope waveforms, channel 1 and channel 2 are envelopes:
    $YCh1min_0$; $YCh1max_0$; $YCh2min_0$; $YCh2max_0$; $Ymin_1$; $Ymax_1$; $YCh2min_1$; $YCh2max_1$; $Ymin_2$; $Ymax_2$; $YCh2min_2$; $YCh2max_2$; $Ymin_3$; $Ymax_3$; $YCh2min_3$; $YCh2max_3$; ...
  - Normal waveforms, interleaved x/y data:
    X_0; $YCh1_0$; $YCh2_0$; X_1; $YCh1_1$; $YCh2_1$; X_2; $YCh1_2$; $YCh2_2$; X_3; $YCh1_3$; $YCh2_3$; ...
  - Envelope waveform and normal waveform, interleaved x/y data:
    X_0;$YCh1min_0$; $YCh1max_0$; $YCh2_0$; X_1; $YCh1min_1$; $YCh1max_1$; $YCh2_1$; X_2; $YCh1min_2$; $YCh1max_2$; $YCh2_2$; X_3; $YCh1min_3$; $YCh1max_3$; $YCh2_3$; ...

In XML and CSV waveform value files, the data of each sample is grouped. The example shows the values of two samples for two waveforms and interleaved x/y data. The first waveform is an envelope, the second one is a normal waveform.

In CSV files, the data values for a given sampling time are written in one row.

```
-1.96e-008    -0.0079051387    -0.0059288535    -0.1027668
-1.95e-008    -0.0098814229    -0.0079051387    -0.10474309
```

In XML format, an empty line marks the beginning of the next sample.

```
<Data>-1.96e-008</Data>
<Data>-0.0079051387 </Data>
<Data>-0.0059288535 </Data>
<Data>-0.1027668 </Data>

<Data>-1.95e-008</Data>
<Data>-0.0098814229 </Data>
<Data>-0.0079051387 </Data>
<Data>-0.1027668 </Data>
```

If multiple acquisitions (Data logging / Multiple waveforms) are exported, the first acquisition is written in the same way as with single acquisition export. The following acquisitions are appended in the same way. If the signal is a spectrum, the last frame of each acquisition is saved.

In BIN files, the instrument writes some leading and trailing settling samples before and after the waveform data. They ensure that the analysis of the reloaded reference waveform returns the same results as analysis of the original waveform. The number of leading settling samples is provided in the header file.

11.2.1.3 Number of Samples in the Waveform Data File

In this section, a sample is defined as one or more values acquired at a given sampling time. The number of samples for one channel and acquisition is given in the header file by the property `SignalHardwareRecordLength`.

If the waveform has more than one Y-value per sample (e.g. envelope), the property `InterleavedTraceCount` is > 1, and the number of values in the file for this waveform is:

*No of values per waveform = InterleavedTraceCount * SignalHardwareRecordLength*

If multiple acquisitions are exported, the total number of values in the file is:

*No of values = InterleavedTraceCount * SignalHardwareRecordLength * No of exported acquisitions*

If "Interleaved x/y"is enabled, one horizontal value is added per sample. The total number of values in the file is:

*No of values = (1+ InterleavedTraceCount) * SignalHardwareRecordLength * No of exported acquisitions*

In BIN files, the value `SignalHardwareRecordLength` includes the number of required samples in the waveform and additional samples at the beginning (leading samples) and the end of the file (trailing samples). The number of additional samples is:

No of additional samples = SignalHardwareRecordLength - SignalRecordLength

The number of leading additional samples is given in the header file: `LeadingSettlingSamples`.

The number of trailing additional samples is:

No of trailing additional samples = No of additional samples - LeadingSettlingSamples
= SignalHardwareRecordLength - SignalRecordLength - LeadingSettlingSamples

MSO option R&S RTE-B1:

If the data of digital channels is stored in BIN format, 1 bit is written for each sample. 8 data samples are written in 1 byte (data word). Thus, the file size is

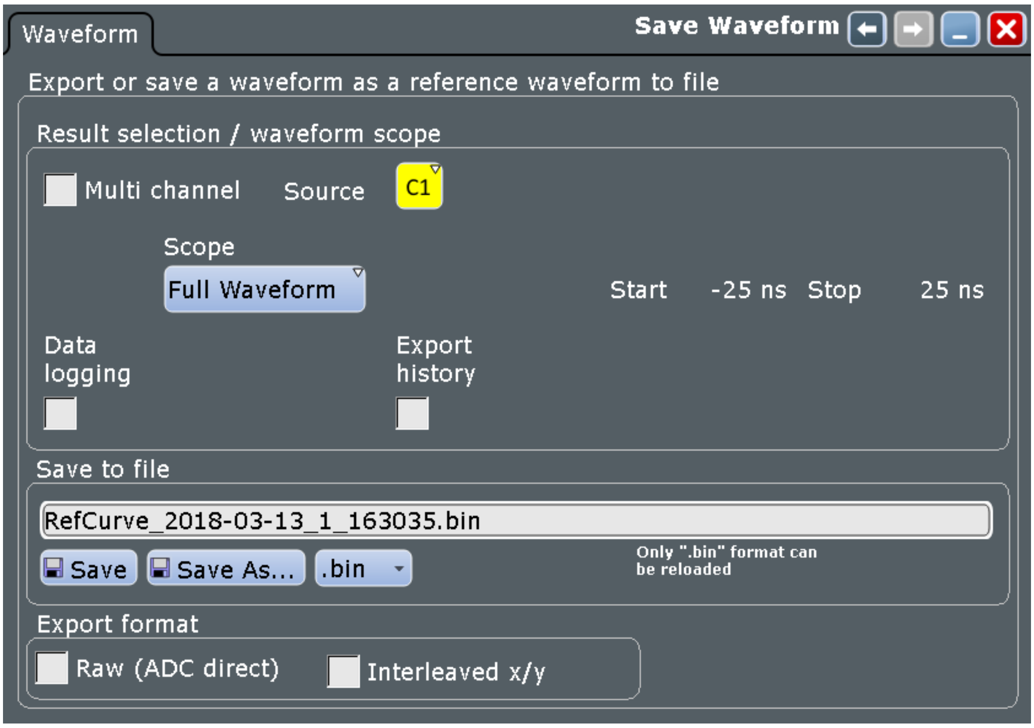
File size = Number of samples / 8

For example, 100 MSa are written into a 12.5 MByte BIN file. After reading the file, you have to extract the samples from the data words.

11.2.2 Waveforms - Export Settings

Access: [FILE] > "Waveform"

In this tab, the storage settings for waveform data are defined.



See also: Chapter 11.2.7, "Saving and Loading Waveform Data", on page 457.

Source

Selects the waveform to be exported if "Multichannel export" is disabled.

Active waveforms of input channels, math signals and reference waveforms are available for export.

If the MSO option R&S RTE-B1 is installed, you can save also digital channels and parallel buses.

Remote command:
`EXPort:WAVeform:SOURce` on page 1363

Multi-channel export

Enables or disables the export of multiple input channels. If enabled, you can export the data of selected input channels (Selected sources) into one file.

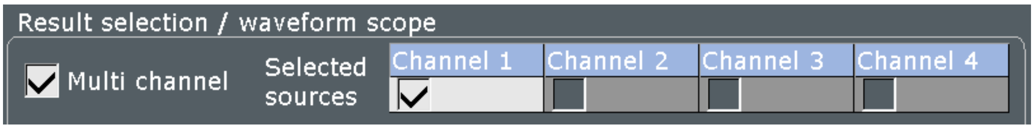
If disabled, you can export one Source waveform.

You can reload exported multiple channels if they are stored in BIN format. The import asks you to assign each stored waveform to a reference waveform.

Remote command:
`EXPort:WAVeform:MULTichannel` on page 1364

Selected sources

Select the channels to be included in data export if "Multichannel export" is enabled. Waveform1 of up to four input channels can be saved into one file.



Remote command:
`CHANnel<m>:EXPortstate` on page 1364

Scope

Defines the part of the waveform record that has to be stored.

"Full wave-form" Saves the complete waveform record.

"Zoom" Saves the data included in the zoom area if at least one zoom is defined for the source waveform. The start and stop values of the area are shown. If several zooms are defined, select the "Zoom" to be used for export.

"Cursor" Saves the data between the cursor lines if at least one cursor measurement is defined for the source waveform. The start and stop values of the area between the cursor lines are shown. If several cursor sets are defined, select the "Cursor set" to be used for export.

"Gate" Saves the data included in the measurement gate if a gated measurement is defined for the source waveform. Select the "Measurement" for which the required gate is defined. The start and stop values of the gate are shown.

"Manual" Saves the data between user-defined "Start" and "Stop" values.

Remote command:
`EXPort:WAVeform:SCOPe` on page 1365
`EXPort:WAVeform:STARt` on page 1365
`EXPort:WAVeform:STOP` on page 1366
`EXPort:WAVeform:ZOOM` on page 1366
`EXPort:WAVeform:CURSorset` on page 1366
`EXPort:WAVeform:MEAS` on page 1366

Data logging / Multiple Wfms

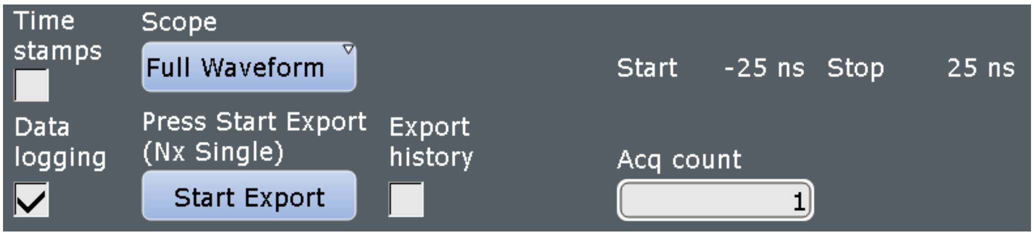
The "Data logging / Multiple Wfms" setting enables the export of subsequent acquisitions of the selected waveforms. If "Export history" is disabled, the setting is named "Data logging", and it exports the data of a running Nx Single acquisition. If "Export history" is enabled, the setting is named "Multiple Wfms", and it exports the history waveform data to file.

If multiple acquisitions of one waveform are exported into a BIN file, the first acquisition can be reloaded as reference waveform.

"Data logging" enables the export of all waveforms of an Nx Single acquisition into one file. The waveform records are written in historical order one after the other, either the complete records or the sections as defined in "Scope". Set the number of acquisitions to be acquired and stored with "Acq. count". The maximum amount of data that can be written is shown in "Max. file size".

Enabling "Data logging" stops a running acquisition. To start the logging, tap Start Export or press [RUN N× SINGLE].

Pressing "Run cont" disables data logging.



If "Export history" is enabled, the option "Multiple Wfms" allows you to save several or all history waveforms. Define the part of the history to be exported using "Start acq" and "Stop acq".

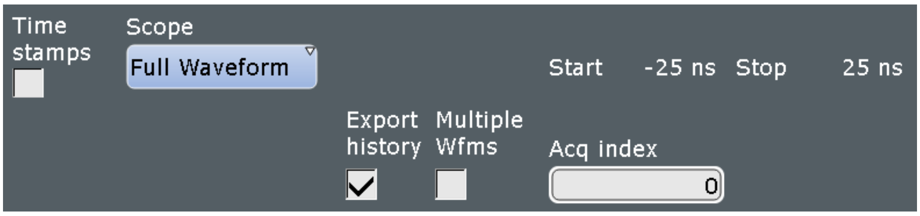
Remote command:
`EXPort:WAVeform:DLOGging` on page 1367

Start Export
Starts an Nx Single acquisition series and simultaneously saves the waveform data to a file if data logging is enabled.

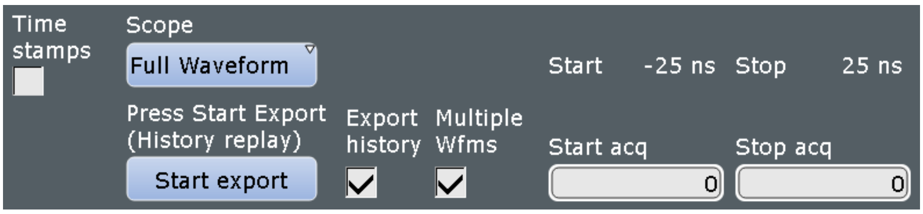
If "Export history" is enabled, the button starts the history replay and simultaneously saves the history waveforms.

Remote command:
`RUNSingle` on page 1070 (Nx Single acquisition)
`CHANnel<m>[:WAVeform<n>]:HISTory:PLAY` on page 1196 (history export)

Export history
Enables the history mode and the export of history waveforms to file. The setting is also available in the "History" dialog box under the designation"Show history".



To save one waveform from the history, enter the number of the required acquisition in "Acq index", and tap "Save".



To save several subsequent history waveforms, enable "Multiple Wfms" and define the range of the waveforms to be saved using "Start acq" and "Stop acq". These range settings are also available in the "History" dialog box. Start the history replay and simultaneous saving with "Start Export".

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]` on page 1194
`CHANnel<m>[:WAVeform<n>]:HISTory:STARt` on page 1195
`CHANnel<m>[:WAVeform<n>]:HISTory:STOP` on page 1195

Timestamps
If enabled, the relative timestamps of all history waveforms are written into the waveform data file at the beginning of each waveform record.

Remote command:
`EXPort:WAVeform:TIMestamps` on page 1368

Save to file
Enter the filename to save the waveform to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "RefCurves_". You can define a pattern for automatic naming in the "Autonaming" tab.

"Save"	Saves the waveform as a reference waveform in the selected file.
"Save As..."	Opens the file selection dialog box and saves the waveform to the selected file. See also Chapter 11.7, "File Selection Dialog", on page 471
".bin/.xml/.csv"	Selects the file format. Note that reference waveforms can be loaded from `.bin` files only. See also: Chapter 11.2.1, "Waveform Export Files", on page 439.

Remote command:
`EXPort:WAVeform:NAME` on page 1364
`EXPort:WAVeform:SAVE` on page 1365

Interleaved x/y
Includes horizontal values in the export data (time or frequency values, depending on the waveform). X and Y-values are written alternately to the file. If disabled, only Y-values - mostly voltage values - are written.

Interleaved x/y data cannot be exported as raw values, the "Raw (ADC direct)" option is not available.

Remote command:
`EXPort:WAVeform:INCXvalues` on page 1368

Raw (ADC direct)
Enables the export of analog channel data in the raw sample format of the ADC. The data format is integer 8 bit (signed 8-bit binary format). This format reduces the file size (1 Byte/sample instead of 4 Bytes/sample in binary files) but decreases the precision of the values.

If the high definition mode is active, the data format is integer 16 bit, except for peak detect decimation (8 bit). See "Export" on page 154.

Only y-values are exported, the "Interleaved x/y" option is not available.

Currently, the setting is not available for the export of digital channel data and data of R&S RT-ZVC channels.

Data conversion:

To convert INT8 or INT16 data to physical quantities, e.g. voltages, use the following formulas:

*ConversionFactor = VerticalScale * VerticalDivisionCount / NofQuantisationLevels*

*PhysicalQuantity = (Value_ADC * ConversionFactor) + VerticalOffset - (VerticalScale * VerticalPosition)*

The raw values are written in the `*.Wfm.*` file, all other values can be found in the corresponding header file.

Table 11-2: Example of raw data conversion

	INT8	INT16, HD mode
VerticalScale	0.05	0.05
VerticalPosition	0	0
VerticalOffset	0	0
NofQuantisationLevels	253	253 * 256
VerticalDivisionCount	10	10
Value_ADC	-61	-61
ConversionFactor	0.05 * 10 / 253 = 0.00197628	0.05 * 10 / (253 * 256) = 0.0000771986
Voltage	(-61 * 0.00197628) + 0 = -120.5 mV	(-61 * 0.0000771986) + 0 = -4.7091146 mV

Remote command:
`EXPort:WAVeform:RAW` on page 1368

Byte order
Sets the endianess for INT16 data:
- LSB first: little endian, least significant byte first
- MSB first: big endian, most significant byte first

Remote command:
`FORMat:BORDer` on page 1046

11.2.3 Waveform Histogram

Access: [FILE] > "Histogram"

The waveform histogram export saves data in two files. The `*.Wfm.*` file contains 256 or 512 absolute or relative histogram values. The other file is the header file.

Contents of the header file:

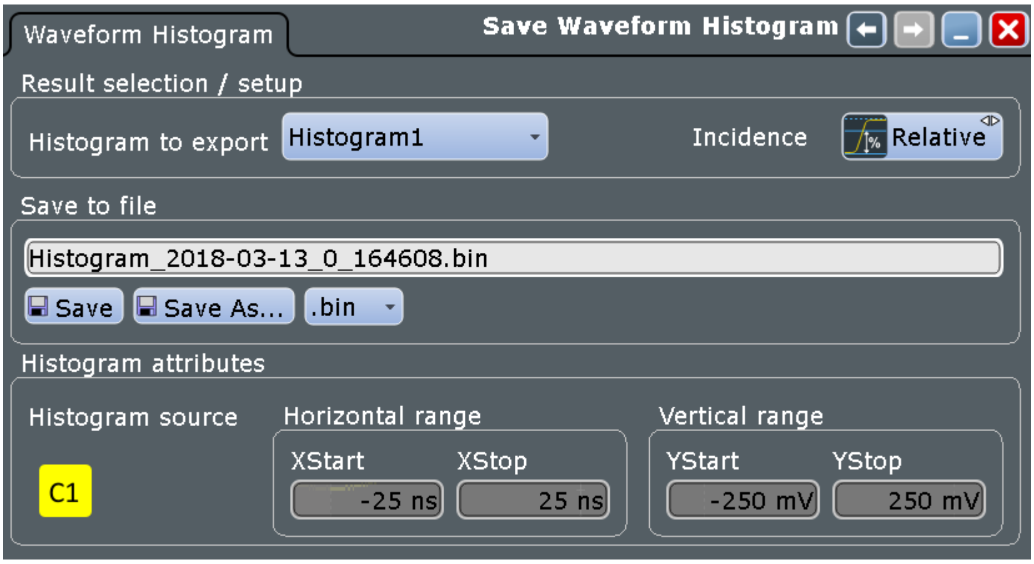
- Source waveform of the histogram
- Histogram mode: vertical or horizontal
- Incidence of exported values: absolute or relative
- Histogram range: XStart, XStop, YStart, YStop
- Name of the exported histogram

Data conversion:

Using the header data, you can calculate the waveform value to which a histogram value belongs:

*YValue = (YStop - YStart) / HistogramValuesCount * HistogramValueNumber + YStart*

YStart	-0.25 V
YStop	0.25 V
HistogramValuesCount	256 (total number of written rows in a CSV file)
HistogramValueNumber	68 (number of the row in a CSV file)
Y-Value	(0.25 - (-0.25)) / 256 * 68 - 0.25 = -0.11719 V



Histogram to export
Selects the histogram to be exported. All active waveform histograms are shown in the list.

Measurement histograms can also be exported, see Chapter 11.2.5, "Result Analysis", on page 454.

Remote command:
`EXPort:HISTogram:SELect` on page 1369

Incidence
Sets the mode of exported histogram data: relative or absolute count of values. If relative values are exported, the sum of all values is 1, and the count of each value is set in relation to the sum.

Remote command:
`EXPort:HISTogram:INCidence` on page 1370

Save to file
Enter the filename to save the waveform histogram to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "Histogram_". You can define a pattern for automatic naming in the "Autonaming" tab. The default directory is:

`C:\Users\Public\Documents\Rohde-Schwarz\RTE\Histograms`

"Save"	Saves the histogram data in the selected file.
"Save As..."	Opens the file selection dialog box and saves the histogram data to the selected file. See also Chapter 11.7, "File Selection Dialog", on page 471
".bin/.xml/.csv"	Selects the file format.

Remote command:
`EXPort:HISTogram:NAME` on page 1370
`EXPort:HISTogram:SAVE` on page 1370
`EXPort:HISTogram:DATA?` on page 1370

Histogram source, Horizontal range, Vertical range
Show the source and the limits of the histogram area for information. The ranges are set in the "Histogram" dialog box ("Meas" menu > "Histogram"). See also: Chapter 7.2.8.4, "Histogram Setup", on page 334.

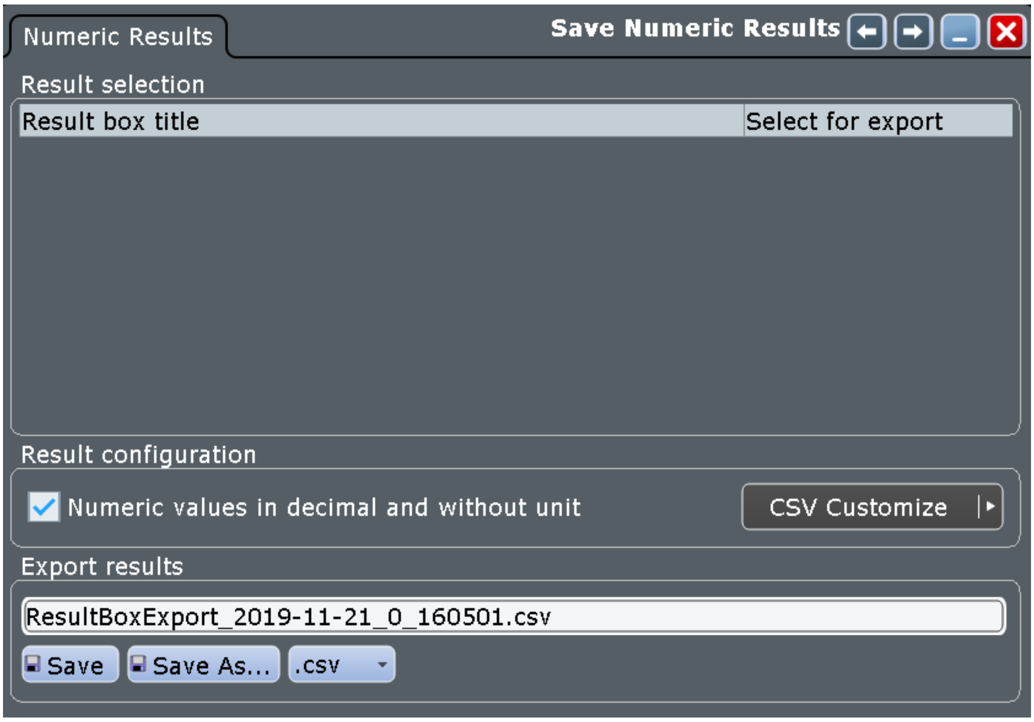
11.2.4 Numeric Results

In this tab, you can select the result boxes to be saved, and define the storage settings.

Access: [FILE] > "Numeric Results"

Access to the tab is available in all tabs where measurement and analysis settings are defined, for example, in the "Measurements Setup", "Cursors Setup", and "Masks Test Definition" tabs: Simply tap the "Result Export" button.

Furthermore, you can export the decode result tables of serial protocol analysis.



Result selection
The table lists all result boxes and decode tables that are currently open, including minimized boxes and docked boxes. Select the results that you want to save to file. All results are written into one file.

Note: If the result box is minimized, only the columns shown on the result icon are saved (2 columns). Statistical results are not shown on the minimized results icon, and they are not saved.

Remote command:
`EXPort:RESult:SELect` on page 1371

Numeric values in decimal and without unit
By default, numeric result values are written with their unit to the file. If the option is enabled, the values are saved with more decimal places.

Remote command:
`EXPort:RESult:NUMeric` on page 1371

CSV Customize
Opens a dialog for customizing the settings of the CSV file. See Chapter 11.2.6, "CSV Export", on page 456.

Export results
Enter the filename to save the results to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "ResultBoxExport_". You can define a pattern for automatic naming in the "Autonaming" tab.

"Save" Saves the selected results to the indicated file.

"Save As..." Opens the file selection dialog box and saves the selected results to the selected file. See also Chapter 11.7, "File Selection Dialog", on page 471

".csv/.html" Selects the file format.

- CSV: the values are saved in a file
  You can select the value delimiter and the list separator symbol in the "CSV Export" dialog, see Chapter 11.2.6, "CSV Export", on page 456.
  Tip for using MS Excel: It is recommended that you use the semi-colon as csv decimal symbol. When you open the file with MS Excel, use "File > Open" and follow the wizard to set the separators correctly, or set the separator settings with "Tools > Options > International".
- HTML: Results are saved as web page for display in a browser.

Remote command:
`EXPort:RESult:NAME` on page 1371
`EXPort:RESult:SAVE` on page 1371

11.2.5 Result Analysis

Access: [FILE] > "Result Analysis"

You can export the data of long-term measurements, the measurement histogram and track data to file.

The measurement export saves results in two files. The `*.Wfm.*` file contains data values, and the other file is the header file.

The header file contains:

- Source waveform of the measurement
- Measurement scale
- Export type = Histogram, Long term or Track
- Exported measurement
- Histogram range: XStart, XStop, YStart, YStop
  The range is only relevant for export type = histogram. The measurement axis is the X-axis, which can be a horizontal or vertical axis depending on the histogram mode.

Long-term measurements: The `*.Wfm.*` file contains one value or value set for each long-term measurement point. The maximum number of points is defined in the "Horizontal scaling" dialog box.

- If statistics are disabled, the current result of the main measurement is written - one double value per long-term point.
- If statistics are enabled, seven values for each long-term point are saved:
  - Upper peak
  - Lower peak
  - RMS

- Standard deviation
- Average
- Event count per point: number of measurement results that creates one long-term point
- Waveform count per point: number of waveforms included in one long-term point.

Measurement histogram: The `*.Wfm.*` file contains 1000 absolute or relative histogram values.

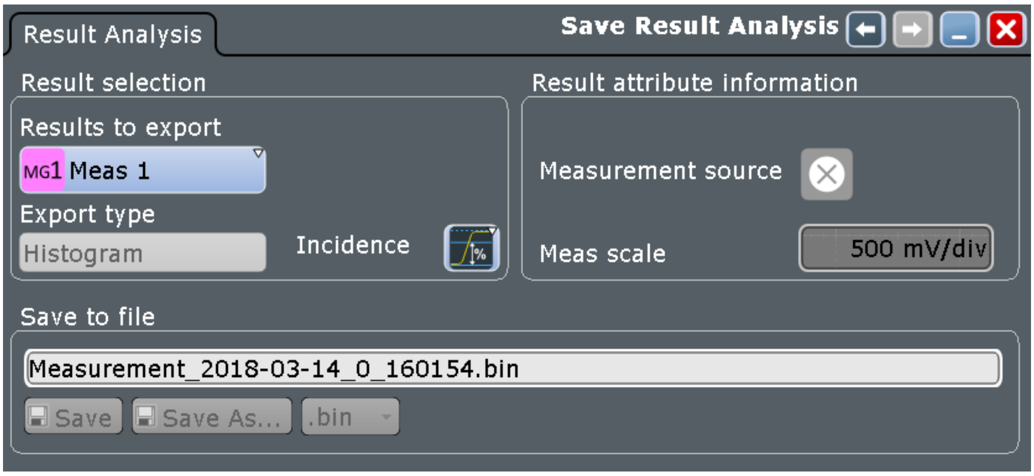
Data conversion of measurement histogram data:

Using the header data, you can calculate the measurement value to which a histogram value belongs:

*MeasValue = (XStop - XStart) / 1000 * HistogramValueNumber + XStart*

Example: The 273rd histogram value is 0.491749. That means, the relative frequency of the measurement value 0.1246 V is 0.491749.

XStart	0.07 V
XStop	0.27 V
HistogramValueNumber	273 (number of the row in a CSV file)
MeasValue	(0.27 - 0.07) / 1000 * 273 + 0.07 = 0.1246 V



Results to export
Selects the measurement to be exported.

Remote command:
`EXPort:MEASurement:SELect` on page 1372

Export type
You can export the result data of the long-term measurement, the measurement histogram or the track. To export the data, the required type must be enabled in "Measurements > Long Term/Track": "Long term Enable", "Histogram Enable" or "Track".

Remote command:
EXPort:MEASurement:TYPE on page 1372

Incidence
Sets the mode of exported histogram data: relative or absolute count of values. If relative values are exported, the sum of all values is 1, and the count of each value is set in relation to the sum.

Remote command:
EXPort:HISTogram:INCidence on page 1370

Measurement source, Meas scale
Show the measurement settings source and scale for information.

Save to file
Enter the filename to save the measurement data to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "Measurement_". You can define a pattern for automatic naming in the "Autonaming" tab. The default directory is:

C:\Users\Public\Documents\Rohde-Schwarz\RTE\Measurements

"Save"	Saves the measurement data in the selected file.
"Save As..."	Opens the file selection dialog box and saves the measurement data to the selected file. See also Chapter 11.7, "File Selection Dialog", on page 471
".bin/.xml/.csv"	Selects the file format.

Remote command:
EXPort:HISTogram:NAME on page 1370
EXPort:MEASurement:SAVE on page 1373
EXPort:MEASurement:DATA? on page 1373

11.2.6 CSV Export

Access: [FILE] > "Save" tab > "CSV Export".

In this dialog, you can define the format of your CSV file.

Decimal symbol
Selects if point or comma is used as a decimal symbol in the exported CSV file.

Remote command:
EXPort:RESult:DECSymbol on page 1372

List separator
Selects the list separator symbol from a list. Available are semicolon, comma, space, tab and colon.

Remote command:
EXPort:RESult:DELimiter on page 1372

11.2.7 Saving and Loading Waveform Data

You can save the data of a channel, math or reference waveform to an `.xml`, `.csv`, or `.bin` file. The data export of several channels into one file is also possible. Files in `.bin` format can be reloaded to the R&S RTE as reference waveforms.

Instead of a complete waveform, you can also save a part of it, limited by a previously defined zoom, cursor lines, measurement gate or user-defined time values.

To save waveform data quickly, you can add the "Save Waveform" icon to the toolbar and use it for saving. The icon does not work for saving actions that are started with "Start export" (data logging and multiple history waveforms).

It is also possible to save history data to file. Furthermore, you can save a "live record" of a running RUN Nx SINGLE acquisition to one data file.

For details on waveform save/recall settings, see Chapter 11.2.2, "Waveforms - Export Settings", on page 445.

The following procedures are described:
- "To save a waveform or a part of a waveform to a file" on page 457
- "To save a waveform using the toolbar icon" on page 458
- "To export waveform data of a running acquisition" on page 458
- "To save the history data" on page 279
- "To load a reference waveform" on page 253
  "To save a reference waveform" on page 253

To save a waveform or a part of a waveform to a file

1. Press the [FILE] key.
2. In the "Save" tab, tap the "Waveform" button.
3. Select the waveforms to be saved:
   - To save one waveform, tap the "Source" icon and select the waveform.
   - To save data of several channels, enable "Multi channel" and select the channels.
4. In the "Scope" list, select the part of the waveform record to be saved. Zoom, cursor and gate segments require the same setup for the selected waveform before saving. For "Manual", enter the "Start" and "Stop" time of the section.

5. Check the filename under "Save to file" and change it if needed. Usually, autonaming is used.
6. Check the file format and the "Export format" settings and change them if needed.
7. Tap "Save" to save the waveform data to the specified file.
   Tap "Save As" to save the waveform data to a different file or file type. Select the file from the file selection dialog box.

To save a waveform using the toolbar icon

1. Add the "Save Waveform" icon to the toolbar, see Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.
2. Set the scope, export format, and other parameters in [FILE] > "Waveforms / Results" > "Waveforms".
3. If necessary, adjust the autonaming pattern and the storage path in [FILE] > "Autonaming".
4. To save the waveform data:
   a) Tap the "Save Waveform" icon on the toolbar.
   
   b) Tap the waveform to be saved.
   If you tap the diagram background, the data of the focused waveform is saved.

To export waveform data of a running acquisition

1. Select the waveforms to be saved and the scope as described in "To save a waveform or a part of a waveform to a file" on page 457, step 1 to 5.
2. If you want to save only a section of each waveform, set the "Scope".
3. Enable "Data logging".
4. Enter the number of acquisitions to be acquired and saved in "Acq count".
5. Check the filename under "Save to file" and change it, if needed. Usually, autonaming is used.
6. Tap "Start Export" to start the acquisition and to save the acquired waveform data to the specified file.

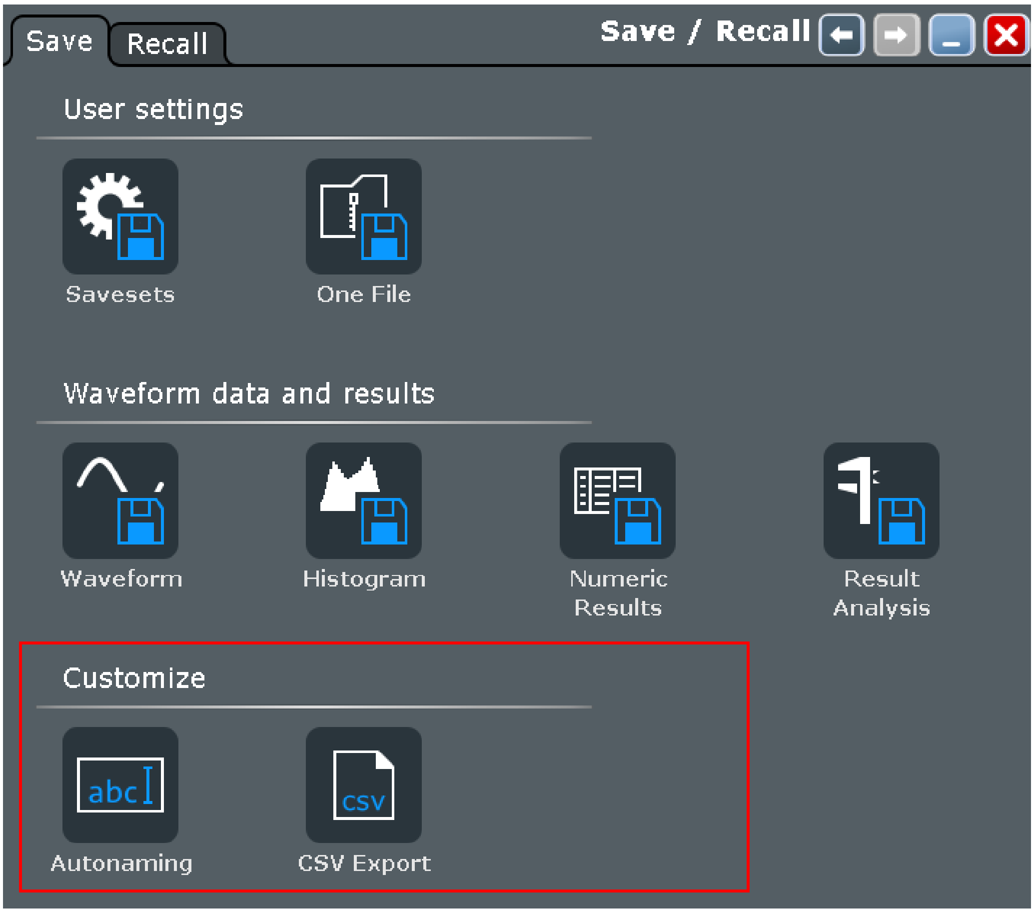
To load waveform data as a reference waveform

To reload waveform data from a previous measurement, the waveform must have been stored as a reference waveform in a BIN file before.

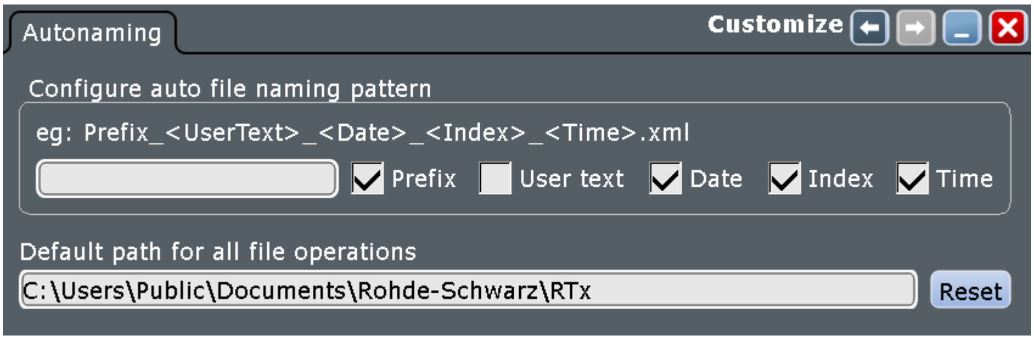
The procedure is described in "To load a reference waveform" on page 253

11.3 Autonaming

Access: [FILE] > "Save" tab



11.3.1 Autonaming Settings



In this tab, you can define the pattern for automatic file name generation. This name is used as the default file name. The default path is the storage location for all saved files and their subdirectories.

Prefix
If enabled, inserts the default prefix in the file name. The prefix indicates the type of data that is saved, for example, Histogram, RefCurve, Settings.

Remote command:
`MMEMory:AUTonaming:PREFix` on page 1361

Text input
User-defined text to be inserted after the prefix.

Remote command:
`MMEMory:AUTonaming:TEXT` on page 1361

User text (enable)
If enabled, inserts the specified user text after the prefix.

Remote command:
`MMEMory:AUTonaming:USERtext` on page 1361

Date
If enabled, inserts the current date.

Remote command:
`MMEMory:AUTonaming:DATE` on page 1361

Index
If enabled, inserts an index.

Remote command:
`MMEMory:AUTonaming:INDex` on page 1361

Time
If enabled, inserts the current time.

Remote command:
`MMEMory:AUTonaming:TIME` on page 1361

Default path for all file operations
Defines the default path displayed in the file selection dialog box for loading and storing operations. If a USB flash drive is connected, the path is set automatically to the drive letter of the USB flash drive.

To switch the directory quickly, double-tap the input field. Use the symbols on the left of the file explorer box to change the directory.

Remote command:
`MMEMory:AUTonaming:DEFaultpath` on page 1361

Reset
Resets the default file path to the factory default.

Remote command:
`MMEMory:AUTonaming:RESPath` on page 1362
`MMEMory:AUTonaming:RESall` on page 1362

11.3.2 Defining Default File Paths and Names

When a save or load operation is performed, a default file name and path is provided. You can configure which path is used and how the file name is generated. In the file selection dialog box, you can change the folder and name as desired.

To define the default file path

1. Press the [FILE] key.
2. Select the "Save" tab.
3. Press the "Autonaming" button.
4. Double-tap the "Default path for all file operations" field.
   The directory selection dialog box is opened.
5. Select the folder in which the data is stored by default. Use the symbols on the left of the file explorer box to switch to often used directories.
6. To restore the factory-set default path, tap "Reset" next to the path field.

To define the automatic file name pattern

The automatic file name pattern can consist of the following elements:

<Prefix>_<UserText>_<Date>_<Index>_<Time>

The prefix depends on the data type to be stored and cannot be changed by the user. The other elements can be enabled or disabled as required.

1. Press the [FILE] key.
2. Select the "Autonaming" tab.
3. To insert a user-defined text after the prefix, enter the text in the edit field. and enable "User text".
4. If you want to exclude the prefix, current date, time or an index (serial number), disable the corresponding option.
   The specified elements are used to generate the default file name for the next storage operation.

11.4 Screenshots

To store the graphical results of the measurement, you can either print the current display on a printer or save an image to a file. The instrument saves or prints a screenshot of the graphic area. To document current settings, the open dialog box can be included in the screenshot.

The "Save Screenshot" toolbar icon saves the current display to a file according to the settings in "File" menu > "Print setup". See also Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.

You can configure the [PRINT] key to save or print screenshots by a single keypress. See also Chapter 3.3.2, "Hardkeys: Function Assignment", on page 101.

If a USB flash drive is connected to the instrument, the default path of the user data directory is set to the drive letter of the USB flash drive. Thus, you save data to USB flash drive automatically, and you can change the directory in the file explorer at any time.

Screenshots on a computer using the Web interface

If the R&S RTE is connected to a LAN, you can create and save screenshots of the instrument's display on a computer. See Chapter 16.3.2, "Web Browser", on page 1003.

Meta information in screenshots

The meta data of the screenshot also contains instrument information. In PNG and JPEG files, meta information is saved as EXIF information and can be read, for example, using the ExifTool.

Example:

Reading meta information using the ExifTool.

Command: `# exif C:\Screenshot_2016-07-14_0_110551.png`

Result:

```
ExifTool Version Number          : 10.20
File Name                        : Screenshot_2016-07-14_0_110551.png
Directory                        : C:/
File Size                        : 37 kB
File Modification Date/Time      : 2016:07:14 11:05:51+02:00
File Access Date/Time            : 2016:07:14 11:05:51+02:00
File Creation Date/Time          : 2016:07:14 11:05:51+02:00
...
Instrument Firmware Version      : 3.30.0.46
Instrument Material Number       : 1329.7002k44
Instrument Serial Number         : 123456
Image Size                       : 1280x800
Megapixels                       : 1.0
```

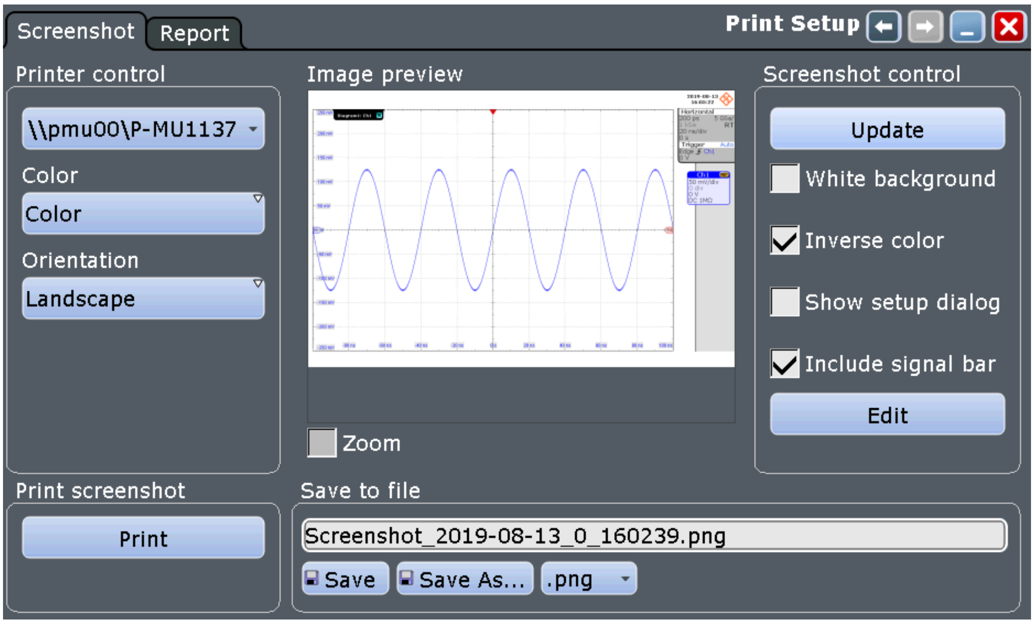
11.4.1 Screenshot Settings

Access: "File" menu > "Print setup"

In the "Screenshot" dialog box, you configure the image to be printed, saved, or included in a report. You select the printer and the storage location for screenshot files. The image is created when you open the dialog box, and can be updated at any time.

You can also edit the image before saving or printing, and include an open dialog box in the image. The signal bar is always shown beside the diagram.

You can print and save the image in the "Screenshot" dialog box. To save screenshots quickly, use the "Save Screenshot" toolbar icon, or configure and use the [PRINT] key.



Printer....................463
Color....................463
Orientation....................464
Print....................464
Preview....................464
└ Zoom....................464
Update....................464
White background....................464
Inverse color....................464
Show setup dialog....................465
Include signal bar....................465
Edit....................465
Save to file....................465

Printer

Selects a configured printer to print screenshots and reports. You can use a local printer or a network printer. The instrument firmware uses the Windows printer configuration, no additional printer setup is required. To make a printer available for R&S RTE, add and configure it in the Windows operating system: "Settings" > "Devices" > "Printers and scanners". Only users with administrator rights can fulfill this task.

Depending on the printer driver, printing to a file is also possible. By default, the Rohde & Schwarz printer drivers for BMP, JPG, PDF, PNG, and TIFF files are installed.

Remote command:

`SYSTem:COMMunicate:PRINter:SELect<1..2>` on page 1379

Color

Defines the color mode for printing. The setting affects the output on a printer and also the printing to BMP, JPG, PDF, PNG, and TIFF files.

"Black and white"	Black and white output

"Color" Color output

Remote command:
HCOPy:DEVice<m>:COLor on page 1376

Orientation
Toggles the page orientation between "Landscape" and "Portrait."

Remote command:
HCOPy:PAGE:ORIentation<1..2> on page 1376

Print
Prints the current image together with saved editing changes on the selected Printer.

If the printer is configured to print to a file, "Print" is an alternative of "Save image to file".

Remote command:
HCOPy:DESTination<1..2> on page 1374
HCOPy:IMMediate<m>[:DUM] on page 1378
HCOPy:IMMediate<m>:NEXT on page 1378

Preview
Shows a preview of the screenshot. The image is created when the dialog box opens.

Zoom ← Preview
Enlarges the preview display and adds scrollbars to zoom into specific areas of the print image. Zooming does not affect the original display.

Update
Updates the preview of the screenshot with the current display view, e.g. after changes to the settings have been made, or an additional channel has been activated.

White background
Inverts the background color. So you can print waveforms with normal waveform colors on white background.

If both "White background" and Inverse color are enabled, the instrument inverts the background twice, and it appears black.

"White background"	"Inverse color"	Background	Waveform and results
On	Off	White	Screen colors
Off	On	White	Inverted colors
On	On	Black	Inverted colors
Off	Off	Black	Screen colors

Remote command:
HCOPy:WBKG on page 1376

Inverse color
Inverts the colors of the output, i.e. a dark waveform is printed on a white background.

See also: "White background" on page 464.

Remote command:
HCOPy:DEVice<m>:INVerse on page 1376

Show setup dialog
If you want to save dialog boxes in screenshots, enable "Show setup dialog". The currently open dialog box is included in the screenshot. Use the [PRINT] key to print or save the display.

Remote command:
HCOPy:SSD on page 1377

Include signal bar
If enabled, the screenshot shows the signal bar beside the diagram area.

Remote command:
HCOPy:ISBA on page 1377

Edit
Opens the screenshot in the Paint application. Edit the image as necessary. You can store the file using "Save as" or print the file from Paint. Alternatively, save the file and close the Paint application to return to the "Print Setup" dialog, then print or save the edited image. The changes are not shown in the preview.

Save to file
Defines the filename to which screenshot is saved. By default, the filename has the prefix "Screenshot_". Double-tap the filename filed to change the name.

If a USB flash drive is connected to the instrument, the default path of the user data directory is set to the drive letter of the USB flash drive. Thus, you save data to USB flash drive automatically, and you can change the directory in the file explorer at any time.

"Save"	Saves the current screenshot to the specified file.
"Save As..."	Opens the file selection dialog box. Here you can adjust the target directory and the file name and save the current screenshot to the file. The symbols of important target folders are listed on the left of the file explorer.
"Delete"	Opens the file selection dialog box and deletes the selected file.

Remote command:
HCOPy:DEVice<m>:LANGuage on page 1375
HCOPy:DESTination<1..2> on page 1374
MMEMory:NAME on page 1375
HCOPy:IMMediate<m>[:DUM] on page 1378
HCOPy:IMMediate<m>:NEXT on page 1378
MMEMory:DELete on page 1357

11.4.2 Printing Screenshots

You can configure the format and colors used for printing, and edit the image.

1. Open the "File" menu and select "Print setup". You can use a local printer or a network printer. Depending on the printer driver, printing to a file is also possible. See also "Printer" on page 463.
2. Tap the printer selection box. Select the printer.
3. Tap the "Color" selection box to configure black and white or color images.
4. Tap the "Orientation" selection box and select the paper format.
5. To enhance the images for print on white paper, enable "White background" or "Inverse color".
6. If the current display is likely to have changed since you have opened the "Print Setup" dialog box (e.g. due to a running measurement), tap "Update image".

   The screenshot is updated.
7. To zoom into the screenshot, enable the "Zoom" option beneath the preview area.

   The image is enlarged and scrollbars are displayed to scroll through the image.
8. To edit the image in an external application and process it further from there, tap "Edit image".
9. To print the image to the selected printer, tap "Print".

11.4.3 Configuring and Saving Screenshots

You can edit the image, invert all colors, and set the background color. A preview of the current image is shown for reference.

1. Open the "File" menu and select "Print setup".
2. To enhance the images for later print on white paper, enable "White background" or "Inverse color". If you print this image later on a monochrome printer, you get a grayscaled picture. The contrast of the resulting gray lines depends on waveform colors and the used printer.
3. Select the file format: png, jpg, or another one.
4. To change the directory, tap "Save As" and configure the path.

   The symbols of often used target folders are listed on the left of the file explorer. By default, screenshots are saved in the `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\ScreenShots` directory.
5. Tap "Save".

The file is saved and the dialog box closes.

6. Check if the screenshot is saved to the desired directory.
7. To save further screenshots, use one of the following ways:
   - Configure the [PRINT] key. Press the key to save a screenshot. See also Chapter 3.3, "Frontpanel Setup", on page 101.
   - Add the "Save Screenshot" icon to the toolbar. Tap the icon to save an image. See also Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.
   - In the "Print setup" dialog box, tap "Save" to save the image to the specified file.
   - To save the image with a dedicated filename or to another directory, open the "Print setup" dialog box and tap "Save As". Select the path, enter a filename, and tap "Save".

Printing on a black-and-white printer

- If you use the "White background" or "Inverse color" settings and save the image to a file, you get a grayscaled picture. The contrast of the resulting gray lines on the printout depends on waveform colors and the used printer.
- To get a monochrome image, set the "Color = Black and white" and tap "Print" to start the direct printout or the print to file. See also "Printer" on page 463.

11.5 Reports

Reports document the current measurement and test results. The report contains general information, current vertical and horizontal settings, trigger settings, active channels and all current results except for zoom and search results. A screenshot is also included.

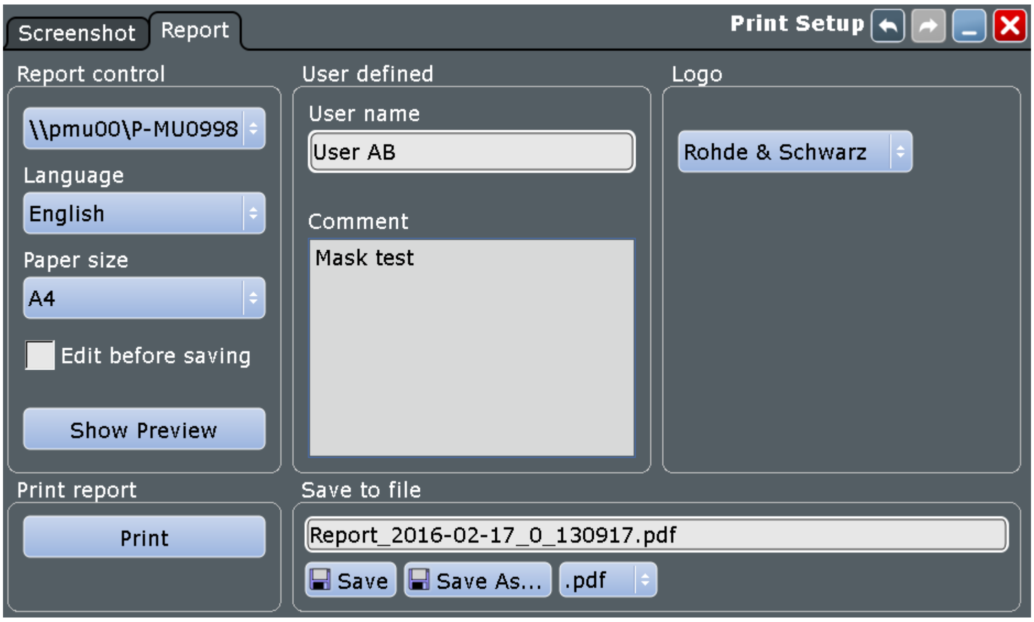
The report is configured in the "Report" tab, the screenshot is configured in the "Screenshot" tab.

You can create the report manually, or automatically on defined events:

- Press the [PRINT] key. Before, configure the key to create or print reports, see Chapter 3.3, "Frontpanel Setup", on page 101
- Tap the "Create report" toolbar icon.
  Before, add the icon to the toolbar, see also Chapter 2.4.7.2, "Configuring the Toolbar", on page 80.
- Action on micro button, available on active Rohde & Schwarz probes
- Action on trigger
- Event action at mask testing

11.5.1 Report Settings

Access: "File" menu > "Report Setup"



Printer
See "Printer" on page 463.

Language
Selects the language to be used in the report. Available languages are listed in the data sheet.

Remote command:
`REPort:LANGuage` on page 1379

Paper size
Selects the paper size: A4 or US Letter.

Remote command:
`REPort:PAPersize` on page 1380

Edit before saving
Enables you to edit the report info when you save reports using the [PRINT] key. When you press the key, a dialog box opens where you can change the user name and the comment.

Show Preview
Opens the current report in PDF format.

User name / Comment
Enter information that appears in the general information section at the beginning of the report.

Remote command:
`REPort:USER` on page 1380
`REPort:COMMent` on page 1380

Logo
By default, the Rohde & Schwarz logo is shown in the header of the report pages. You can switch the logo off, or select your logo to be shown. A preview of the selected log file is shown.

Remote command:
`REPort:LOGType` on page 1380
`REPort:LOGO` on page 1380

Print
Starts the printout to the configured printer.

Save to file
Select the file format and define the filename of the report file. By default, the filename has the prefix "Report_".

Double-tap the filename to open the file selection dialog box.

"pdf/doc/html"	Selects the report format.
"Save"	Saves the current report to the specified file.
"Save As..."	Opens the file selection dialog box and saves the report to the selected file.

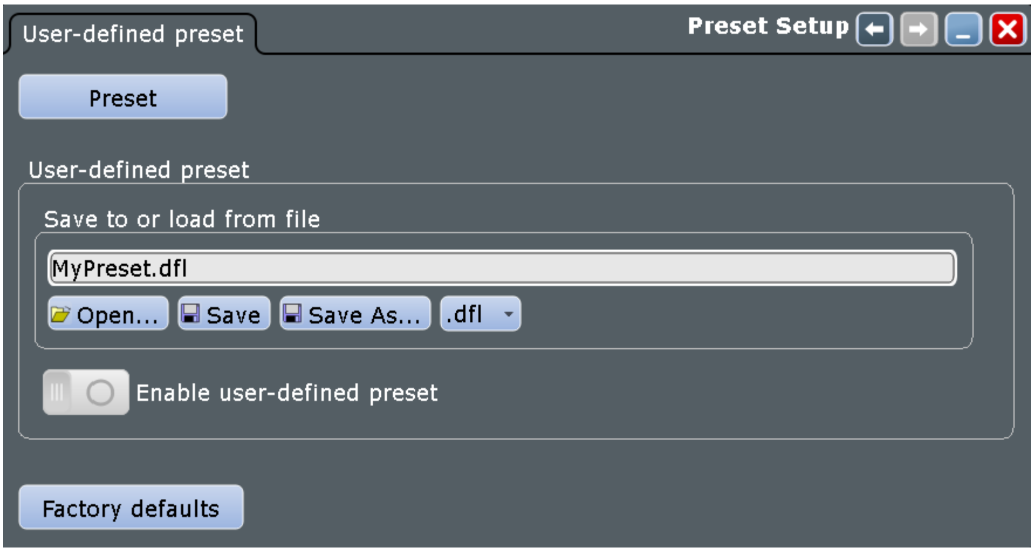
Remote command:
`REPort:FILE:NAME` on page 1381
`REPort:FILE:SAVE` on page 1381

11.6 Preset Setup

A user-defined preset contains the complete instrument setup including display settings, except for transparency and intensity. You can save the current configuration to a preset file, and load a previously saved preset file. You can then specify that these settings are to be applied with the [PRESET] key.

11.6.1 User-defined Preset - Settings

Access: "File" menu > "Preset setup"



Save to or load from file

The file name with extension `.dfl` to load or to save the settings to.

For details, see the Save to file function in the "Settings" tab.

Enable user-defined preset

If enabled, the settings from the selected preset file are restored when the [PRESET] key is pressed.

If disabled, [PRESET] sets the instrument to the factory defaults.

Factory defaults

Resets the instrument to the factory default settings, to the initial state. Factory settings comprise all instrument settings, including display, intensity and transparency settings. After loading factory defaults, perform a self-alignment to synchronize the signal data ("File" menu > "Selfalignment").

Remote command:

`SYSTem:PRESet` on page 1049

11.6.2 Restoring Settings

When you have changed many different settings on the instrument and are no longer sure which settings are causing which effect in the measurement, you may want to restore the default settings and start anew. The following methods are available:

- Saving instrument settings to a user-defined preset and restoring the instrument settings to user-defined default values
- Restoring all settings on the R&S RTE to the factory-defined values
- Restoring settings from a file (see "To load settings from a saveset file" on page 437)

For details on save/recall instrument settings and associated remote commands, see Chapter 11.1.1, "Savesets", on page 432.

To save a user-defined preset

1. Press the "File" menu > "Preset setup".
2. Enter a name for the preset file. Select the file format.
3. Tap Save.

   Note: If you want to store the file in another directory than the default one, select "Save As". See also: Chapter 11.7, "File Selection Dialog", on page 471

To restore the instrument settings to user-defined default values

1. Press the "File" menu > "Preset setup".
2. Tap "Open" and select the preset file that contains the required settings.

   The instrument settings are restored to values that are stored in the file.
3. To use these settings as preset values, select "Enable user-defined preset".
4. Press the [PRESET] key.

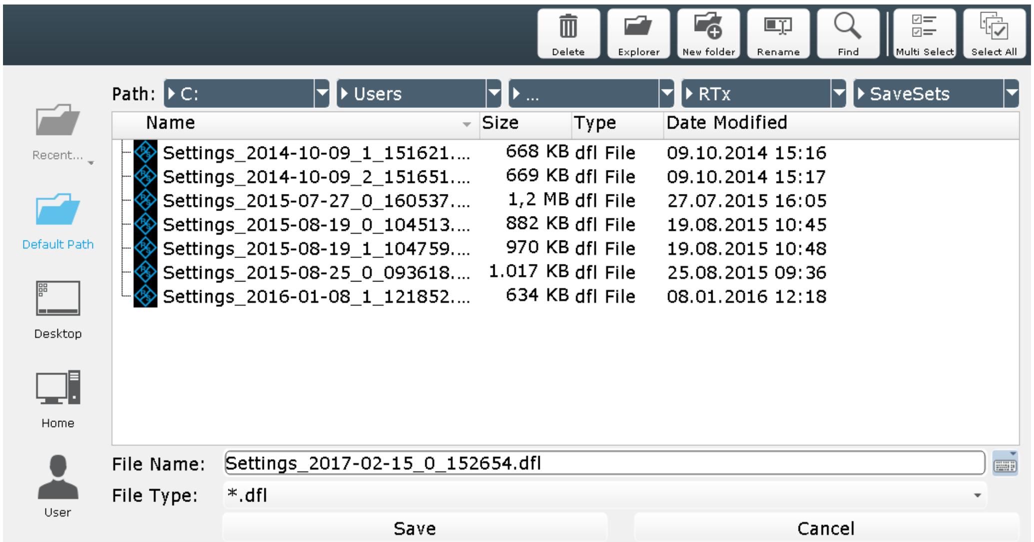
To restore all settings to the factory defaults

1. Press the "File" menu > "Preset setup".
2. Tap the "Factory defaults" button.

   All settings on the R&S RTE are reset to their factory-defined values. As long as no user-defined preset file is loaded and Enable user defined preset is disabled, the [PRESET] key also resets the instrument settings to factory defaults.

11.7 File Selection Dialog

The file selection dialog provides a file explorer from which you can select a file to load or to save data to. You can also manage your files in this dialog.



Path
Tap the path elements to change the current folder. The default folder is defined in Default path for all file operations.

You can save the data in a local folder on the instrument, to an external storage device (usually a USB flash drive), or to a folder on a connected network drive. The path list provides all available drives and folders.

On the left, shortcut icons provide access to often used folders.

Toolbar
The toolbar on the top provides various functions for file management.

File Name
The file name to be loaded or stored to. Double-tap the file name, or tap the keyboard icon to enter the file name using the online keyboard.

The default file name for new files is defined in the "Autonaming" tab, see Chapter 11.3, "Autonaming", on page 459.

File Type
The file extension of the file to be loaded or stored to.

Save, Select
Selects the specified file for the open or save operation and closes the dialog box.

Cancel
Closes the dialog box without selecting a file.

12 Protocol Analysis

Using the serial protocol options for the R&S RTE, you can analyze various serial protocols.

- Basics of Protocol Analysis....473
- I²C (Option R&S RTE-K1)....482
- SPI Bus (Option R&S RTE-K1)....501
- UART/RS-232/RS-422/RS-485 (Option R&S RTE-K2)....513
- CAN and CAN FD (Options R&S RTE-K3 and -K9)....523
- LIN (Option R&S RTE-K3)....561
- FlexRay (Option R&S RTE-K4)....579
- Audio Signals (Option R&S RTE-K5)....598
- MIL-1553 (Option R&S RTE-K6)....620
- ARINC 429 (Option R&S RTE-K7)....641
- Ethernet 10BASE-T and 100BASE-TX (Option R&S RTE-K8)....656
- Ethernet 100BASE-T1 (Option R&S RTE-K57)....675
- SENT (Option R&S RTE-K10)....696
- Custom: Manchester / NRZ (Option R&S RTE-K50)....733
- MDIO (Option R&S RTE-K55)....768
- USB (Option R&S RTE-K60)....783
- USBPD (Option R&S RTE-K63)....815
- SpaceWire (Option R&S RTE-K65)....831
- CXPI (Option R&S RTE-K76)....846

12.1 Basics of Protocol Analysis

Before you can analyze a serial signal, the bus has to be configured according to the protocol and specifics of the signal. The configuration contains:

- Assignment of the data and clock lines to the input channels
- Logical thresholds
- Protocol-specific settings

Serial data can be analyzed in several ways:

- Triggering: You can trigger on various events that are typical for the selected protocol type, for example, on start and stop of messages, or on specified data patterns in the message.
  Triggering on a trigger event sequence is not supported, and holdoff settings are not available.
- Protocol decoding: The digitized signal data is displayed on the screen together with the decoded content of the messages in readable form, and the decode results are listed in a table.
- Search on decoded signal data: For most serial protocols, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, because several event types can be combined. Thus, you get the results for the complete acquisition cycle.

12.1.1 Decode - General Settings

For all protocols, configuration starts with the selection of the serial bus and the protocol. The "Trigger Setup" button leads directly to the trigger configuration.

Table 12-1: Configuration settings are protocol-specific. They are described in the related chapters:

ARINC 429	Chapter 12.10.2, "ARINC 429 Configuration", on page 642
Audio	Chapter 12.8.2.1, "Audio Signal Configuration", on page 600
CAN, CAN FD	Chapter 12.5.1, "CAN and CAN-FD Configuration", on page 523
Custom: Manchester / NRZ	Chapter 12.14.2, "Custom: Manchester / NRZ Configuration", on page 735
CXPI	Chapter 12.19.2, "CXPI Configuration", on page 847
Ethernet 10BASE-T and 100BASE-TX	Chapter 12.11.2, "Ethernet Configuration", on page 657
Ethernet 100BASE-T1	Chapter 12.12.2, "100BASE-T1 Configuration", on page 677
FlexRay	Chapter 12.7.1.1, "FlexRay Configuration", on page 579
I²C	Chapter 12.2.2, "I²C Configuration", on page 484
LIN	Chapter 12.6.2, "LIN Configuration", on page 563
MDIO	Chapter 12.15.2, "MDIO Configuration", on page 770
MIL-1553	Chapter 12.9.2, "MIL-STD-1553 Configuration", on page 623
SENT	Chapter 12.13.2.1, "SENT Configuration", on page 700
SpaceWire	Chapter 12.18.2, "SpaceWire Configuration", on page 833
SPI	Chapter 12.3.2, "SPI Configuration", on page 502
UART	Chapter 12.4.2.1, "UART Configuration Settings", on page 514
USB	Chapter 12.16.2, "USB 2.0 Configuration", on page 789
USB PD	Chapter 12.17.2, "USBPD Configuration", on page 817

Make sure that the tab of the correct serial bus is selected on the left side.

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1382

Decode
Enables the decoding of the selected bus. The signal icon of the bus appears on the signal bar.

Remote command:
`BUS<m>[:STATe]` on page 1382

12.1.2 Full Autoset

If you want to perform a quick protocol measurement, you can use the "Full Autoset" function. It allows you to find the correct instrument settings for the desired protocol. It displays at least one decoded frame after the execution.

Performing a full autoset

1. Press [PROTOCOL] on the front panel.
2. Select the bus you want to configure.
3. Select the "Decode" tab.
4. Tap the "Protocol" you want to configure.
5. Assign the sources for the input channels.
6. Press the "Full autoset" button.

"Full autoset" performs the following steps:
- Executes "Autoset" for the horizontal and vertical scale.
- Adjusts the horizontal scale to display at least one frame or packet .
- Executes "Auto thresholds" to determine the thresholds.
- If necessary, performs bit rate estimation.
- Sets default protocol trigger (frame start) .
- Turns on the protocol decoder.

Remote command:

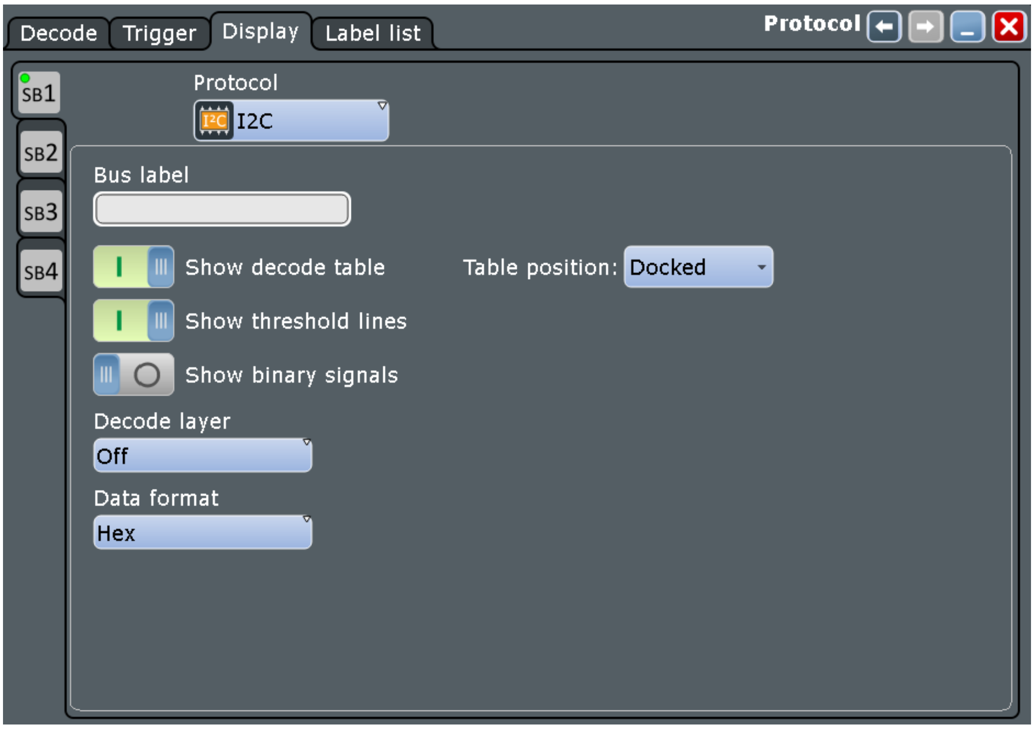
`BUS<m>:FAUToset` on page 1383

There are some limitations for the full autoset:

- It is not available, if the digital MSO channels are selected as a source for the waveform.
- Signals with longer idle phases might not find horizontal adjustment.
- On duplex signals, different forward/reverse amplitudes can fail to determine thresholds.

12.1.3 Display

For all protocols, you can select to display the decoded signal as a table and to show the binary signal on the screen. Optionally, you can assign a label to the bus.



For some protocols, the result table provides a button to show the details of the selected frame.

Bus label

Defines a label to be displayed with the bus.

Remote command:

BUS<m>:LABel on page 1383

Show decode table

Opens a table with decoded data of the serial signal. The function requires the option for the analyzed protocol.

Table 12-2: Decode results are protocol-specific. They are described in the related chapters:

ARINC 429	Chapter 12.10.5, "ARINC 429 Decode Results", on page 650
Audio	Chapter 12.8.4, "Audio Decode Results", on page 610
CAN, CAN FD	Chapter 12.5.4, "CAN and CAN FD Decode Results", on page 542
Custom: Manchester / NRZ	Chapter 12.14.5, "Custom: Manchester / NRZ Decode Results", on page 762
CXPI	Chapter 12.19.5, "CXPI Decode Results", on page 863
Ethernet 10BASE-T and 100BASE-TX	Chapter 12.11.5, "Ethernet Decode Results", on page 668
Ethernet 100BASE-T1	Chapter 12.12.5, "100BASE-T1 Decode Results", on page 687
FlexRay	Chapter 12.7.4, "FlexRay Decode Results", on page 590
I²C	Chapter 12.2.5, "I²C Decode Results", on page 493

LIN	Chapter 12.6.5, "LIN Decode Results", on page 572
MDIO	Chapter 12.15.5, "MDIO Decode Results", on page 777
MIL-1553	Chapter 12.9.5, "MIL-STD-1553 Decode Results", on page 633
SENT	Chapter 12.13.5, "SENT Decode Results", on page 722
SpaceWire	Chapter 12.18.4, "SpaceWire Decode Results", on page 839
SPI	Chapter 12.3.4, "SPI Decode Results", on page 508
UART	Chapter 12.4.4, "UART Decode Results", on page 520
USB	Chapter 12.16.4, "USB 2.0 Decode Results", on page 803

Remote command:
`BUS<m>:RESult` on page 1383

Table position
Defines the position of the decode table on the screen.

"Floating" Floating result box in front of the diagrams.

"Preview"" Result icon on the sidebar.

"Docked" Fixed tab below the diagrams.

Remote command:
`DISPlay:RESultboxes:DEPosition` on page 1385

Show binary signals
For each configured line, the binary signal is also displayed.

Show threshold lines
If selected, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1383

Data format
Sets the data format for decoded data values of the selected bus in the "Decode results" box and in the combs of the decoded signal.

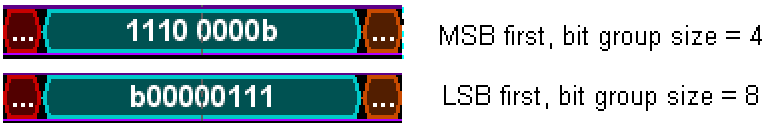
Remote command:
`BUS<m>:FORMat` on page 1384

Binary bit order
Select MSB or LSB to define the data bit order in the combs of the decoded signal. The setting is only available for the binary data format, and only for protocols sending data LSB first.

If the "Binary bit order" is LSB, you can read the bits of an LSB first signal in LSB first order in the combs, while the results table displays the correct values MSB first.

Binary bit group size

Sets the number of bits that forms a bit group in the comb display of the selected bus. The setting is only available for the binary data format, and only for protocols sending data LSB first.



Result export

Opens the "File" dialog box. Select the "Waveforms/Results" tab > "Numeric" subtab to save the decode results to file.

Show details

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

Remote command:

`BUS<m>:RESDetail` on page 1384

12.1.4 Label Lists

For all protocols using ID or address identification, it is possible to create label lists containing addresses or IDs, a symbolic name for each node (symbolic label), and some protocol-specific information.

You can load label lists, and activate its usage for decoding. As a result, an additional "Label" column appears in the "Decode results" table, containing the symbolic label. The frame captions of the decoded signal show the symbolic label instead of the ID or address values. Hence it is easy to identify the messages of the different bus nodes.

You can also use the label list to trigger on an identifier or address. Instead of entering the value, you select the name, which is defined in the label list.

Available file formats are PTT, CSV, DBC (CAN only), and XML (FIBEX files, FlexRay only).

Label lists are protocol-specific. Their contents are described in the corresponding protocol chapters:

- Chapter 12.2.4, "I^2C Label List", on page 492
- Chapter 12.5.3, "CAN / CAN FD Label List", on page 540
- Chapter 12.6.4, "LIN Label List", on page 569
- Chapter 12.7.3, "FlexRay Label List", on page 588
- Chapter 12.10.4, "ARINC 429 Label List", on page 649
- Chapter 12.9.4, "MIL-STD-1553 Label List", on page 632

12.1.4.1 Content and Format of the PTT File

Label lists are stored as PTT (protocol translation table) files. The PTT file format is an extension of the CSV format (comma-separated values). You can edit it with standard editors, for example, with MS Excel or a text editor.

The PTT file has three types of lines:

- Comment lines begin with a hash character #. A hash character at any other position in the line is treated like a standard character.
- Command lines begin with a commercial at character @. An @ character at any other position in the line is treated like a standard character.
- Standard lines are the lines that not qualify as comment or command lines. They build the core of the label list.

Command lines

Command lines define the version of the PTT file and the protocol name:

- @FILE_VERSION: must appear exactly once in the file
- @PROTOCOL_NAME: must appear at least once in the file. Thus, one file can contain several label lists for different protocols.

```
# --- Start of PTT file
@FILE_VERSION  = 1.0
@PROTOCOL_NAME = i2c
[... Label list for I2C]
@PROTOCOL_NAME = can
[... Label list for CAN]
# --- End of PTT file
```

Standard lines

Standard lines define the contents of the label list. The rules for standard lines follow the csv convention, they are:

- Values are separated by commas
- Space characters following a delimiter are ignored
- Values with a special character (comma, newline, or double quote) must be enclosed in double quotes
- Text in double quotes must be escaped by double quote characters

The format of the numeric value is indicated by a suffix. The following formats are supported:

Format	Suffix	Example
Decimal	<empty> d	`106, DeviceName` `106d, DeviceName`
Hexadecimal	h	`6Ah, DeviceName` or prefix: `0x6A, DeviceName`

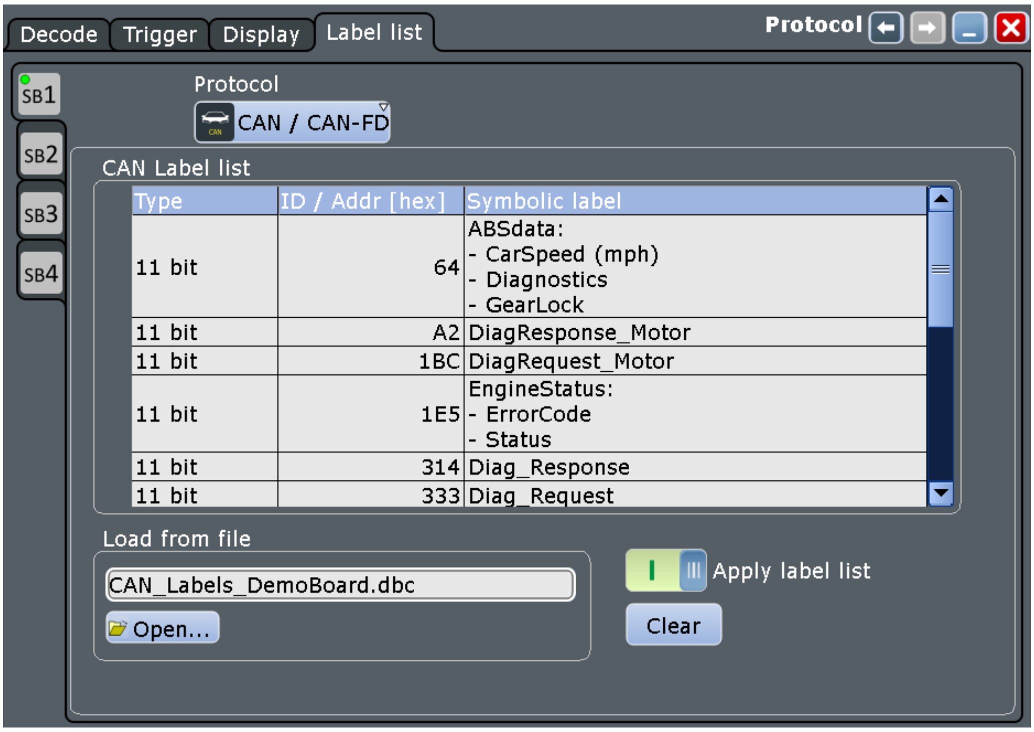
Format	Suffix	Example
Octal	o	`152o, DeviceName`
Binary	b	`01101010b, DeviceName`

The maximum supported word size for (unsigned) integers is 64 bits.

```
# --- Start of PTT file
@FILE_VERSION  = 1.0
@PROTOCOL_NAME = i2c
#    Following two lines are equal:
7,01h,Temperature
7,01h, Temperature
#    A comma must be enclosed in double quotes:
7,01h,"Temperature, Pressure, and Volume"
#    A double quote must also be enclosed in double quotes:
7,7Fh,"Highspeed ""Master"" 01"
#    Following lines yield the same result:
7d,0x11,Pressure
7h,11h,Pressure
0x7,17d,Pressure
7,17,Pressure
```

12.1.4.2 Label List - General Settings

In the "Label List" tab, you can load, read and activate label list files.



The common settings for all protocols are:

Save to or load from file
Selects and loads a label list file. Available file formats are PTT, CSV, DBC (CAN only), and XML (FIBEX files, FlexRay only).

Label lists are protocol-specific. Their contents are described in the corresponding protocol chapters:

- Chapter 12.2.4, "I^{2}C Label List", on page 492
- Chapter 12.5.3, "CAN / CAN FD Label List", on page 540
- Chapter 12.6.4, "LIN Label List", on page 569
- Chapter 12.7.3, "FlexRay Label List", on page 588
- Chapter 12.10.4, "ARINC 429 Label List", on page 649
- Chapter 12.9.4, "MIL-STD-1553 Label List", on page 632

Remote command:
BUS<m>:NEWList on page 1384

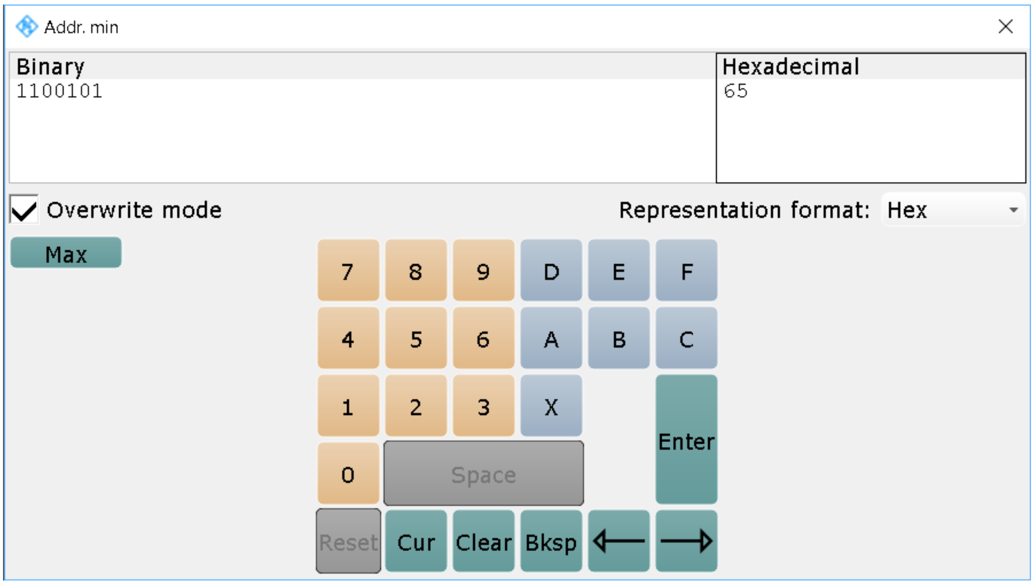
Clear
Deletes the label list from the instrument.

Apply label list
Activates the label list to be used for decoding. The "Label" appear in the "Decode results" table and in the frame captions of the decoded signal.

Remote command:
BUS<m>:SYMBols on page 1385

12.1.5 Bit Pattern Editor

If you want to enter a specified address or data pattern, the bit pattern editor helps you to enter the pattern in various formats - decimal, hexadecimal, octal, binary and ASCII.



The editor displays the pattern in two columns. The left column always shows binary data. For the right column, you can select the format, the default depends on the data specifics. You can edit data in the left or right column. The keypad adapts itself to the column format, only keys appropriate to the format are enabled.

The data is grouped and converted in bit groups. The size of a bit group depends on the address or data specifics and is set by the instrument. Groups are automatically separated by blanks. The maximum size of a bit group is 64 bit, the most common group size is 1 byte.

"Overwrite mode": If disabled, the data behind the new digit is shifted to the right. Bit groups are rearranged automatically.

Format-specific information:

- Unsigned: Decimal data format without sign. It is available for I²C, SPI, UART, CAN, LIN and FlexRay protocols. If you enter a decimal number that is too large for the defined bit group, the number is truncated and a message appears. X (do not care) in the decimal column sets all binary digits of the bit group to X.
- Signed: Signed decimal format, available for audio protocols. The first bit represents the sign. You can use the 2's complement or 1's complement format.
- Binary: 0, 1 and X (do not care) is allowed.
- Hex: most common format in the right column.
- Octal: Each digit represents 3 bit.
- ASCII: In the ASCII column, "X" is the character X. The binary X (do not care) is not allowed. If an X is included in the binary value in the left column, the ASCII column displays "§" to indicate that the value is not defined.

Where applicable, frequently used values are provided in a "Predefined values" list below the pattern table, for example, reserved end words of data packets in the UART protocol.

12.2 I²C (Option R&S RTE-K1)

The Inter-Integrated Circuit is a simple, low-bandwidth, low-speed protocol used for communication between on-board devices.

- The I²C Protocol....482
- I²C Configuration....484
- I²C Trigger....487
- I²C Label List....492
- I²C Decode Results....493
- Search on Decoded I²C Data....496

12.2.1 The I²C Protocol

This chapter provides an overview of protocol characteristics, data format, address types and trigger possibilities. For detailed information, read the "I2C-bus specification and user manual" available on the NXP manuals web page at http://www.nxp.com/.

I²C characteristics

Main characteristics of I²C are:

- Two-wire design: serial clock (SCL) and serial data (SDA) lines
- Master-slave communication: the master generates the clock and addresses the slaves. Slaves receive the address and the clock. Both master and slaves can transmit and receive data.
- Addressing scheme: each slave device is addressable by a unique address. Multiple slave devices can be linked together and can be addressed by the same master.
- Read/write bit: specifies if the master reads (=1) or writes (=0) the data.
- Acknowledge: takes place after every byte. The receiver of the address or data sends the acknowledge bit to the transmitter.

The R&S RTE supports all operating speed modes: high-speed, fast mode plus, fast mode, and standard mode.

Data transfer

The format of a simple I²C message (frame) with 7-bit addressing consists of the following parts:

- Start condition: a falling slope on SDA while SCL is high
- 7-bit address of the slave device that either is written to or read from
- R/W bit: specifies if the data is written to or read from the slave
- ACKnowledge bits: is issued by the receiver of the previous byte if the transfer was successful
  Exception: At read access, the master terminates the data transmission with a NACK bit after the last byte.
- Data: several data bytes with an ACK bit after every byte
- Stop condition: a rising slope on SDA while SCL is high

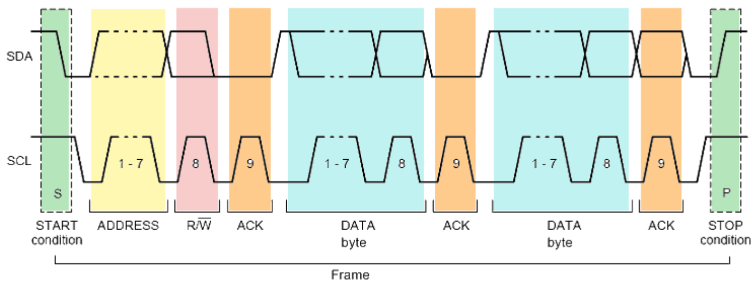


Figure 12-1: I2C writes access with 7-bit address

Address types: 7-bit and 10-bit

Slave addresses can be 7 bits or 10 bits long. A 7-bit address requires 1 byte, 7 bits for the address followed by the R/W bit.

A 10-bit address for write access requires 2 bytes: the first byte starts with the reserved sequence 11110, followed by the two MSB of the address and the write bit. The second byte contains the remaining 8 LSB of the address. The slave acknowledges each address byte.

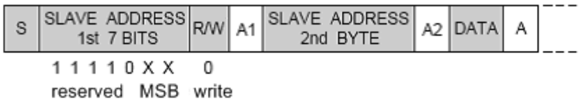


Figure 12-2: 10-bit address, write access

A 10-bit address for read access requires 3 bytes. The first 2 bytes are identical to the write access address. The third byte repeats the address bits of the first byte and sets the read bit.

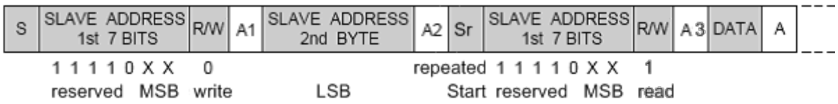


Figure 12-3: 10-bit address, read access

Trigger

The R&S RTE can trigger on various parts of I²C messages. The data and clock lines must be connected to the input channels, triggering on math and reference waveforms is not possible.

You can trigger on:

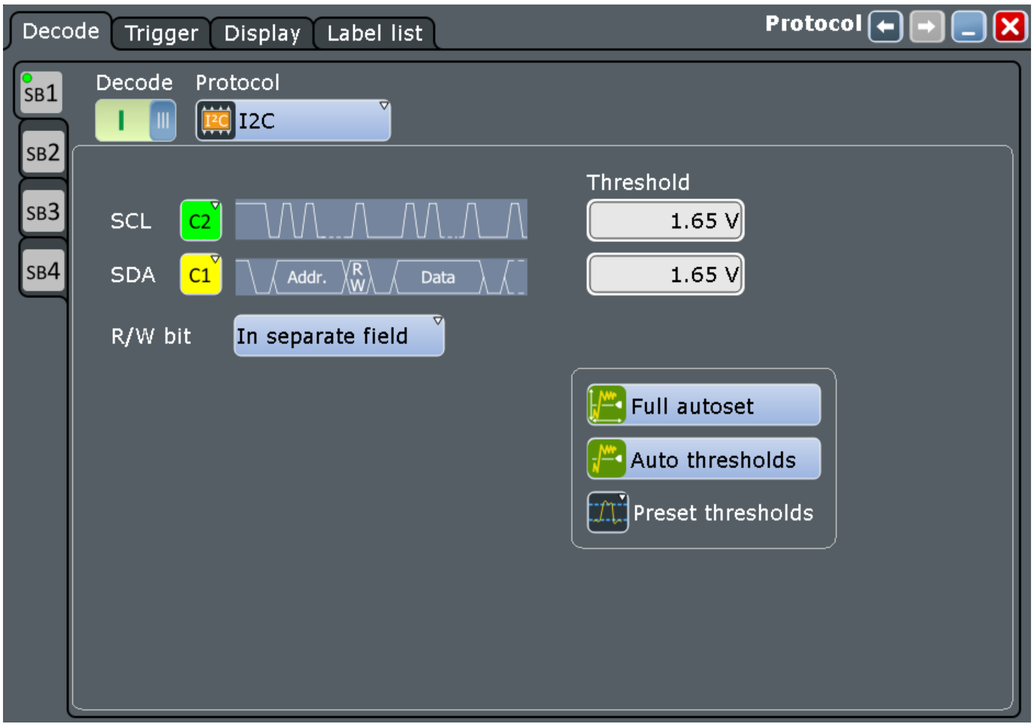
- Start or stop condition
- Repeated start condition
- Transfer direction (read or write)
- Bytes with missing acknowledge bit
- Specific slave address or address range
- Specific data pattern in the message

12.2.2 I²C Configuration

12.2.2.1 I²C Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *I2C*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

SDA, SCL

Set the waveforms of the data line (SDA) and clock line (SCL).

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:I2C:SDA:SOURce` on page 1388
`BUS<m>:I2C:SCL:SOURce` on page 1388

Threshold

Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the threshold.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Full autoset"

Starts software algorithms for determining the signal threshold levels and bitrate.
See also Chapter 12.1.2, "Full Autoset", on page 475.

- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Preset thresholds"
  Selects the default threshold voltage for various signal technologies from a list. The value is set to "Manual" if the threshold was set with "Auto threshold", or was entered directly.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:I2C:SCL:THReshold` on page 1389
`BUS<m>:I2C:SDA:THReshold` on page 1389
`BUS<m>:I2C:TECHnology` on page 1389
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

R/W bit
Defines if the R/W bit is considered separately or as part of the address. The setting affects the Address setup of the trigger conditions.

Remote command:
`BUS<m>:I2C:RWBit` on page 1390

12.2.2.2 Configuring I²C Protocol

The configuration of the I²C is simple - assign the two lines to input channels, and set the thresholds.

For details on configuration settings, see Chapter 12.2.2, "I²C Configuration", on page 484.

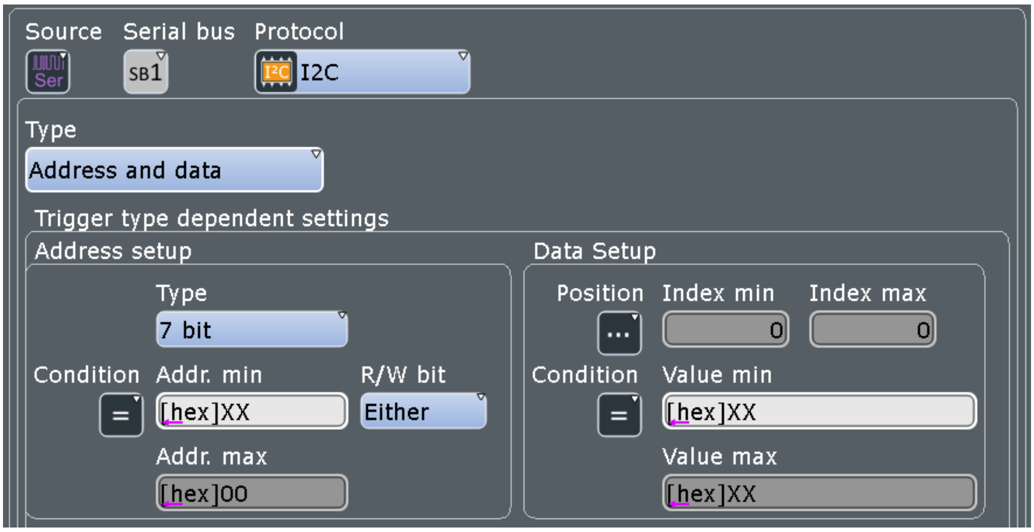
1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "I2C".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Tap the "SDA" button, and select the waveform of the data line.
7. Tap the "SCL" button, and select the waveform of the clock line.
8. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to an automatic value with "Full autoset"/ "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.

9. Enable "Decode", if available.

12.2.3 I²C Trigger

12.2.3.1 I²C Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = I2C"



Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:
TRIGger<m>:SOURce:SBSelect on page 1387

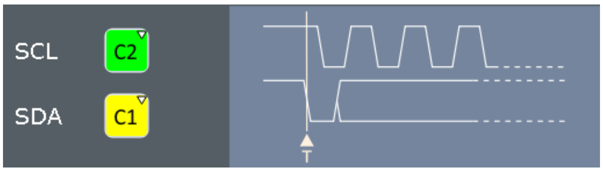
Protocol
Defines the protocol type of the selected serial bus.

Remote command:
BUS<m>:TYPE on page 1382

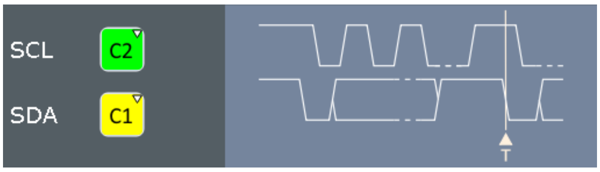
Trigger type

Selects the trigger type for I²C analysis.

"Start" Sets the trigger to the start of the message. The start condition is a falling edge on SDA while SCL is high. The trigger instant is the falling edge of the SDA line.
You can change the SDA and SCL lines here if necessary.



"Repeated start" Sets the trigger to a repeated start - when the start condition occurs without previous stop condition. Repeated start conditions occur when a master exchanges multiple messages with a slave without releasing the bus.



"Stop" Sets the trigger to the end of the message. The stop condition is a rising slope on SDA while SCL is high.

"No Ack (Missing Ack)" Missing acknowledge bit: the instrument triggers if the data line remains HIGH during the clock pulse following a transmitted byte.
You can also localize specific missing acknowledge bits by setting the No Ack conditions.

"Address" Sets the trigger to one specific address condition or a combination of address conditions. The trigger time is the falling clock edge of the acknowledge bit after the address.

- Address type
- Specified address or address range
- Read/Write bit

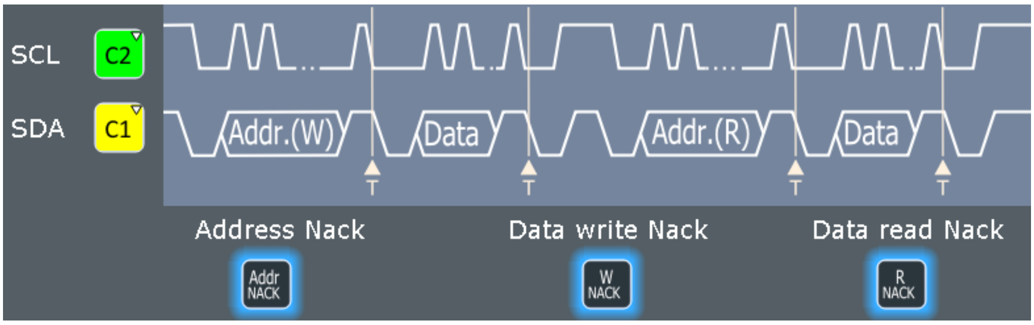
Description of trigger type specific settings: "Address setup" on page 489.

"Address OR" Triggers on one to four address conditions.
Description of trigger type specific settings: "Address OR conditions" on page 490.

"Address and data" Sets the trigger to a combination of address and data condition. The address conditions are the same as for the "Address" trigger type, see "Address setup" on page 489 and "Data setup" on page 491.

Remote command:
`TRIGger<m>:I2C:MODE` on page 1391

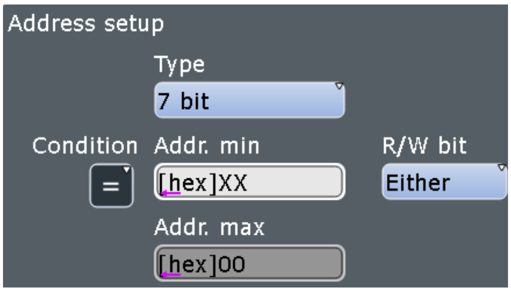
No Ack conditions
Selects which missing acknowledge bits is detected if the trigger type is set to "Missing Ack".



"Address Nack"	No slave recognizes the address.
"Data write Nack"	The addressed slave does not accept the data.
"Data read Nack"	Marks the end of the read process when the master reads data from the slave. This Nack is sent according to the protocol definition, it is not an error.

Remote command:
`TRIGger<m>:I2C:ADNack` on page 1392
`TRIGger<m>:I2C:DWNack` on page 1392
`TRIGger<m>:I2C:DRNack` on page 1392

Address setup
Specifies the address conditions:



Type ← Address setup
Sets the address length to be triggered on: 7 bit, 7+1 bit, or 10 bit. Available settings depend on the R/W bit setting of the bus configuration.

For "7 bit" and "10 bit", enter the address bits in the Addr. min / Addr. max field, and use the "R/W bit" on page 490 field to select the transfer direction.

For "7+1 bit", enter the seven address bits and also the R/W bit in the "Address" field.

If the trigger type is "Address + data", you can set the address type "Any" to trigger on data only, regardless of the address.

Remote command:
TRIGger<m>:I2C:AMODe on page 1392

Addr. min / Addr. max ← Address setup
Defines the bit pattern of the slave device address. The length of the entry is adjusted to the selected address type. In binary format, use the following characters: 1; 0; or X (any bit). The use of X is restricted to the "Address operator"s "Equal" and "Not equal".

The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Depending on the Condition, a specific address or an address range must be defined.

To trigger on any address, set the "Address operator" to "Equal" and enter *X* for each address bit.

Remote command:
TRIGger<m>:I2C:ADDRess on page 1393
TRIGger<m>:I2C:ADDTo on page 1393

Condition ← Address setup
Sets the operator to set a specific address ("Equal" or "Not equal") or an address range. The address values are set with Addr. min / Addr. max.

Remote command:
TRIGger<m>:I2C:ACONdition on page 1393

R/W bit ← Address setup
Toggles the trigger condition between Read and Write access of the master. Select "Either" if the transfer direction is not relevant for the trigger condition.

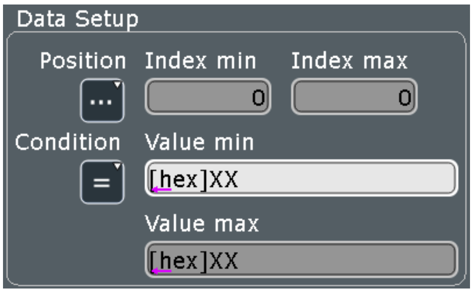
Remote command:
TRIGger<m>:I2C:ACCess on page 1391

Address OR conditions
Triggers on one to four address conditions. For each condition to be used, select "Monitor".

Each condition requires an exact address. The definition of address ranges is not possible here. X (do not care) can be used.

Remote command:
TRIGger<m>:I2C:ADOR<n>:ENABle on page 1393
TRIGger<m>:I2C:ADOR<n>:ADRType on page 1394
TRIGger<m>:I2C:ADOR<n>[:VALue] on page 1394
TRIGger<m>:I2C:ADOR<n>:RWBit on page 1394

Data setup
Specifies the data conditions:



Position ← Data setup
Operator for the data position within a frame. You can define an exact position, or a position range. Select "Any", if the position of the required pattern is not relevant.

Remote command:
TRIGger<m>:I2C:DPOPerator on page 1395

Index min, Index max ← Data setup
Sets the number of data bytes to be skipped after the address. The index 0 is associated with the first data byte. If the Position defines a range, the first and the last byte of interest are defined.

Remote command:
TRIGger<m>:I2C:DPOSition on page 1395
TRIGger<m>:I2C:DPTO on page 1395

Condition ← Data setup
Selects the operator for the "Data" pattern: "Equal", "Not equal", or a range definition.

Remote command:
TRIGger<m>:I2C:DCONdition on page 1395

Value min / Value max ← Data setup
Specifies the data bit pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

The instrument ensures that the max value is always ≥ the min value, and X bits (do not care) are at the same position in both values.

The bit pattern editor helps you to enter the pattern, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Remote command:
TRIGger<m>:I2C:DMIN on page 1396
TRIGger<m>:I2C:DMAX on page 1396

12.2.3.2 Triggering on I²C Signals

Prerequisites: An I²C bus is configured, see Chapter 12.2.2.2, "Configuring I²C Protocol", on page 486.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to I²C.
5. Select the "Trigger type".
6. For more complex trigger types, enter the address and/or data conditions: address, acknowledge bits, R/W bit, and data pattern.
   For details, see Chapter 12.2.3, "I²C Trigger", on page 487

12.2.4 I²C Label List

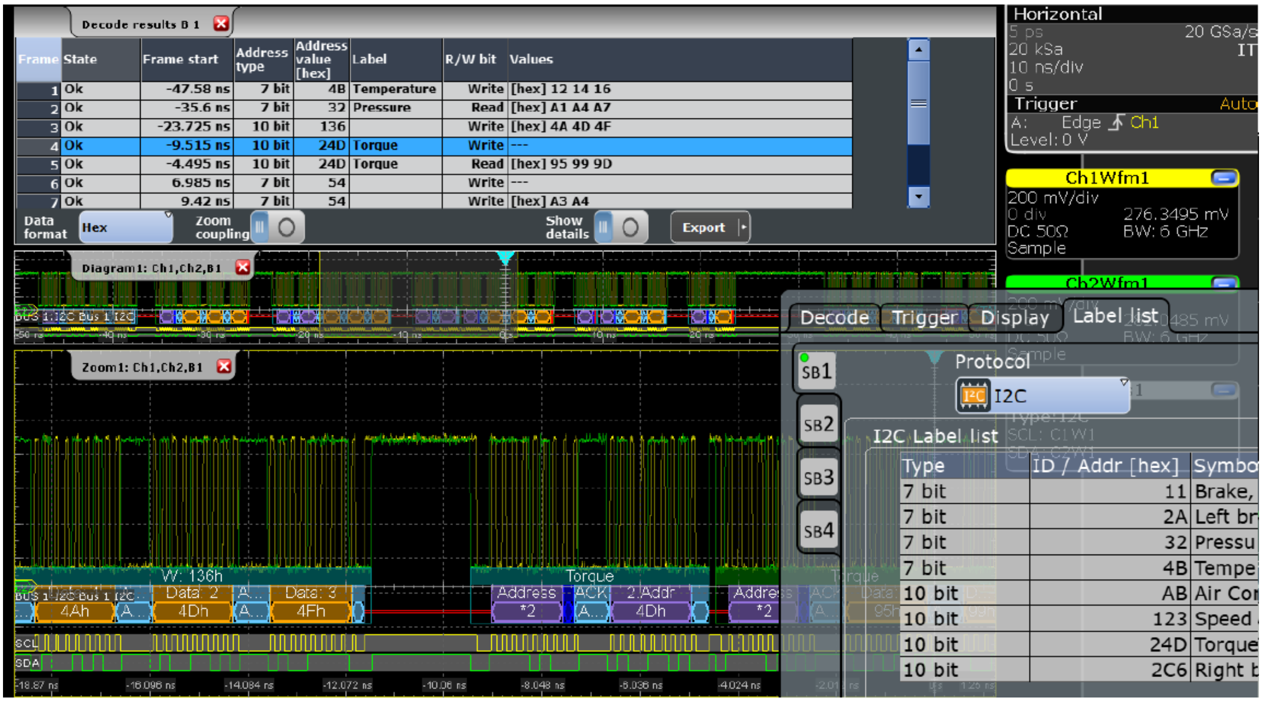
Label lists are protocol-specific. Label lists for I²C are available in CSV and PTT format.

An I²C label file contains three values for each address:

- Address type, 7-bit or 10-bit long
- Address value
- Symbolic label: name of the address, specifying its function in the bus network.

Example: I²C PTT file

```
# ----------------------------------------------------------------------------
@FILE_VERSION  = 1.00
@PROTOCOL_NAME = i2c
# ---------------------------------------------------------------------------
# Labels for I2C protocol
#   Column order: Identifier type, Identifier value, Label
# ---------------------------------------------------------------------------
7,0x1E,Voltage
7,38h,Pressure
7,2Ah,Temperature
7,16h,Speed
7,118,Acceleration
7,07h,HighSpeed_Master_0x3
7,51h,EEPROM
10,3A2h,DeviceSetup
10,1A3h,GatewayStatus
10,06Eh,LeftSensor
# ----------------------------------------------------------------------------
```



For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- `BUS<m>:I2C:FRAMe<n>:SYMBol?` on page 1402

12.2.5 I²C Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

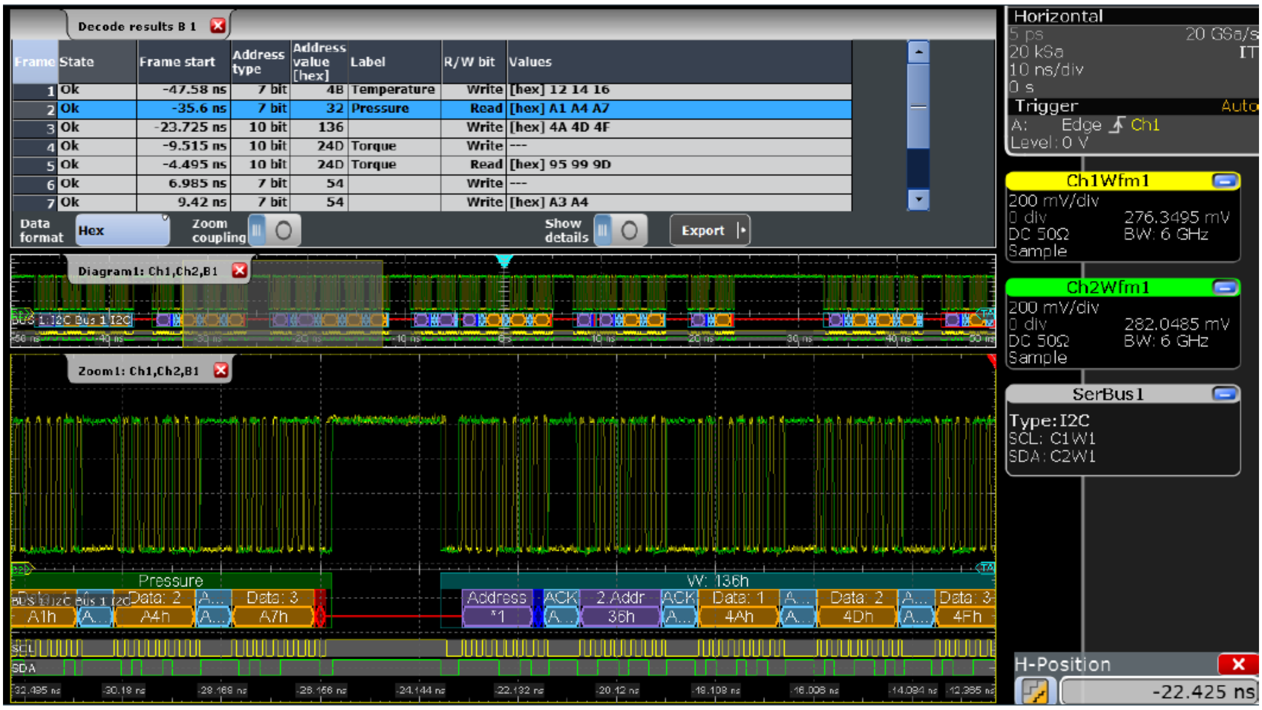


Figure 12-4: Decoded and binary I2C signal, and decode results

green brackets [...]	= start and end of frame
blue frame header	= write frame ok, with transfer direction and address value
green frame header	= read frame ok, with transfer direction and address value
yellow	= address
blue	= correct data
light orange	= R/W bit
purple	= acknowledge bit
red	= No ack (missing acknowledge bit)

The signal in Figure 12-4 shows a write access followed by a read access, both with 10bit address. The decoded data shows a No Ack bit at the end of the read data. This No Ack bit is sent according to the protocol definition and is not an error. Thus, the decode results in the table indicate "Ack" for the second frame.

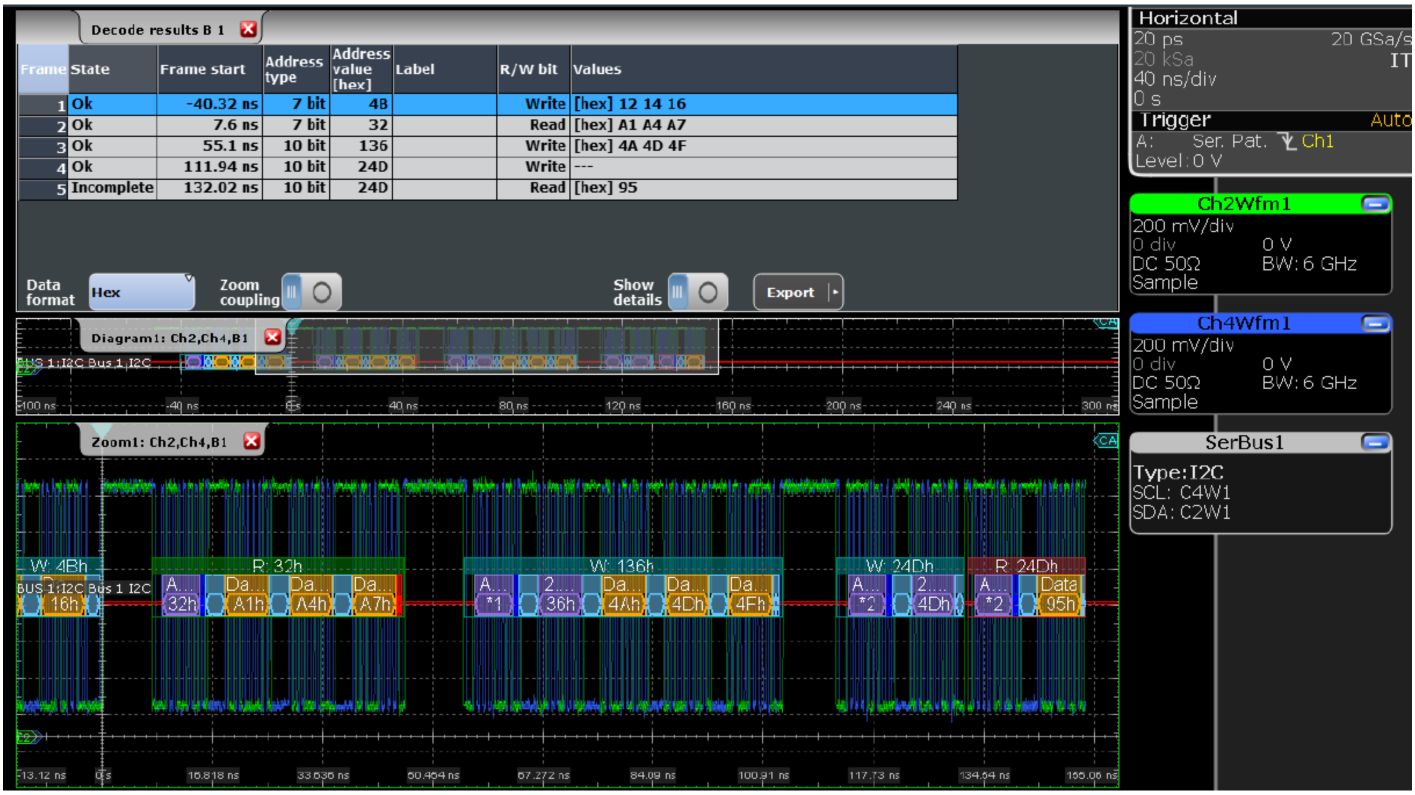


Figure 12-5: Decoded I2C signal with incomplete data, and decode results

magenta frame header	= incomplete frame, missing bits in data words
dark orange frame header	= insufficient frame (end of acquisition before decoding has been completed), with transfer direction and address value
red	= insufficient data word (end of acquisition before end of word)

The "Decode results" box shows the detailed decoded data for each data frame.

Table 12-3: Content of the "Decode results" table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Frame start	Time of frame start
Address type	Address length, 7 bit or 10 bit
Address value (hex)	Hexadecimal value of the address
Label	Symbolic label name defined in the label list
R/W bit	Value of the R/W bit
Ack bit	Value of the address acknowledge bit

Column	Description
Values	Value of all data bytes of the frame. The data format is selected below the table.
Bit rate	Value of the bit rate

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.3.3, "Decode Results", on page 1396.

12.2.6 Search on Decoded I²C Data

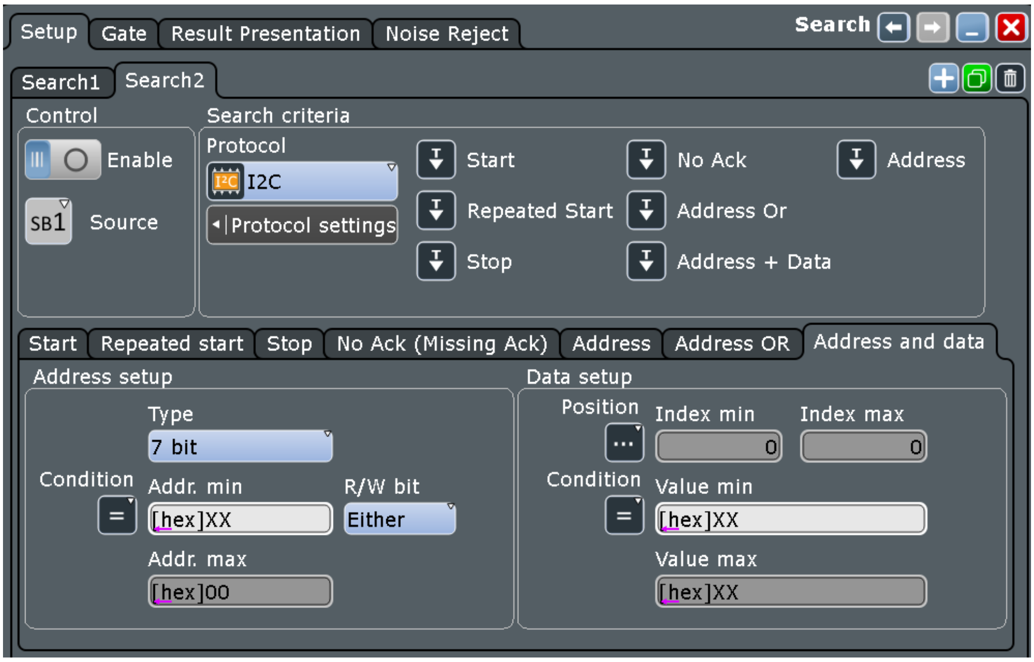
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.2.6.1 I²C Search Setup

Access: [SEARCH] > "Setup" tab



Search criteria

Enable the events to be searched for. Unlike triggering, where you can trigger only on one defined event, you can search for various different events in one search.

"Start" Searches for the start of the message. The start condition is a falling edge on SDA while SCL is high. The event is the falling edge of the SDA line.
You can change the SDA and SCL lines here if necessary.

"Repeated start" Searches for a start condition without previous stop condition. Repeated start conditions occur when a master exchanges multiple messages with a slave without releasing the bus.

"Stop" Searches for the end of the message. The stop condition is a rising slope on SDA while SCL is high.

"No ACK (Missing ACK)" Searches for a missing acknowledge bit: an event is found if the data line remains HIGH during the clock pulse following a transmitted byte. For details, see "No Ack conditions" on page 489.

"Address" Searches for one specific address condition or a combination of address conditions. The event is the falling edge of the acknowledge bit after the address.

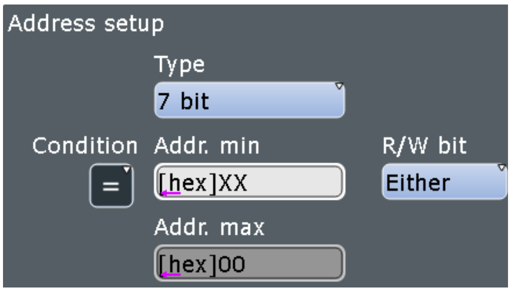
"Address OR" Searches for one to four address conditions.
See "Address OR conditions" on page 490.

"Address and data" Searches for a combination of address and data conditions.

Remote command:
SEARch:TRIGger:I2C:SCONdition on page 1405
SEARch:TRIGger:I2C:RCONdition on page 1405
SEARch:TRIGger:I2C:STCNdition on page 1406
SEARch:TRIGger:I2C:NACKnowledge on page 1406
SEARch:TRIGger:I2C:SADDress on page 1406
SEARch:TRIGger:I2C:ADOR on page 1407
SEARch:TRIGger:I2C:ADData on page 1407

Address setup: Condition, Type, Addr. min, Addr. max, R/W bit
The address setup consists of the condition, type, R/W bit and one or two address patterns.

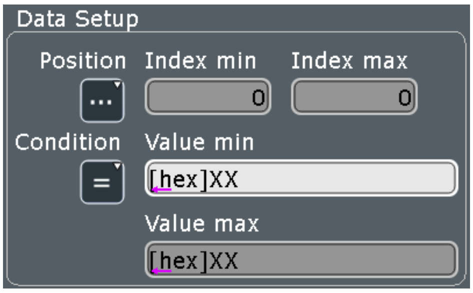
The address setup settings are the same as in the I2C trigger setup, see "Address setup" on page 489.



Remote command:
SEARch:TRIGger:I2C:ACONdition on page 1407
SEARch:TRIGger:I2C:ADDRess on page 1408
SEARch:TRIGger:I2C:ADDTo on page 1408
SEARch:TRIGger:I2C:AMODe on page 1408
SEARch:TRIGger:I2C:ACCess on page 1409

Data setup: Condition, Position, Index min, Index max, Value min, Value max
The data setup consists of the condition, position, and one or two index/value patterns.

The data setup settings are the same as in the I2C trigger setup, see "Data setup" on page 491.



Remote command:
SEARch:TRIGger:I2C:DPOPerator on page 1411
SEARch:TRIGger:I2C:DPOSition on page 1411
SEARch:TRIGger:I2C:DPTO on page 1411

SEARch:TRIGger:I2C:DCONdition on page 1412
SEARch:TRIGger:I2C:DMIN on page 1412
SEARch:TRIGger:I2C:DMAX on page 1412

Address OR setup: Monitor, Address type, Address, R/W bit

The address OR setup consists of the monitor, address type, address and an R/W bit.

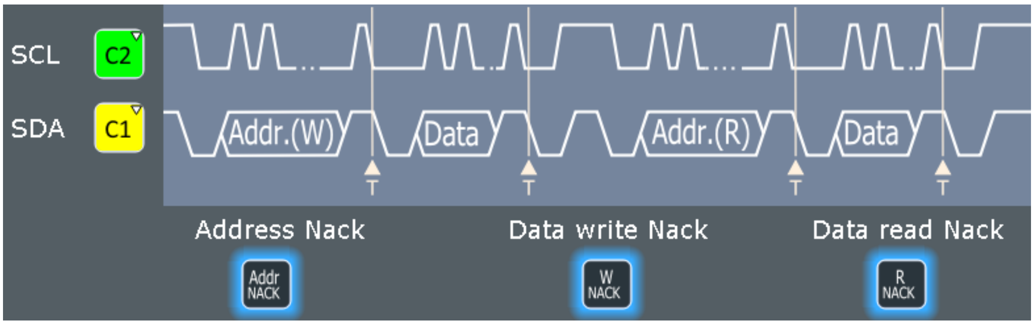
The address OR setup settings are the same as in the I2C trigger setup, see "Address OR conditions" on page 490.

Remote command:
SEARch:TRIGger:I2C:ADDO<m>:ENABle on page 1409
SEARch:TRIGger:I2C:ADDO<m>:ADRType on page 1409
SEARch:TRIGger:I2C:ADDO<m>[:VALue] on page 1410
SEARch:TRIGger:I2C:ADDO<m>:RWBit on page 1410

No ACK setup: Addr/W/R NACK

The no ACK setup consists of the Addr/W/R NACK.

The no ACK setup settings are the same as in the I2C trigger setup, see "No Ack conditions" on page 489.



Remote command:
SEARch:TRIGger:I2C:DRNack on page 1413
SEARch:TRIGger:I2C:DWNack on page 1413
SEARch:TRIGger:I2C:NACKnowledge on page 1406

12.2.6.2 I²C Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

The columns in the search result table are the same as in the decoding table, see Chapter 12.2.5, "I²C Decode Results", on page 493.

Remote commands:

- SEARch:RESult:I2C:FCOunt? on page 1414
- SEARch:RESult:I2C:FRAMe<m>:STATus? on page 1414
- SEARch:RESult:I2C:FRAMe<m>:STARt? on page 1414
- SEARch:RESult:I2C:FRAMe<m>:STOP? on page 1415
- SEARch:RESult:I2C:FRAMe<m>:SYMBol? on page 1418
- SEARch:RESult:I2C:FRAMe<m>:DATA? on page 1417
- SEARch:RESult:I2C:FRAMe<m>:ADDRess? on page 1416
- SEARch:RESult:I2C:FRAMe<m>:AACCess? on page 1415
- SEARch:RESult:I2C:FRAMe<m>:ACCess? on page 1415
- SEARch:RESult:I2C:FRAMe<m>:ACOMplete? on page 1416
- SEARch:RESult:I2C:FRAMe<m>:ADBStart? on page 1416
- SEARch:RESult:I2C:FRAMe<m>:ADEVice? on page 1416
- SEARch:RESult:I2C:FRAMe<m>:AMODe? on page 1417
- SEARch:RESult:I2C:FRAMe<m>:ASTart? on page 1417
- SEARch:RESult:I2C:FRAMe<m>:BCOunt? on page 1418
- SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess? on page 1418
- SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart? on page 1419
- SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete? on page 1419
- SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt? on page 1419
- SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue? on page 1420

12.3 SPI Bus (Option R&S RTE-K1)

- The SPI Protocol....501
- SPI Configuration....502
- SPI Trigger....505
- SPI Decode Results....508
- Search on Decoded SPI Data....511

12.3.1 The SPI Protocol

A 4-channel instrument is required for full support of the SPI protocol, or the MSO option R&S RTE-B1.

The Serial Peripheral Interface SPI is used for communication with slow peripheral devices, in particular, for transmission of data streams.

Main characteristics of SPI are:

- Master-slave communication
- No device addressing; The slave is accessed by a chip select, or slave select line.
- No acknowledgement mechanism to confirm receipt of data
- Duplex capability

Most SPI buses have four lines, two data and two control lines:

- Clock line to all slaves (SCLK)
- Slave Select or Chip Select line (SS or CS)
- Master data output, slave data input (MOSI or SDI)
- Master data input, slave data output (MISO or SDO)

When the master generates a clock and selects a slave device, data may be transferred in either or both directions simultaneously.

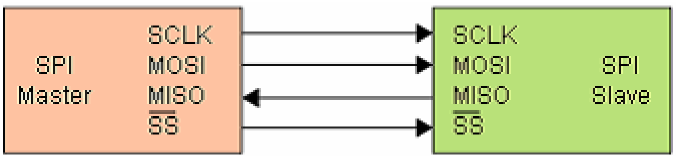


Figure 12-6: Simple configuration of SPI bus

The data bits of a message are grouped by following criteria:

- A word contains a number of successive bits. The word length is defined in the protocol configuration.
- A frame contains a number of successive words, at least one word.

For SPI buses, the R&S RTE provides the following trigger possibilities:

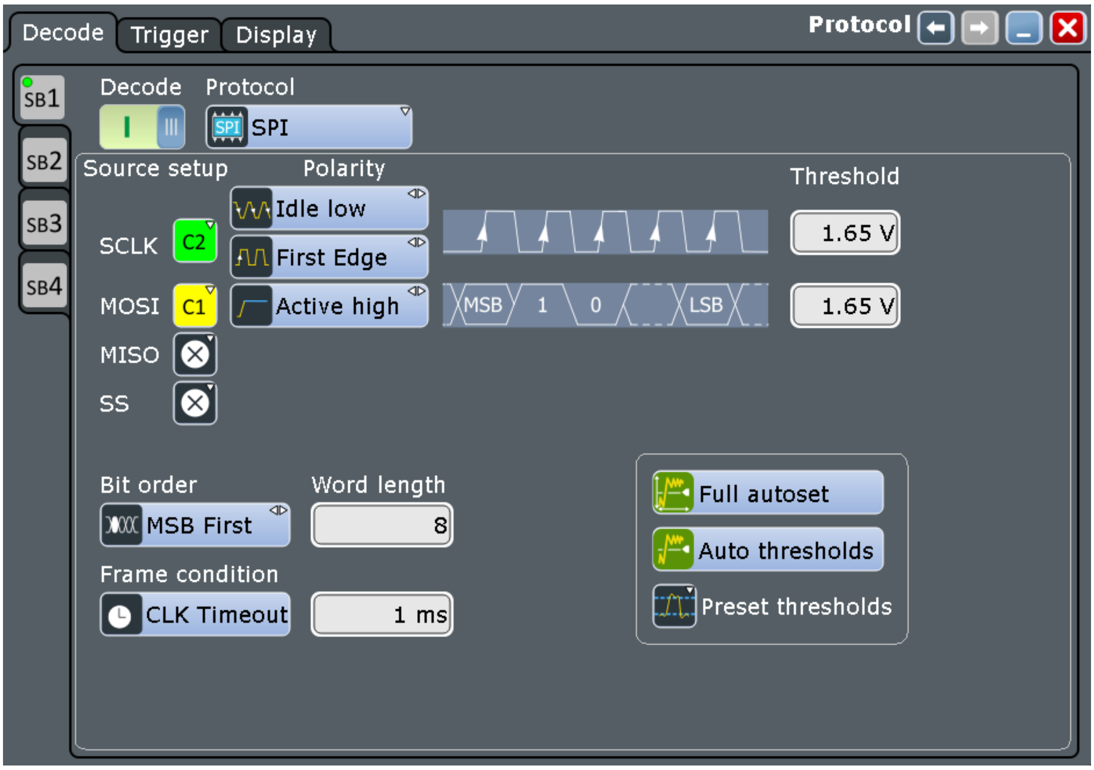
- On frame start
- On a serial pattern at a specified position

12.3.2 SPI Configuration

12.3.2.1 SPI Configuration Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol" = *SPI*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

SCLK
Defines the settings for the clock line.

SCLK source ← SCLK
Sets the input channel of the clock line. Analog channels, math waveforms, and reference waveforms can be used for decoding.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:SPI:SCLK:SOURce` on page 1421

Polarity ← SCLK
Two settings define the clock mode: the clock polarity and the clock phase. Together, they determine the edges of the clock signal on which the data are driven and sampled. A master/slave pair must use the same parameter pair values to communicate.

The clock polarity is "Idle low" (idle = 0) or "Idle high" (idle = 1).

The clock phase defines the slope. It selects if data is stored with the rising or falling slope of the clock. The slope marks the begin of a new bit.

SS, MISO, MOSI
Configures the Slave Select, MISO and MOSI lines.

Source ← SS, MISO, MOSI
Sets the input channel of the selected line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:SPI:SSELect:SOURce` on page 1421
`BUS<m>:SPI:MISO:SOURce` on page 1422
`BUS<m>:SPI:MOSI:SOURce` on page 1423

Polarity ← SS, MISO, MOSI
Selects whether transmitted data or the slave select signal is high active (high = 1) or low active (low = 1).

Remote command:
`BUS<m>:SPI:SSELect:POLarity` on page 1422
`BUS<m>:SPI:MISO:POLarity` on page 1422
`BUS<m>:SPI:MOSI:POLarity` on page 1423

Threshold
Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the Polarity.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Full autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
  See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Auto thresholds"

Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

- "Preset thresholds"
  Selects the default threshold voltage for various signal technologies from a list. The value is set to "Manual" if the threshold was set with "Auto threshold", or was entered directly.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:SPI:SCLK:THReshold` on page 1424
`BUS<m>:SPI:MISO:THReshold` on page 1424
`BUS<m>:SPI:MOSI:THReshold` on page 1424
`BUS<m>:SPI:SSELect:THReshold` on page 1424
`BUS<m>:SPI:TECHnology` on page 1423
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Bit order
Defines if the data of the messages starts with msb (most significant bit) or lsb (least significant bit). The display of the decoded signal considers this setting, results are displayed in the specified order.

Remote command:
`BUS<m>:SPI:BORDer` on page 1420

Word length
Sets the number of bits in a word. The maximum length is 32 bit.

Remote command:
`BUS<m>:SPI:WSIZe` on page 1421

Frame condition
Defines the start of a frame. A frame contains a number of successive words, at least one word.

"SS"	Start and end of the frame is defined by the active state of the slave select signal.
"CLK timeout"	Defines a timeout on the clock line SCLK as limiter between two frames. The timeout condition is used for SPI connections without an SS line. Enter the minimum clock idle time in the field.

Remote command:
`BUS<m>:SPI:FRCondition` on page 1424

Timeout
Sets the minimum clock idle time if a timeout on the clock line SCLK is used as limiter between two frames.

See also: "Frame condition" on page 504.

Remote command:
`BUS<m>:SPI:TIMeout` on page 1425

12.3.2.2 Configuring SPI Signals

For configuration, assign the lines to the input channels, and define the active states and the logical thresholds.

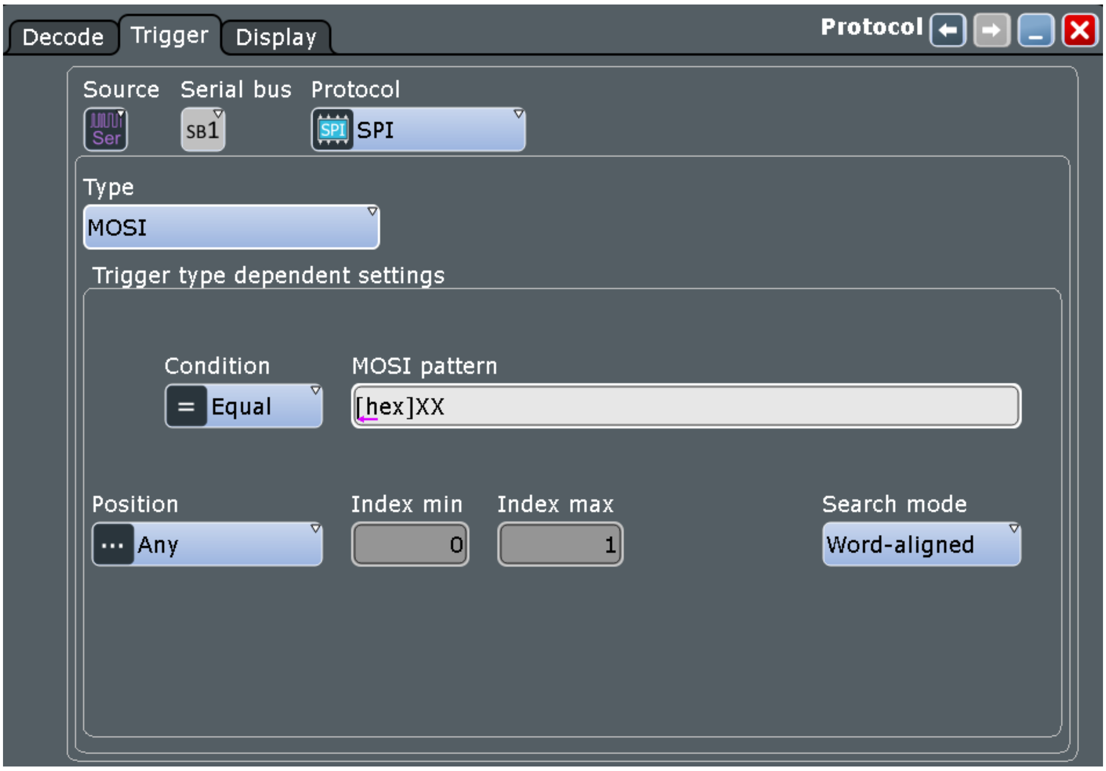
For details on configuration settings, see Chapter 12.3.2, "SPI Configuration", on page 502.

1. Press the [PROTOCOL] key on the front panel.
2. At the left hand-side, select the vertical tab of the bus you want to set up.
3. Select the "Configuration" tab.
4. Tap the "Protocol" button and select the protocol: "SPI".
5. Optionally, you can enter a "Bus label" in the "Display" tab.
6. Tap the "SCLK Source" button, and select the waveform of the clock line.
7. Set the polarity (clock mode) for SCLK.
8. For each of the available SS, MISO and MOSI lines, assign the waveform. Define the polarity (active state) of the line.
9. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to an automatic value with "Full autoset"/ "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.
10. Set the "Bit order", "Word length", and "Frame condition" according to your signal.

12.3.3 SPI Trigger

12.3.3.1 SPI Trigger

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = SPI"



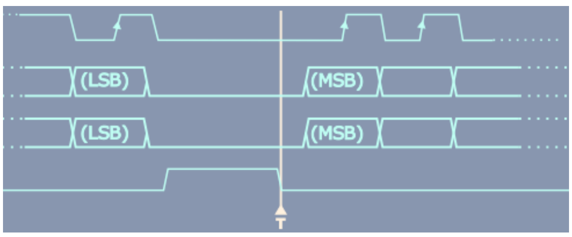
Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

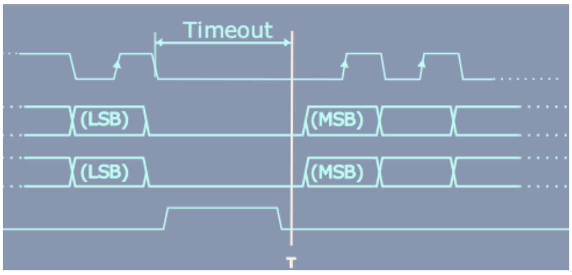
Trigger type

Selects the trigger type for SPI analysis.

"Frame start (SS)" Trigger on the start of the frame when the slave select signal SS changes to the active state. This trigger type is available if Frame condition is set to "SS".



"Frame start (Timeout)"
: Triggers on the start of the frame when the clock idle time exceeds the "Timeout" time. This trigger type is available if Frame condition is set to "CLK timeout".



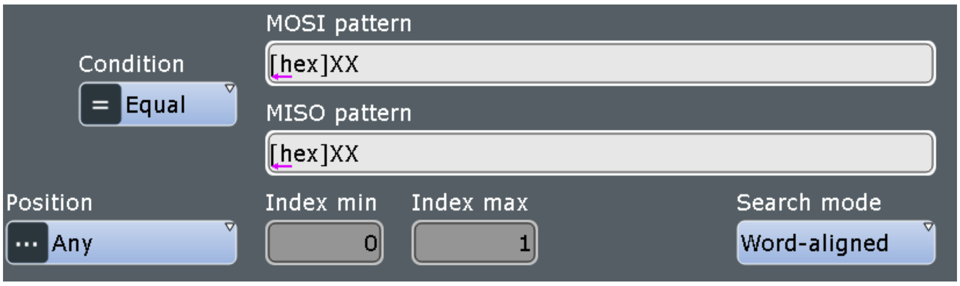
"MOSI"
: Sets the trigger to a specified data pattern expected on the MOSI line.
See: "MOSI and MISO data conditions" on page 507.

"MISO"
: Sets the trigger to a specified data pattern expected on the MISO line.
See: "MOSI and MISO data conditions" on page 507.

"MOSI/MISO"
: Sets the trigger to specified data patterns expected on the MOSI and MISO lines on the same time.

Remote command:
`TRIGger<m>:SPI:MODE` on page 1425

MOSI and MISO data conditions
The trigger on MOSI and MISO patterns is defined in the same way:



Condition ← MOSI and MISO data conditions
Selects the operator for the "Data" pattern: "Equal" or "Not equal".

Remote command:
`TRIGger<m>:SPI:FCONdition` on page 1427

MOSI pattern, MISO pattern ← MOSI and MISO data conditions
Specify the data patterns to be found on the MOSI and/or MISO line.

If the trigger type is "MOSI" or "MISO" (one pattern is defined), the maximum pattern length is 256 bit. If the trigger type is "MOSI/MISO", two patterns must be found at the same time. Thus, both patterns must have the same length, and the maximum pattern length of each pattern is 128 bit.

Enter the words in msb first bit order. The starting point of the pattern is defined by Index min, Index max and Search mode.

The bit pattern editor helps you to enter the pattern, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Remote command:
`TRIGger<m>:SPI:MOSipattern` on page 1427
`TRIGger<m>:SPI:MISopattern` on page 1427

Position ← MOSI and MISO data conditions
Operator for the data position. You can define an exact position, a position range, or let the position undefined ("Any").

Remote command:
`TRIGger<m>:SPI:DPOPerator` on page 1426

Index min, Index max ← MOSI and MISO data conditions
The effect of data positioning depends on the Search mode. It sets the number of bits or words before the first word of interest. These offset bits/words are skipped. If the position operator defines a range, the first and the last bit/word of interest are defined. The index 0 is associated with the first data bit or word.

Remote command:
`TRIGger<m>:SPI:DPOSition` on page 1427
`TRIGger<m>:SPI:DPTO` on page 1427

Search mode ← MOSI and MISO data conditions
Defines how the specified data pattern is searched:

"Word-aligned"	The pattern is matched only at word boundaries.
"Bit-aligned"	Bit-by-bit: the pattern can start at any position in the message.

Remote command:
`TRIGger<m>:SPI:PALignment` on page 1426

12.3.3.2 Triggering on SPI

Prerequisites: A bus is configured for the SPI signal to be analyzed.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Tap the "Source" button and select the "Serial bus" trigger source.
3. Select the serial bus that is set to SPI.
4. Select the "Trigger type".
5. For more complex trigger types, enter the data pattern conditions
   For details, see Chapter 12.3.3, "SPI Trigger", on page 505

12.3.4 SPI Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The decoding process considers the "Bit order" configuration setting of the signal and displays the binary result MSB first. Binary values in the combs of the decoded signal also consider the "Binary bit order" setting in the "Display" tab. Thus, you can read the bits of an LSB first signal in LSB first order in the combs while the results table displays the correct values MSB first.

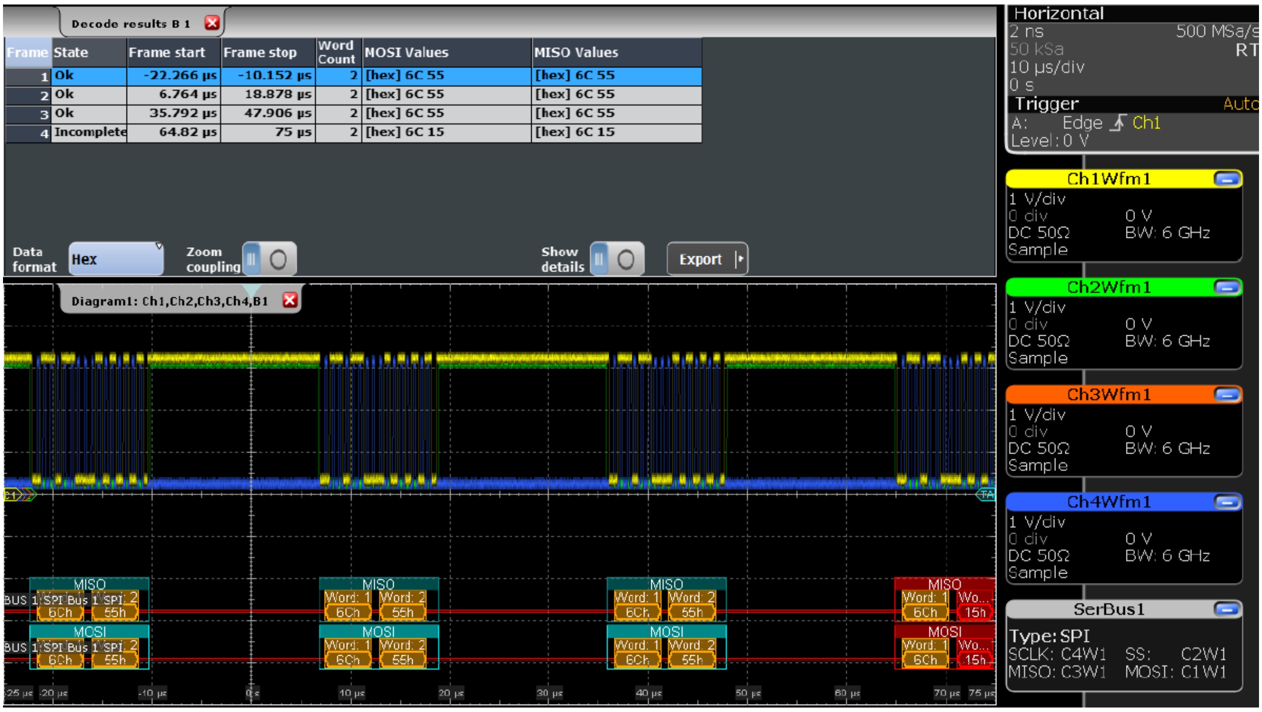


Figure 12-7: Decoded and binary SPI signal with SCLK, MOSI, and SS line

green brackets [...]	= start and end of complete frame
red brackets [...]	= start and end of incomplete frame
yellow	= word
red	= error

The "Decode results" box shows the detailed decoded data for each data frame.

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

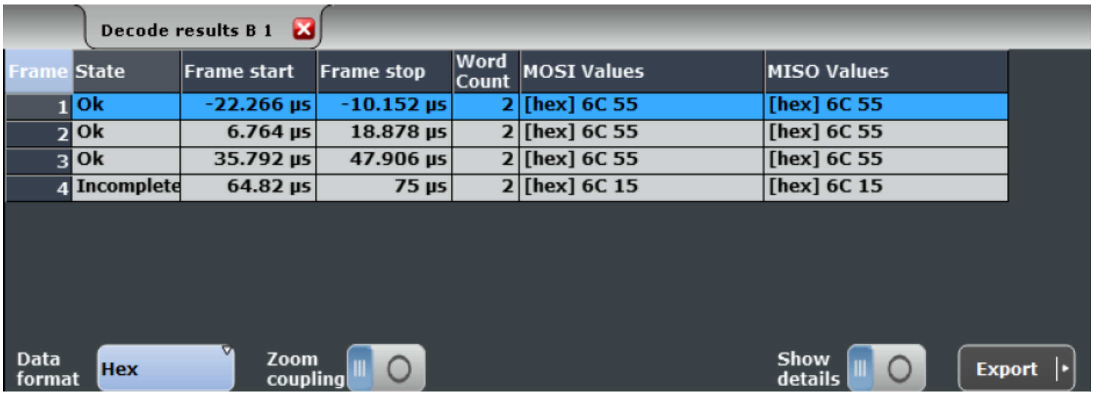


Figure 12-8: Decode results

In the figure above, the first three frames contain two words each. The fourth frame is incomplete, only one word of the frame was recognized

Table 12-4: Content of the "Decode results" table

Column	Description
State	Overall state of the frame
Frame start , Frame stop	Times of frame start and frame end
Word count	Number of words in the frame
MOSI values	Value of the MOSI data words. The data format is selected below the table.
MISO values	Value of the MISO data words. The data format is selected below the table.
Bit rate	Value of the bit rate

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.4.3, "SPI Decode Results", on page 1428.

12.3.5 Search on Decoded SPI Data

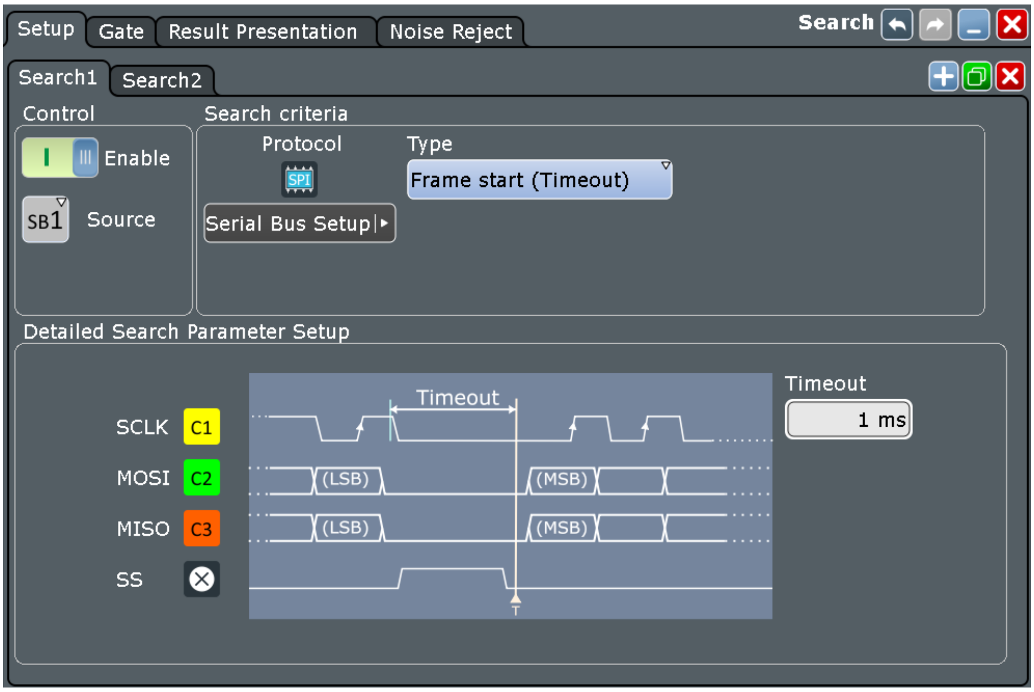
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.3.5.1 SPI Search Setup

Access: [SEARCH] > "Setup" tab



Type

Sets the event to be searched for.

"Frame start (SS)" — Searches for the start of the frame when slave select signal SS changes to the active state. This type is available if the slave select line is configured in the bus setup, and the frame condition is "SS".

"Frame start (Timeout)" — Searches for the start of the frame when the clock idle time exceeds the timeout. This trigger type is available if frame condition is set to "CLK timeout".

"MOSI" — Searches for a specified data pattern expected on the MOSI line.

"MISO" — Searches for a specified data pattern expected on the MISO line.

"MOSI / MISO" — Searches for specified data patterns expected on the MOSI and MISO lines.

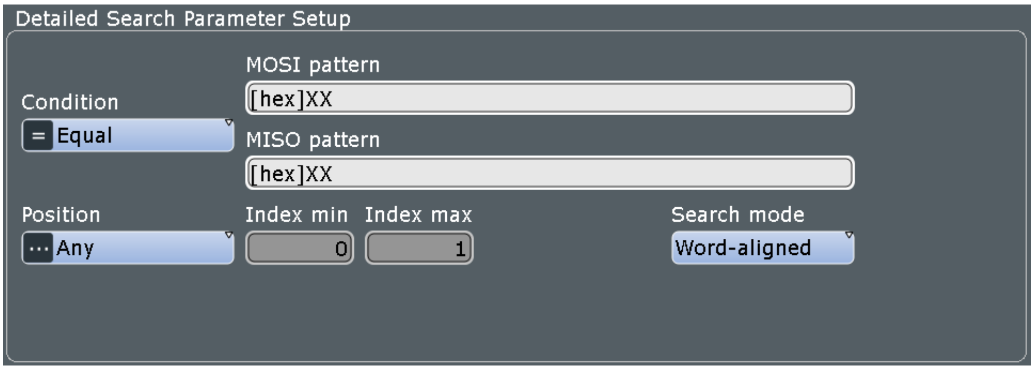
Remote command:

`SEARch:TRIGger:SPI:MODE` on page 1433

MOSI and MISO data search

The MOSI and MISO setup consists of the condition, position, MOSI pattern, MISO pattern, search mode (word-aligned, bit-aligned) and one or two index patterns.

The MOSI and MISO setup settings are the same as in the SPI trigger setup. For details, see "MOSI and MISO data conditions" on page 507.



Remote command:

`SEARch:TRIGger:SPI:FCONdition` on page 1434
`SEARch:TRIGger:SPI:MISopattern` on page 1434
`SEARch:TRIGger:SPI:MOSipattern` on page 1434
`SEARch:TRIGger:SPI:DPOPerator` on page 1434
`SEARch:TRIGger:SPI:DPOSition` on page 1435
`SEARch:TRIGger:SPI:DPTO` on page 1435
`SEARch:TRIGger:SPI:PALignment` on page 1435

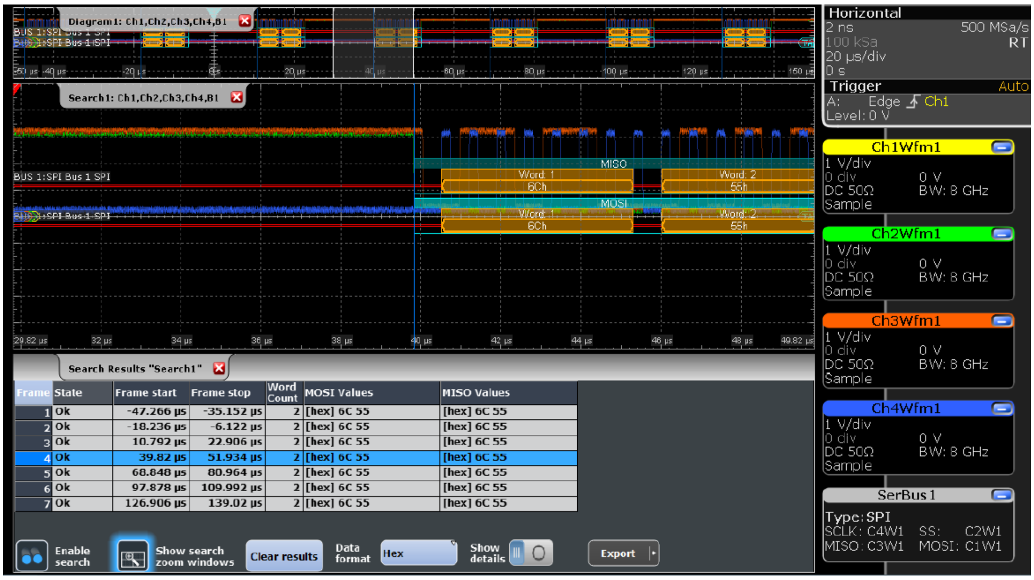
12.3.5.2 SPI Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424



Remote commands are listed in Chapter 17.17.4.5, "SPI Search Results", on page 1436.

12.4 UART/RS-232/RS-422/RS-485 (Option R&S RTE-K2)

12.4.1 The UART / RS232 Interface

The Universal Asynchronous Receiver/Transmitter UART converts a word of data into serial data, and vice versa. It is the base of many serial protocols like of RS-232. The UART uses only one line, or two lines for transmitter and receiver.

Data transfer

The data is transmitted in words, also referred to as symbols or characters. Each word consists of a start bit, several data bits, an optional parity bit, and one or more stop bits. Several words can form a package, or frame. The end of a package is marked with a reserved word or by a pause between two words.

Start	Data0	Data1	Data2	Data3	Data4	[Data5]	[Data6]	[Data7]	[Data8]	[Parity]	Stop

Figure 12-9: Bit order in a UART word (symbol)

- The start bit is a logic 0.
- The stop bits and the idle state are always logic 1.

The UART protocol has no clock for synchronization. The receiver synchronizes by means of the start and stop bits, and the bit rate that must be known to the receiver.

Trigger

The R&S RTE can trigger on specified parts of UART serial signals:

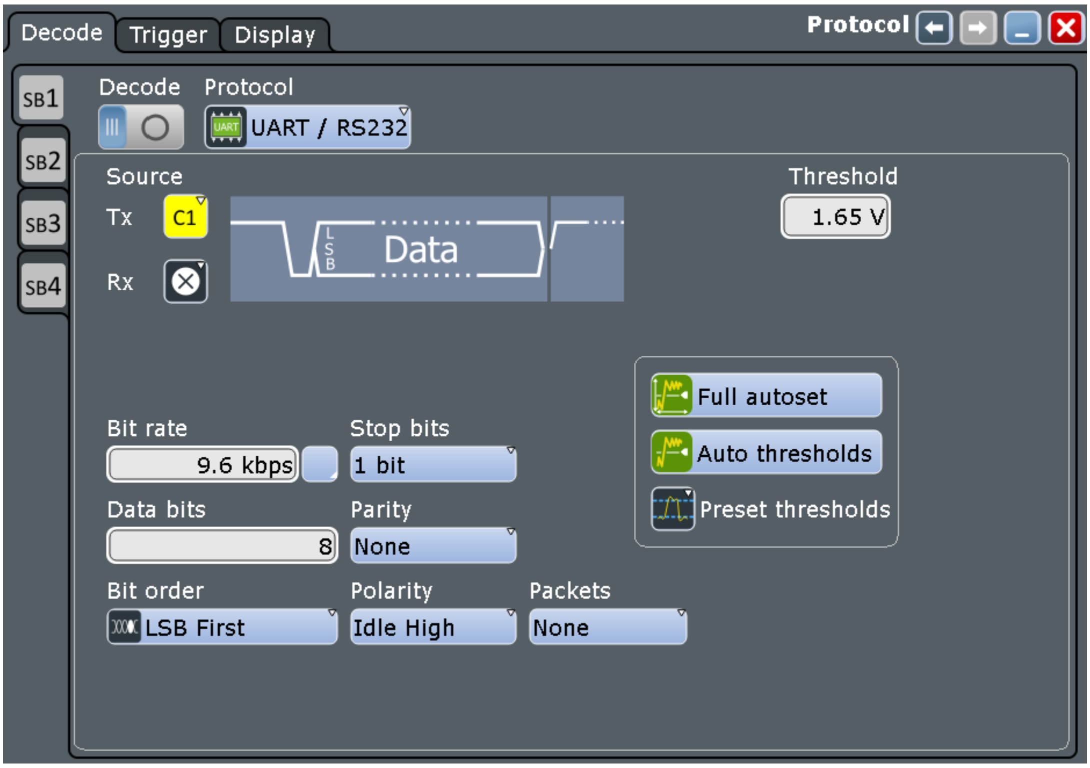
- Start bit
- Packet start
- Parity errors, and breaks
- Stop errors
- A serial pattern at any or a specified position

12.4.2 UART Configuration

12.4.2.1 UART Configuration Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol" = *UART / RS232*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Source: Tx, Rx
Select the input channels for the transmitter and receiver signals.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:UART:TX:SOURce` on page 1440
`BUS<m>:UART:RX:SOURce` on page 1440

Threshold
Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the Polarity.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Full autoset"

Starts software algorithms for determining the signal threshold levels and bitrate. See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Preset thresholds"
  Selects the default threshold voltage for various signal technologies from a list. The value is set to "Manual" if the threshold was set with "Auto threshold", or was entered directly.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:UART:RX:THReshold` on page 1441
`BUS<m>:UART:TX:THReshold` on page 1441
`BUS<m>:UART:TECHnology` on page 1441
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Polarity
Defines the logic levels of the bus. The idle state corresponds to a logic 1. the start bit to a logic 0. "Idle high" (high=1) is used, for example, for control signals, while "Idle low" (low=1) is defined for data lines (RS-232).

Remote command:
`BUS<m>:UART:POLarity` on page 1443

Bit rate
Sets the number of transmitted bits per second. To select a bit rate from list of predefined values, tap the icon beside the "Bit rate" field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

Remote command:
`BUS<m>:UART:BITRate` on page 1442
`BUS<m>:UART:BAUDrate` on page 1442

Data bits
Sets the number of data bits of a word in a range from 5 bits to 8 bits. If no parity bit is used, then 9 data bits are possible.

Remote command:
`BUS<m>:UART:SSIZe` on page 1443

Bit order
Defines if a word starts with msb (most significant bit) or lsb (least significant bit). The display of the decoded signal considers this setting, results are displayed in the specified order.

Stop bits
Sets the number of stop bits: 1 or 1.5 or 2 stop bits are possible.

Remote command:
BUS<m>:UART:SBIT on page 1443

Parity

Defines the optional parity bit that is used for error detection.

"None" No parity bit is used.

"Odd" The parity bit is set to "1" if the number of data bits set to "1" is even.

"Even" The parity bit is set to "1" if the number of data bits set to "1" is odd.

"Mark" The parity bit is always a logic 1.

"Space" The parity bit is always a logic 0.

"Don't care" The parity is ignored.

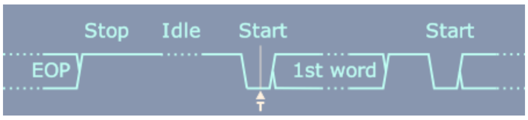
Remote command:
BUS<m>:UART:PARity on page 1442

Packets

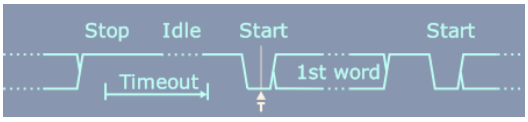
Allows you to define packets of several words in the data stream.

"None" Packets are not considered.

"End word" Defines a pattern as end condition of a packet, for example, a reserved word like CR or LF. The bit pattern editor provides frequently used values in the "Predefined values" list below the pattern table.
A new packet starts with the first start bit after the defined end pattern.



"Timeout" Defines a timeout between a stop bit and the next start bit. Enter the minimum time that marks the end of a packet.
A new packet starts with the first start bit after the timeout.



Remote command:
BUS<m>:UART:PACKets on page 1444
BUS<m>:UART:TOUT on page 1444
BUS<m>:UART:EWORd on page 1445

12.4.2.2 Configuring UART Protocol

For details on configuration settings, see Chapter 12.4.2.1, "UART Configuration Settings", on page 514.

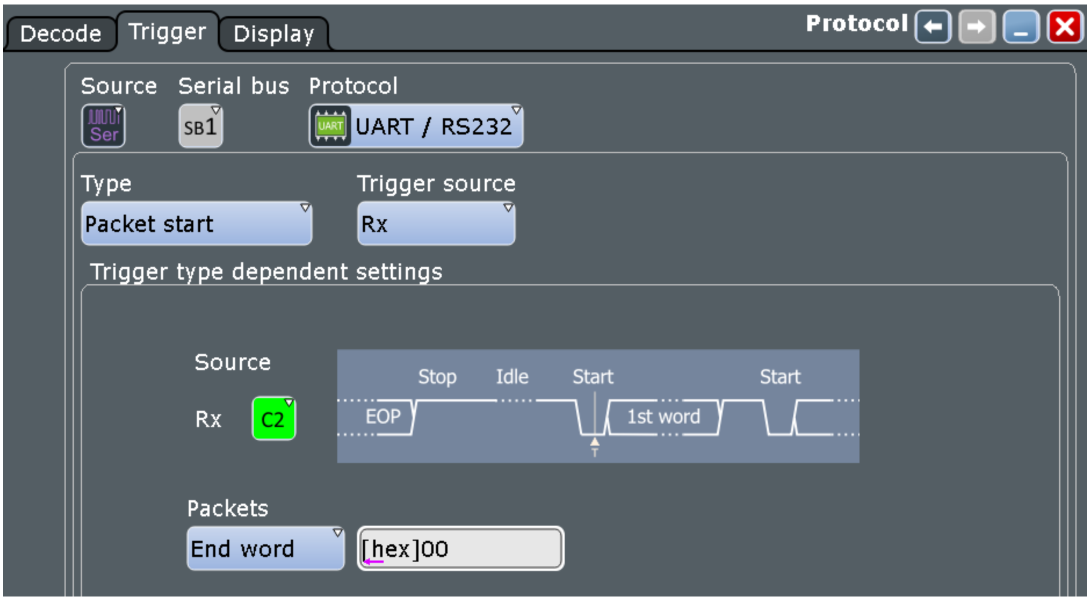
1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.

3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "UART".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Tap the "TX" button, and select the waveform of the transmitter line.
7. Tap the "Rx" button, and select the waveform of the receiver line.
8. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to an automatic value with "Full autoset"/ "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.
9. Set the "Bit rate" and "Stop bits".
10. Set the "Data bits" and "Parity".
11. Set the "Bit order", "Polarity" and "Packets".
12. Enable "Decode", if available.

12.4.3 UART Trigger

12.4.3.1 UART Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = UART / RS232"



Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Type
Selects the trigger type for UART analysis.

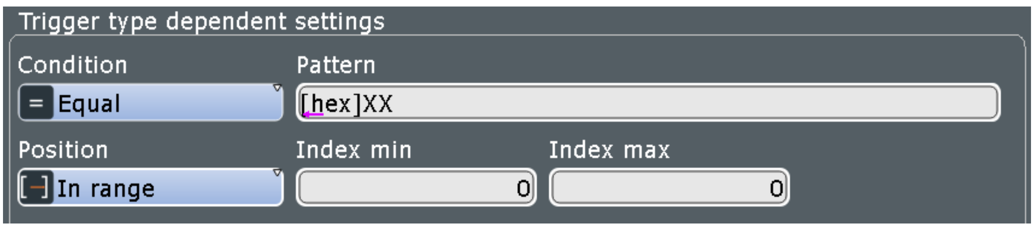
"Start bit"	Triggers on a start bit. The start bit is the first low bit after a stop bit.
"Packet start"	Triggers on the begin of a data packet. The frame start is configured with "Packets" on page 517.
"Data"	Trigger on a serial pattern at a defined position in the data packet. The pattern can include several subsequent symbols (data frames). See "Data conditions" on page 519.
"Parity error"	Triggers on a parity error indicating a transmission error. This trigger type is only available if a parity is configured for the UART bus.
"Break condition"	Triggers if a start bit is not followed by a stop bit, the data line remains at logic 0 for longer than a UART word.
"Stop error"	Triggers if the stop bit is a logic 0.

Remote command:
`TRIGger<m>:UART:TYPE` on page 1446

Trigger source
Selects the transmitter or receiver line as trigger source.

Remote command:
`TRIGger<m>:UART:SOURce` on page 1446

Data conditions
Specify the data conditions if the trigger type is set to "Data".



Condition ← Data conditions
Selects the operator for the "Data" pattern: "Equal" or "Not equal".

Remote command:
`TRIGger<m>:UART:FCONdition` on page 1447

Pattern ← Data conditions
Specifies the data pattern to be found on the specified trigger source, in binary or hex format. Enter the words in msb first bit order. The starting point of the pattern is defined by Position and Index min, Index max.

The bit pattern editor helps you to enter the pattern, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Remote command:
`TRIGger<m>:UART:DATA` on page 1447

Position ← Data conditions
Operator for the data position. You can define an exact position, or a position range.

The setting is available if packet detection is enabled in the protocol configuration.

Remote command:
`TRIGger<m>:UART:DPOPerator` on page 1446

Index min, Index max ← Data conditions
Sets the number of words before the first word of interest. These offset words are ignored. If the Position defines a range, the first and the last words of interest are defined.

The setting is available if packet detection is enabled in the protocol configuration.

Remote command:
`TRIGger<m>:UART:DPOSition` on page 1446
`TRIGger<m>:UART:DPTO` on page 1447

12.4.3.2 Triggering on UART Signals

Prerequisites: An UART bus is configured, see Chapter 12.4.3.1, "UART Trigger Settings", on page 518.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to UART.
5. Select the "Trigger type" and "Trigger source".
6. For more complex trigger types, enter the data conditions: pattern, position, index.
   For details, see Chapter 12.2.3, "I²C Trigger", on page 487

12.4.4 UART Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The decoding process considers the "Bit order" configuration setting of the signal and displays the binary result MSB first. Binary values in the combs of the decoded signal also consider the "Binary bit order" setting in the "Display" tab. Thus, you can read the bits of an LSB first signal in LSB first order in the combs while the results table displays the correct values MSB first.

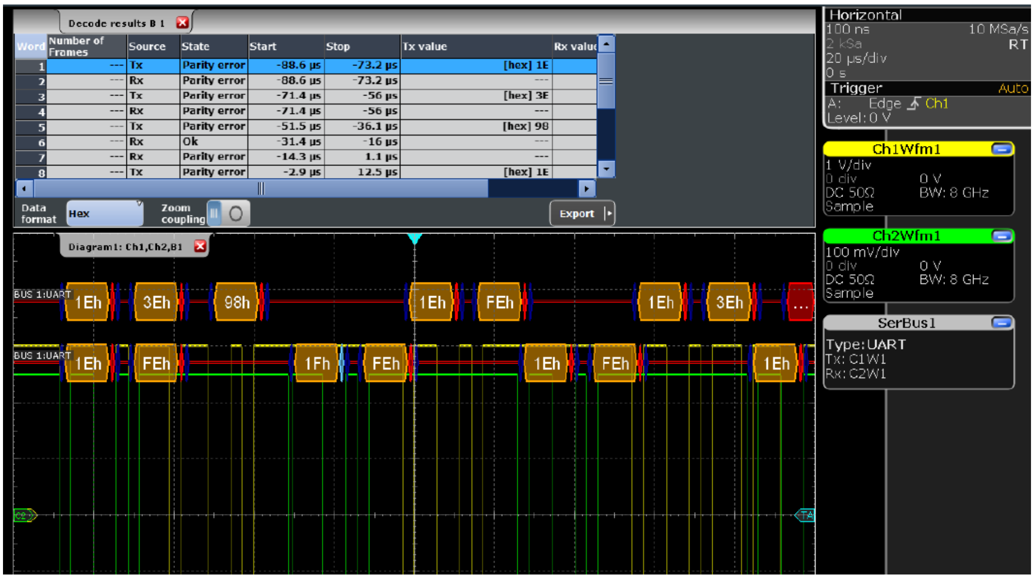


Figure 12-10: Decoded and binary UART signal

blue = start and stop bits if ok
red = start error, stop error, parity error
orange = parity bit if ok
yellow = word ok
magenta = word contains error

The "Decode results" box shows the detailed decoded data for each word.

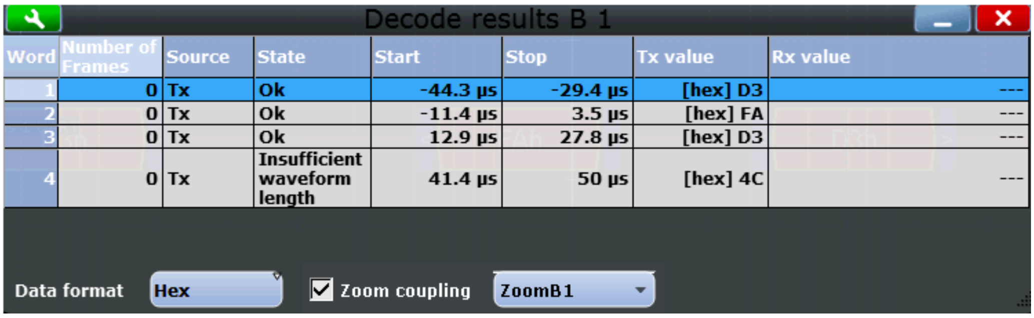


Figure 12-11: Decode results of the UART signal

Table 12-5: Content of the "Decode results" table

Column	Description
Number of Frames	
Source	Line, Tx or Rx
State	Decoding state of the word. "Insuffcient waveform length" indicates that the word is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Start	Time of the word start (start bit)
Stop	Time of the word stop (stop bit)
Tx value	Value of the Tx word. The data format is selected below the table.
Rx value	Value of the Rx word. The data format is selected below the table.
Bit rate	Value of the bit rate

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.5.3, "Decode Results", on page 1447.

12.5 CAN and CAN FD (Options R&S RTE-K3 and -K9)

CAN is the Controller Area Network, a bus system designed by Bosch for use within automotive network architecture, for example, for brake, power train and engine management. Today, it is also used in many other systems, for example, in industrial machines, aerospace, subsea, merchant marine etc..

More than 20 years after the invention of CAN, communication needs have increased, and CAN has reached it's bandwidth limits in some application fields. Therefore, Bosch specified an improved CAN protocol with flexible data rate - CAN FD. It introduces a higher bit rate in the data phase up to 15 Mbit/s and an extended data field from up to 64 bytes.

The R&S RTE provides decoding, triggering and searching CAN and CAN FD signals with following options:

- CAN: option R&S RTE-K3
- CAN FD: option R&S RTE-K9, requires CAN option R&S RTE-K3

12.5.1 CAN and CAN-FD Configuration

Access: [PROTOCOL] key > "Decode" tab > "Protocol" = "CAN" or "CAN/CAN-FD"

Make sure that the tab of the correct serial bus is selected on the left side.

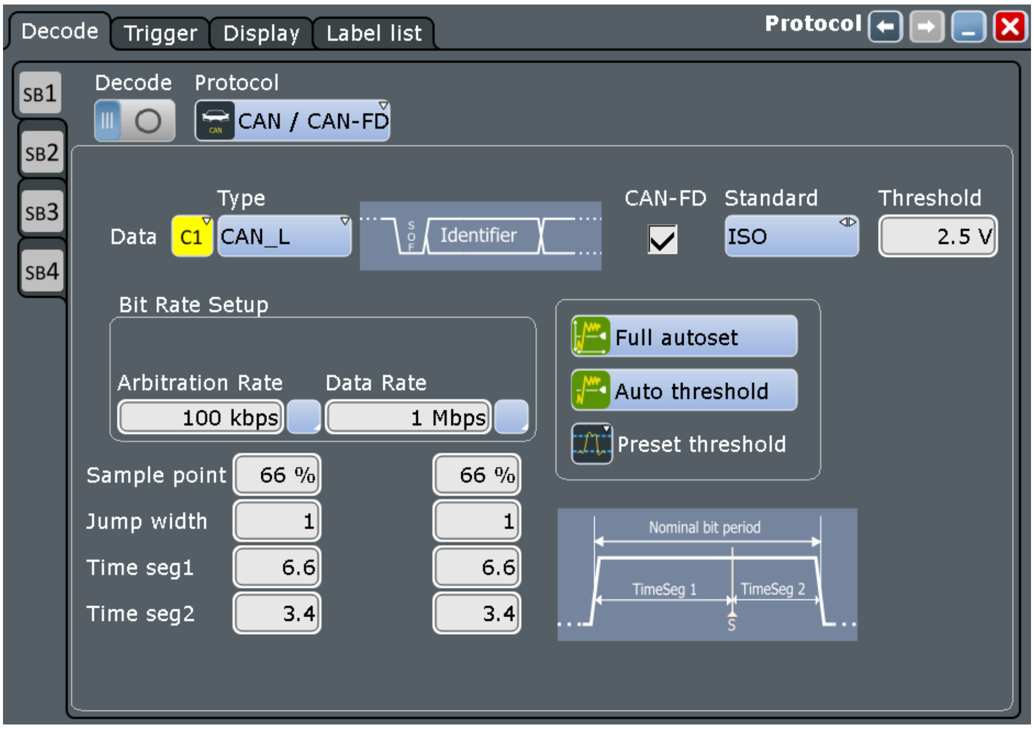


Figure 12-12: Configuration for CAN FD

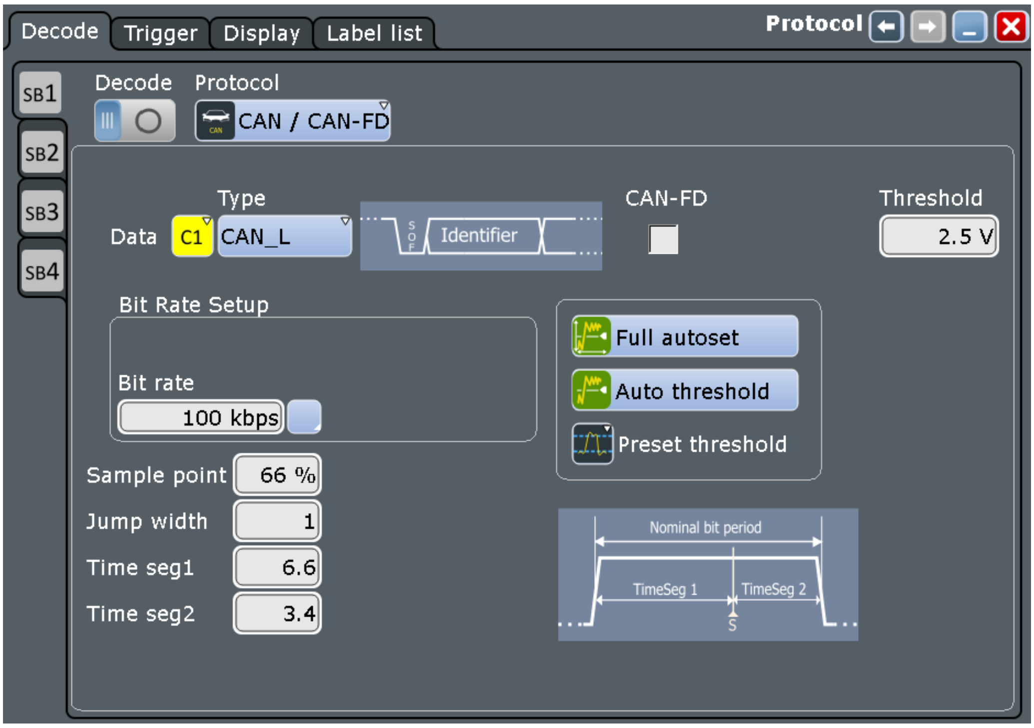


Figure 12-13: Configuration for CAN

See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Protocol

Defines the protocol type of the selected serial bus.

Remote command:
BUS<m>:TYPE on page 1382

Decode

Enables the decoding of the selected bus. The signal icon of the bus appears on the signal bar.

Remote command:
BUS<m>[:STATe] on page 1382

Data

Sets the source of the selected data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

A math waveform can be used, for example, if you probe on CAN-High and CAN-Low using two single-ended probes, and the difference between high and low is calculated and displayed using a math waveform.

Remote command:
BUS<m>:CAN:DATA:SOURce on page 1450

Type

Selects the CAN-High or CAN-Low line. CAN uses both lines for differential signal transmission.

If you measure with a differential probe, connect the probe to both CAN-H and CAN-L lines, and select the data "Type" = *CAN-H*.

If you use a single-ended probe, connect the probe to either CAN_L or CAN_H, and select the data type accordingly.

Remote command:
BUS<m>:CAN:TYPE on page 1451

CAN-FD

Enables the CAN FD protocol configuration and displays additional CAN FD parameters.

The setting is available in CAN FD option R&S RTE-K9.

Remote command:
BUS<m>:CAN:FDATa:ENABle on page 1452
BUS<m>:CAN:FDATa:FRAMe<n>:STANdard? on page 1463

Standard

Only available for CAN FD buses.

"Non-ISO" Signals are decoded according to the the Bosch CAN FD protocol.

"ISO" Signals are decoded according to the the ISO CAN FD protocol. This protocol has an additional stuff count field before the CRC sequence.

Remote command:
`BUS<m>:CAN:FDATa:PSTandard` on page 1451

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the boolean logic). Otherwise, the signal state is considered low (0 or false).

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Full autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
  See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Preset thresholds"
  Selects the default threshold voltage for various signal technologies from a list. The value is set to "Manual" if the threshold was set with "Auto threshold", or was entered directly.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:CAN:DATA:THReshold` on page 1451
`BUS<m>:CAN:TECHnology` on page 1452
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Bit rate (CAN) / Arbitration rate (CAN FD)
For CAN buses, the "Bit rate" sets the number of transmitted bits per second.

For CAN FD buses, this parameter is called "Arbitration rate" and sets the bit rate of the arbitration phase.

The maximum value of this rate is 1 Mbit/s. The bit rate is uniform and fixed for a given CAN or CAN FD bus.

To select a bit rate from the list of predefined values, tap the button beside the field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

Remote command:
`BUS<m>:CAN:BITRate` on page 1452

Data rate
The setting is available in CAN FD option R&S RTE-K9.

Sets the bit rate of the data phase. The data rate can be equal or higher than the arbitration rate; and it is uniform and fixed for a given CAN FD bus.

To select a data rate from the list of predefined values, tap the button beside the field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

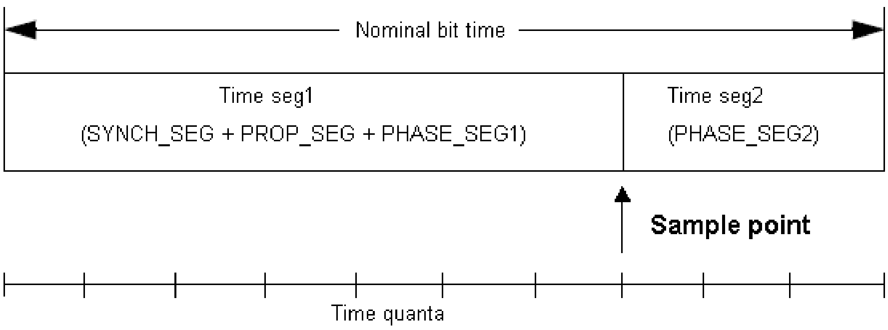
Remote command:
BUS<m>:CAN:FDATa:DBITrate on page 1453

Synchronization: Sample point, Time segments, Jump width

The CAN bus interface uses an asynchronous transmission scheme. The standard specifies a set of rules to resynchronize the local clock of a CAN node to the message.

The sample point divides the nominal bit period into two distinct time segments. The length of the time segments is defined in time quanta according to network and node conditions during CAN development.

For CAN FD signals, you can define the sysnchronization settings separately for the arbitration phase and data phase.



To specify the bit timing, enter either "Time seg1" and "Time seg2", or directly the "Sample point". Additionally, set the "Jump width".

"Time seg1, Time seg2"
Set the number of time quanta before the sample point (Time seg1) and after the sample point (Time seg2). The "Sample point" percentage value is adjusted accordingly.
Time seg1 comprises the segments Synch_seg, Prop_seg, and Phase_seg1 which are specified in the CAN standard. Time seg2 matches Phase_seg2 from the standard.
The maximum sum of Time seg1 and Time seg2 is 24.

"Sample point"
Sets the position of the sample point within the bit in percent ot the nominal bit time.
The time quanta values "Time seg1, Time seg2" are adjusted accordingly.

"Jump width"
Time segment1 may be lengthened or time segment2 may be shortened due to resynchronization. Resynchronization corrects the phase error of an edge caused by the drift of the oscillators. The jump width defines the maximum number of time quanta for phase correction.
The maximum value of the jump width is 4, or *Time seg1 - Time seg2* if this difference is lower than 4.

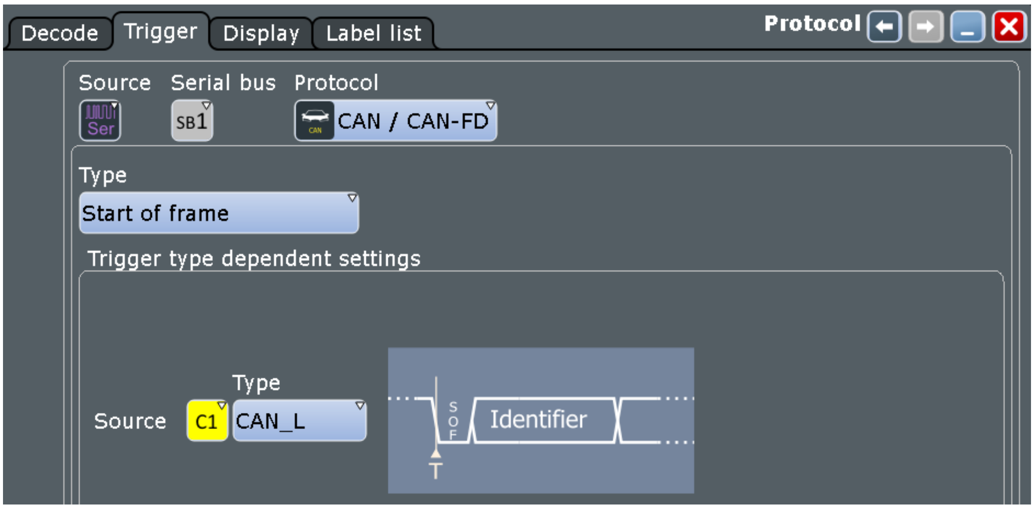
Remote command:
BUS<m>:CAN:T1Segment on page 1453
BUS<m>:CAN:T2Segment on page 1454
BUS<m>:CAN:SAMPlepoint on page 1453
BUS<m>:CAN:JWIDth on page 1454
BUS<m>:CAN:FDATa:T1Segment on page 1453
BUS<m>:CAN:FDATa:T2Segment on page 1454
BUS<m>:CAN:FDATa:SAMPlepoint on page 1453
BUS<m>:CAN:FDATa:JWIDth on page 1454

12.5.2 CAN / CAN FD Trigger

The R&S RTE can trigger on various events in a CAN or CAN FD frame. Trigger conditions include start of frame, frame ID, data pattern, or error conditions.

12.5.2.1 Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = CAN/ CAN-FD"



Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Serial bus

Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:
TRIGger<m>:SOURce:SBSelect on page 1387

Protocol

Defines the protocol type of the selected serial bus.

Remote command:
BUS<m>:TYPE on page 1382

Trigger type

Selects the trigger type for CAN analysis.

"Start of frame"	Triggers on the first edge of the dominant SOF bit (synchronization bit).
"Frame type"	Triggers on a specified frame type (data, remote, error, or overload). For data and remote frames, also the identifier format is considered. For details, see: ● "Frame type" on page 530 ● "ID type" on page 530
"Identifier"	Sets the trigger to a specific message identifier or an identifier range. See "Identifier setup: Condition, Identifier min, Identifier max" on page 531.
"Identifier + Data"	Sets the trigger to a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern. The identifier conditions are the same as for the "Identifier" trigger type, see "Identifier setup: Condition, Identifier min, Identifier max" on page 531. Data conditions are set with "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 532.
"Error condition"	Identifies various errors in the frame, see "Error conditions" on page 534.
"Symbolic"	The "Symbolic" trigger type is available if a DBC label list file is loaded and applied. It allows you to trigger on a specific data message, or a signal and its value that appears inside the message, see Chapter 12.5.6.1, "Symbolic Trigger", on page 555.

Remote command:
TRIGger<m>:CAN:TYPE on page 1455

Standard

Selects the CAN standard: "CAN", "CAN FD", or "Any".

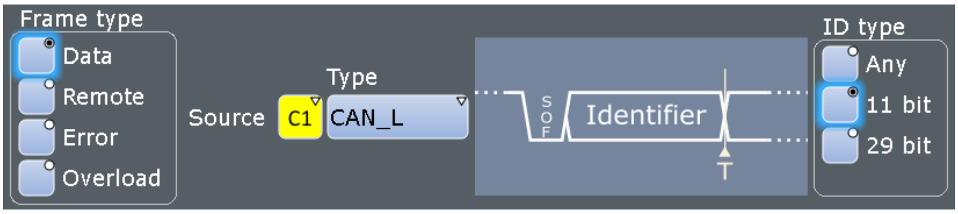
The setting is available in CAN FD option R&S RTE-K9.

Use "Any" to trigger on either CAN or CAN-FD frame. In this case, the trigger configuration provides all possible settings, for CAN as well as for CAN FD.

Remote command:
TRIGger<m>:CAN:FDATa:STANdard on page 1456

Frame type

CAN has four frame types which can be used as trigger condition.



For data and remote frames, the identifier format has to be set with ID type.

"Data" The data frame is the only frame for actual data transmission.

"Remote" Remote frames are only available in the CAN protocol.
The remote frame initiates the transmission of data by another node.
The frame format is the same as of data frames but without the data field.

"Error" When a node recognizes an error, it cancels transmission by sending an error frame.
The instrument triggers seven bit periods after the end of the error flag that is marked by a dominant-recessive edge.
The ID type is irrelevant for error frames.

"Overload" When a node needs a delay between data and/or remote frames, it sends and overload frame.
The instrument triggers seven bit periods after the end of the overload flag that is marked by a dominant-recessive edge.
The ID type is irrelevant for overload frames.

Remote command:
TRIGger<m>:CAN:FTYPe on page 1456

ID type

Selects the length of the identifier:

"11 bit" Identifier length of the CAN base frame format. The instrument triggers on the sample point of the IDE bit (identifier extension flag).

"29 bit" Identifier length of the CAN extended frame format. The instrument triggers on the sample point of the RTR bit.

"Any" The ID type and ID pattern are not relevant for the trigger condition.
If the trigger type is "Identifier", the instrument triggers on any identifier in the specified frame type.
If the trigger type is "Identifier + Data", set the "ID type" to "Any" if you want to trigger only on data.

Remote command:
TRIGger<m>:CAN:ITYPe on page 1457

Identifier setup: Condition, Identifier min, Identifier max
The identifier setup consists mainly of the condition and one or two identifier patterns. Additionally, ID type and frame type may qualify the identifier.

The trigger point depends on the ID type.

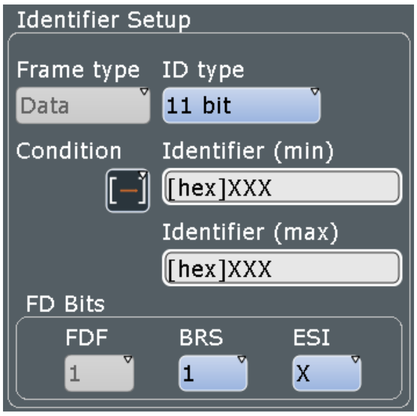


Figure 12-14: Identifier setup for CAN FD

"Frame type" Data frames and remote frames contain an identifier. Select the frame type to be triggered on, or select "Any" if the frame type is not relevant.
In CAN FD, only "Data" frames are available.

"ID type" See: "ID type" on page 530.

"Condition" Defines the operator to set a specific identifier ("Equal" or "Not equal") or an identifier range.

"Identifier min" Defines the bit pattern of the message identifier. In binary format, use the following characters: 1; 0; or X (any bit). The use of X is restricted to the conditions "Equal" and "Not equal".
The length of the bit patterns is restricted to the selected "ID type".
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Identifier max" The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

"FD bits" See: "FD bits" on page 531.

Remote command:
`TRIGger<m>:CAN:ICONdition` on page 1457
`TRIGger<m>:CAN:IMIN` on page 1458
`TRIGger<m>:CAN:IMAX` on page 1458

FD bits
For standard settings "CAN FD" and "Any", you can trigger on CAN FD-specific bits.

"FDF" The bit determines whether a frame is CAN or CAN-FD. It corresponds to the EDL bit (extended data length), which only exists in CAN FD format. If you do not know if the signal is CAN or CAN FD, you can use this bit to identify the format: FDF = 1 is CAN FD, and FDF = 0 is CAN. Set "X" if the format is not relevant.

"BRS" is the bit rate switch bit. Value 1 means that the bit rate switches from the "Arbitration rate" to the faster "Data rate".

"ESI" is the error state indicator. Set "X" if the bit is not relevant.

Remote command:
`TRIGger<m>:CAN:FDATa:FDF` on page 1458
`TRIGger<m>:CAN:FDATa:BRS` on page 1458
`TRIGger<m>:CAN:FDATa:ESI` on page 1458

Data setup: DLC, NDB, Transfer, Condition, Data min, Data max

The data setup consists of the transfer direction, the number of bytes, the condition, and one or two data patterns.

To trigger only on data, set the "ID type" of the identifier setup to "Any".

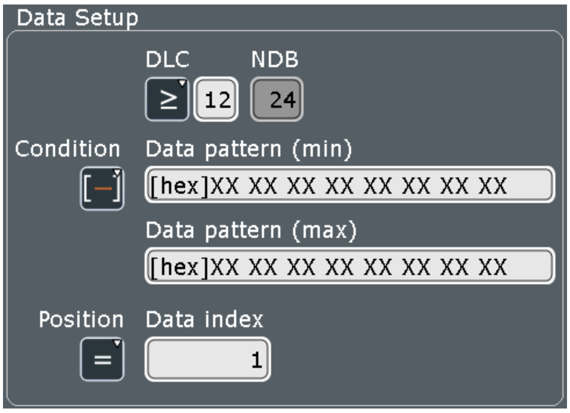


Figure 12-15: Data setup for CAN FD

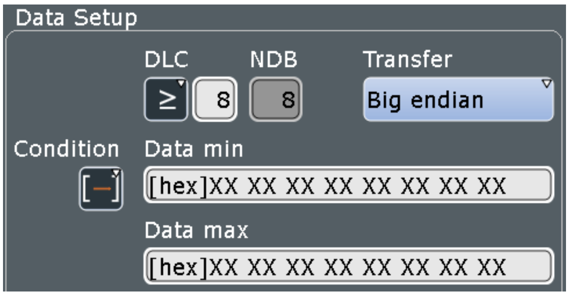


Figure 12-16: Data setup for CAN

"Transfer" CAN only:

Sets the byte order (endianness) of the data transfer. With "Big endian", the data is analyzed and evaluated in the order of reception. With "Little endian", the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

"DLC, NDB" "DLC" sets the Data Length Code, which defines the number of data bytes to be found.
"NDB" shows the number of data bytes that is set by the DLC. DLC and NDB are different in CAN FD for DLCs > 8.
CAN:

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.
Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.
CAN FD:
The data field can have up to 64 bytes, the DLC is defined in the standard. For example, DLC = 9 defines that the data field has 12 bytes, and DLC = 15 sets a 64 byte data field.

"Condition" Sets the operator to set a specific data pattern ("Equal" or "Not equal") or a data range.

"Data min" Defines the data pattern. The pattern length is adjusted to the DLC setting (and vice versa). Enter the pattern MSB first and with big endian byte order.
In binary format, use the following characters: 1; 0; or X (any bit). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CAN:BORDer` on page 1459
`TRIGger<m>:CAN:DCONdition` on page 1459
`TRIGger<m>:CAN:DMIN` on page 1459
`TRIGger<m>:CAN:DMAX` on page 1459
`TRIGger<m>:CAN:DLCCondition` on page 1460
`TRIGger<m>:CAN:DLC` on page 1460
`TRIGger<m>:CAN:NDBYtes?` on page 1460

Data position
The data position sets the location in the data field where the instrument looks for the specified data pattern.

The setting is available in CAN FD option R&S RTE-K9.

The position can be defined if the data field of the frame is longer than 8 bytes - if DLC ≥9.

"Position" Sets the operator to define an exact position ("Equal") or a data range.
Use "Any", if the data position is not relevant for the trigger condition.

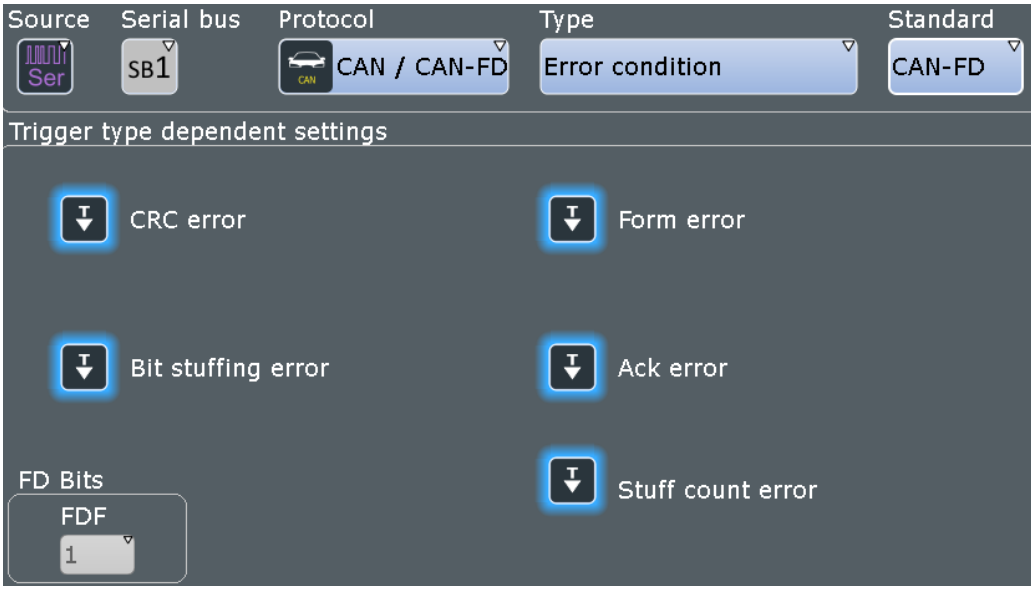
"Data index (min)" Defines the number of the first data byte at which the data pattern may start.

"Data index (max)" Sets the number of the last byte at which the required data pattern may start if the "Position" operator is "In range".

Remote command:
`TRIGger<m>:CAN:FDATa:DPOPerator` on page 1461
`TRIGger<m>:CAN:FDATa:DPOSition` on page 1461
`TRIGger<m>:CAN:FDATa:DPTO` on page 1461

Error conditions

If a CAN detects an error, it transmits an error flag at the next bit. The R&S RTE detects errors in the message and triggers on these errors even if no CAN node sends an error flag.



- CRC error
  CAN uses the Cyclic Redundancy Check, which is a complex checksum calculation method. The transmitter calculates the CRC and sends the result in the CRC sequence. The receiver calculates the CRC in the same way. A CRC error occurs when the calculated result differs from the received value in the CRC sequence.
- Bit stuffing error
  The frame segments Start Of Frame, Arbitration Field, Control Field, Data Field and CRC Sequence are coded by the bit stuffing method. The transmitter automatically inserts a complementary bit into the bit stream when it detects five consecutive bits of identical value in the bit stream to be transmitted. A stuff error occurs when the 6th consecutive equal bit level in the mentioned fields is detected.
- Form error
  A form error occurs when a fixed-form bit field contains one or more illegal bits.
- Ack error
  An acknowledgement error occurs when the transmitter does not receive an acknowledgment - a dominant bit during the Ack Slot.
- Stuff count error

A stuff count error occurs if th received stuff count value does not match the value calculated from the own stuff bit count. Only relevant for CAN FD signals in ISO standard.

Remote command:
TRIGger<m>:CAN:CRCerror on page 1462
TRIGger<m>:CAN:BITSterror on page 1462
TRIGger<m>:CAN:FORMerror on page 1462
TRIGger<m>:CAN:ACKerror on page 1461
TRIGger<m>:CAN:FDATa:SCERror on page 1462

12.5.2.2 Triggering on CAN FD Data

The "Identifier + Data" trigger type supports triggering on data bytes of specific value at specific location in the data field of a frame. The "Data Pattern" field provides 8 data bytes to define the data pattern. For data fields longer than 8 byte, you can define the position where the specified pattern starts.

The following examples demonstrate how the data pattern and data position are defined.

To set up the trigger

1. Set the basic trigger events:
   a) Select the source: "Serial bus".
   b) Select the serial bus.
   c) Select the protocol: "CAN/CAN FD".
   d) Select the trigger type: "Identifier + Data".
   e) Select the standard:"CAN FD" or "Any".
2. In this example, the identifier does not matter. Set the "ID type = Any".
3. Define the data setup as described in the examples.

Example: Triggering on the second data byte

The CAN FD frame has 2 or more data bytes, where the value of the second data byte should be E7.

- Set "DLC ≥ 2".
- Set the data pattern: "= XX E7".

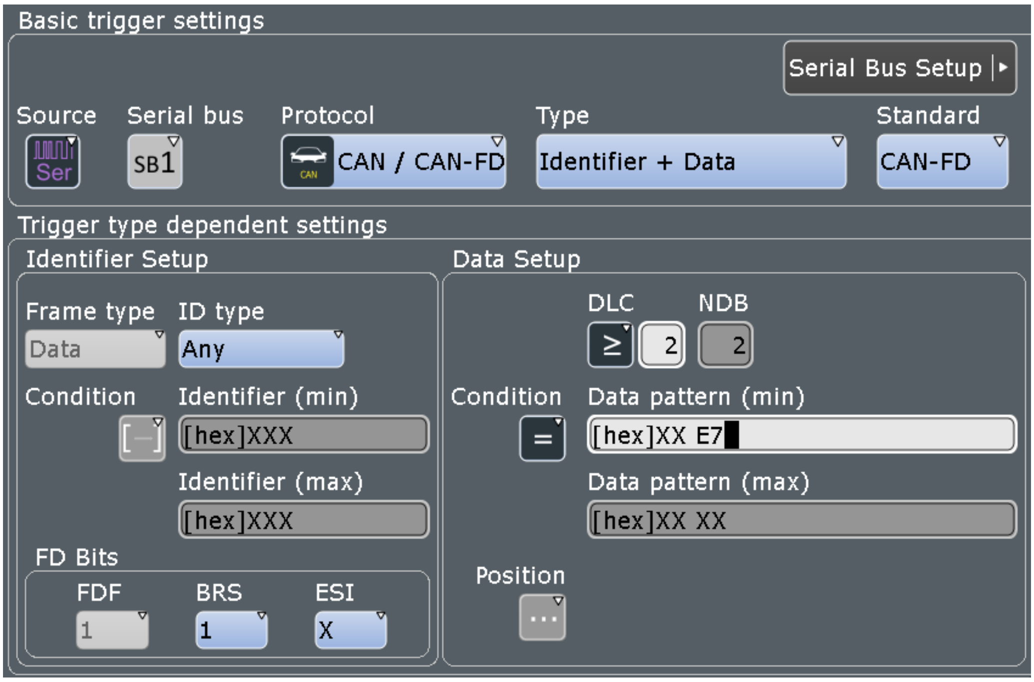
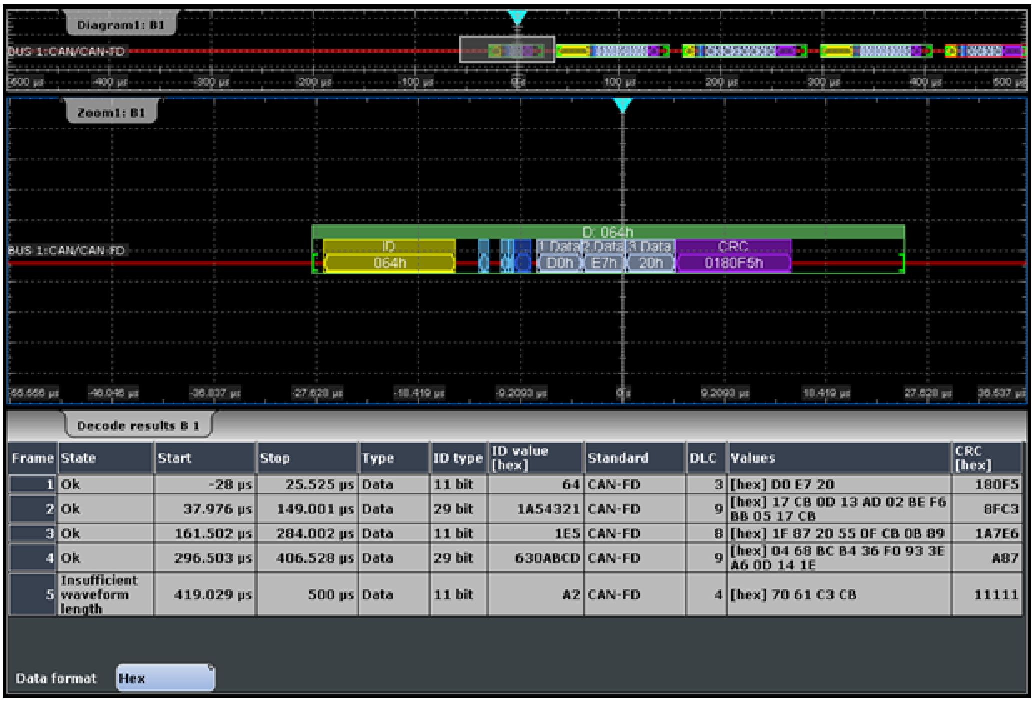


Figure 12-17: Trigger setup to trigger on the 2nd data byte with value = E7



Frame	State	Start	Stop	Type	ID type	ID value [hex]	Standard	DLC	Values	CRC [hex]
1	Ok	-28 µs	25.525 µs	Data	11 bit	64	CAN-FD	3	[hex] D0 E7 20	180F5
2	Ok	37.976 µs	149.001 µs	Data	29 bit	1A54321	CAN-FD	9	[hex] 17 CB 0D 13 AD 02 BE F6 BB 05 17 CB	8FC3
3	Ok	161.502 µs	284.002 µs	Data	11 bit	1E5	CAN-FD	8	[hex] 1F 87 20 55 0F CB 08 89	1A7E6
4	Ok	296.503 µs	406.528 µs	Data	29 bit	630ABCD	CAN-FD	9	[hex] 04 68 BC B4 36 F0 93 3E A6 0D 14 1E	A87
5	Insufficient waveform length	419.029 µs	500 µs	Data	11 bit	A2	CAN-FD	4	[hex] 70 61 C3 CB	11111

Figure 12-18: Data type trigger on 2nd data byte with value = E7

Example: Triggering on the seventh or later data byte

The CAN FD frame has 12 or more data bytes, where the value of the seventh or later data byte should be 17.

- Set "DLC ≥ 9".
- Set the data pattern: "= XX XX XX XX XX XX 17 XX".

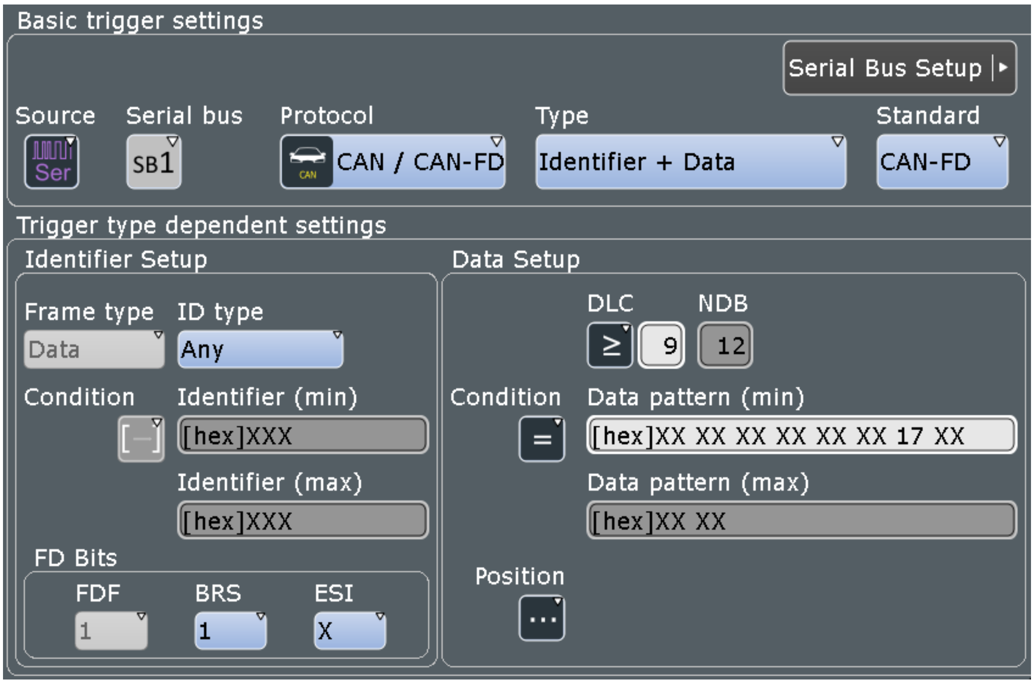


Figure 12-19: Trigger setup to trigger on the 7th or later data byte with value = 17

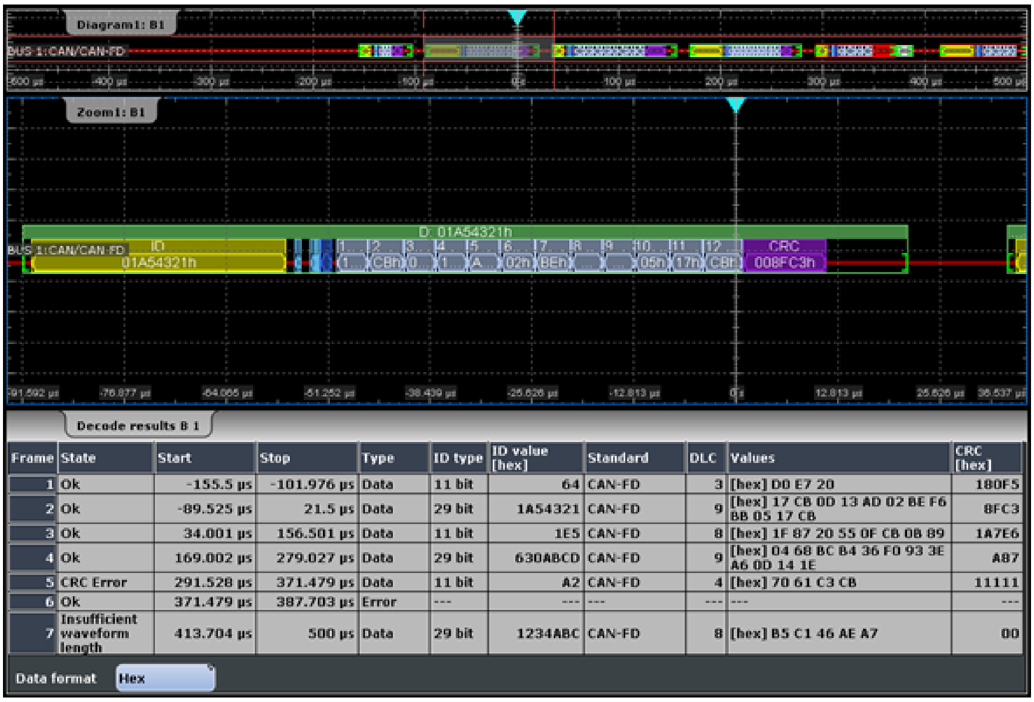


Figure 12-20: Data type trigger on data byte with the 7th or later byte value = 17

Example: Triggering on a data byte at a given position

The CAN FD frame has 12 or more data bytes. the trigger is set at the 8th data byte starting at the 4th data byte or later, with data byte value = 17.

- Set "DLC ≥ 9".
- Set the data pattern: "= XX XX XX XX XX XX XX 17 ".
- Set the position of the data pattern: "In range", "4" to "12".

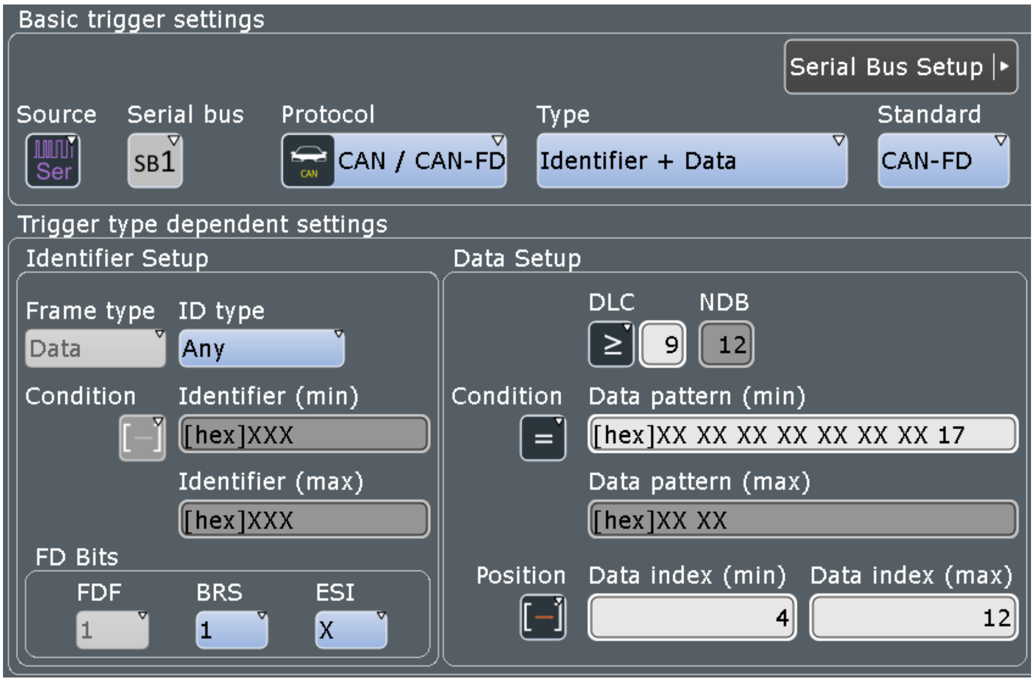


Figure 12-21: Trigger setup to trigger on data byte with value = 17 at 11th data byte location

The instrument skips the first 3 data bytes and starts comparing the data pattern with the 4th data byte. So, the byte with value 17 can be found between the 11th and the 19th data byte.

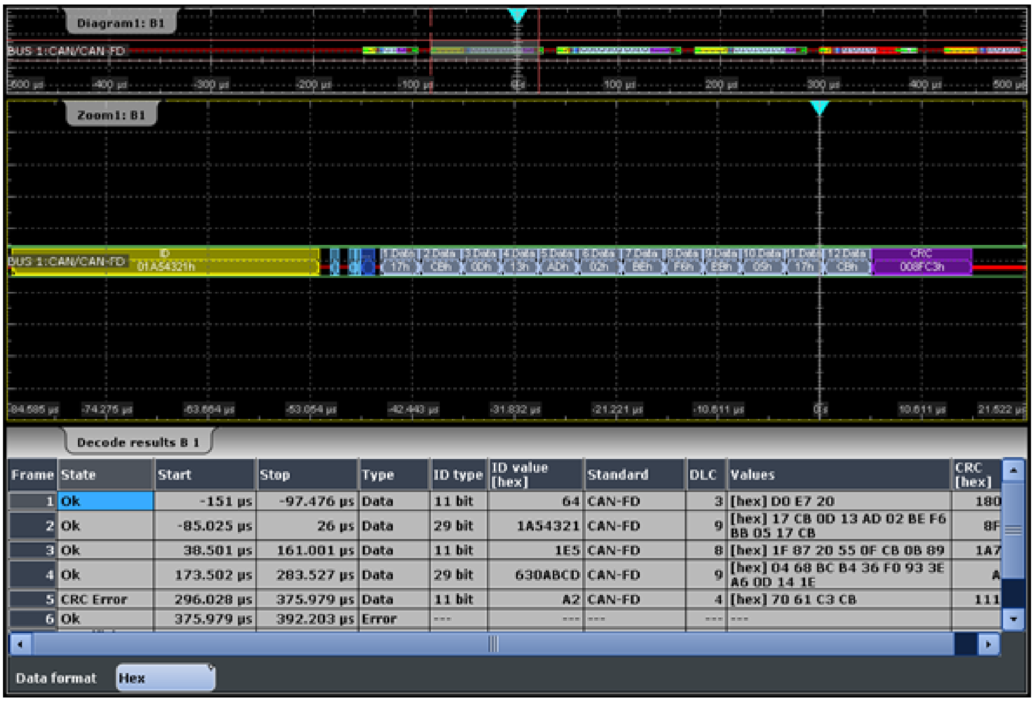


Figure 12-22: Data type trigger on data byte with value = 17 at 11th data byte location

12.5.3 CAN / CAN FD Label List

Label list files (symbolic data files) for CAN and CAN FD protocols are available in PTT and CSV file formats, similar to other serial protocols. In addition, the R&S RTE can read and apply DBC files to the decoded signal and provides settings for symbolic triggering and symbolic search.

Note: In the following, CAN means both protocols: CAN, and CAN FD.

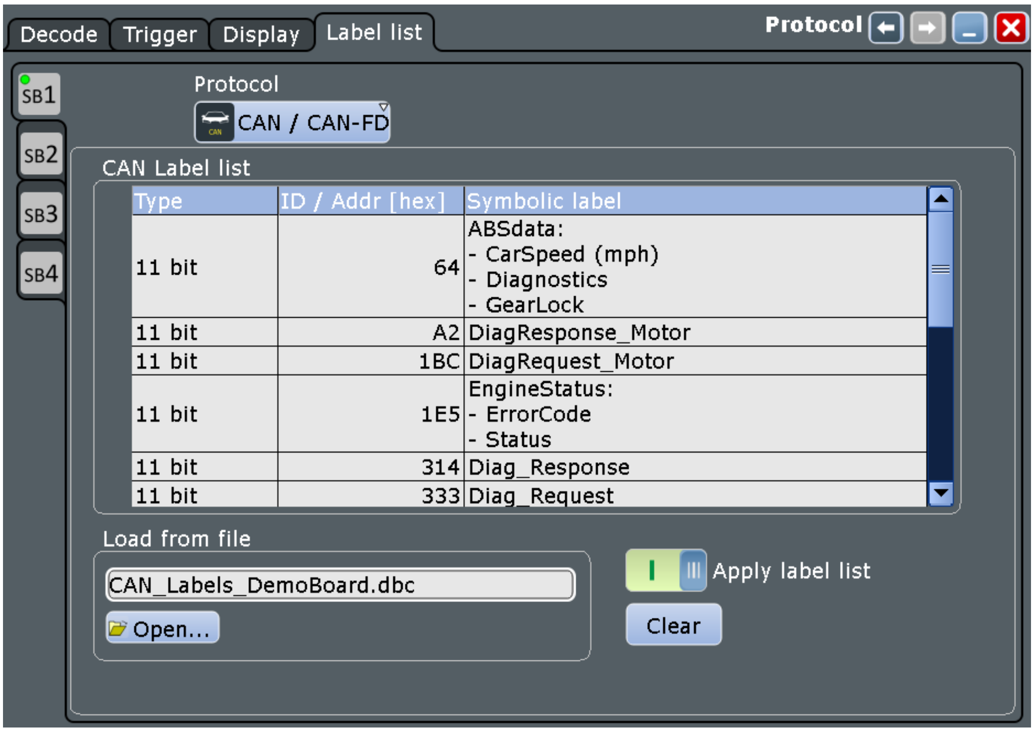
12.5.3.1 DBC Files for CAN / CAN FD

Industry-standard DBC files contain more information than PTT and CSV files and translate the abstract decode results to human language. For each frame, the frame ID and the symbolic name of the ID are given; the frames are also called messages in CAN. The data of a CAN message can consist of several "signals". The DBC file provides the label, unit, start bit, length and other indicators for each signal. For state-encoded signals, the meaning of the states is given.

In the demo example, the message "EngineData" has the decimal ID 2,166,573,756 and consists of 8 data bytes. These 8 bytes are defined as 6 signals. The first one, "PetrolLevel", starts at bit #24, has a length of 8 bit, and the unit is liter. The signal "IdleRunning" is state-encoded. It has only one bit. The binary value 0 means "Running", and the binary value 1 means "Idle".

Example: CAN DBC file section

```
BO_ 2166573756 EngineData: 8 Engine
 SG_ PetrolLevel : 24|8@1+ (1,0) [0|255] "l" ...
 SG_ EngPower : 48|16@1+ (0.01,0) [0|350] "kW" ...
 SG_ EngForce : 32|10@1+ (1,0) [0|1000] "N" ...
 SG_ IdleRunning : 23|1@1+ (1,0) [0|1] "" ...
 SG_ EngTemp : 16|7@1+ (2,-50) [-50|150] "degC" ....
 SG_ EngSpeed : 0|13@1+ (1,0) [0|8000] "rpm" ...
....
 VAL_ 2166573756 IdleRunning 0 "Running" 1 "Idle" ;
```



The usage of DBC files is described in Chapter 12.5.6, "Symbolic Trigger, Decode and Search", on page 555.

12.5.3.2 PTT and CSV Files for CAN / CAN FD

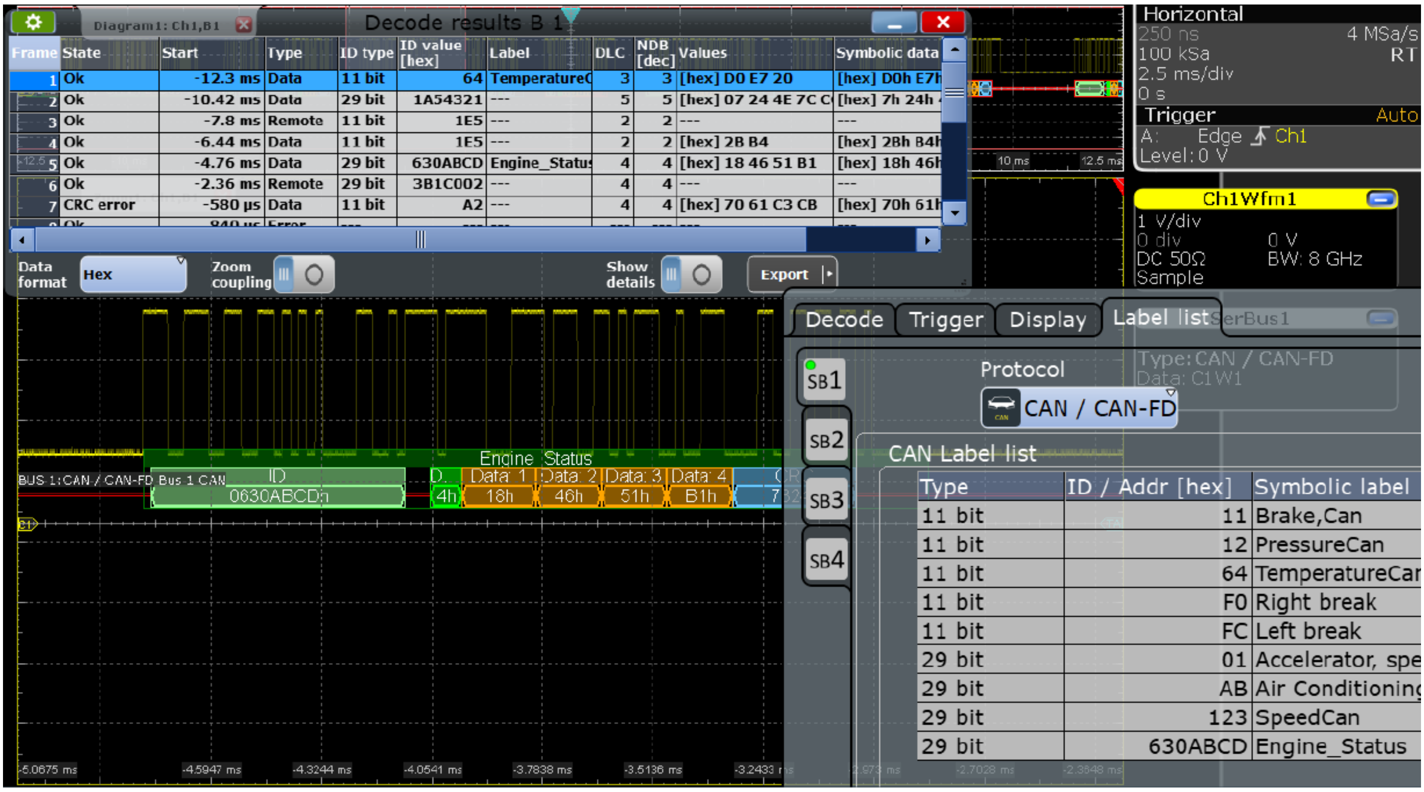
Label list files are protocol-specific. A PTT label file for CAN protocols contains three values for each identifier:

- Identifier type, 11-bit or 29-bit long
- Identifier value
- Label, symbolic name of the identifier, specifying its function in the bus network.

Example: CAN PTT file

```
# ----------------------------------------------------------------------------
@FILE_VERSION  = 1.00
@PROTOCOL_NAME = can
# ---------------------------------------------------------------------------
# Labels for CAN protocol
#   Column order: Identifier type, Identifier value, Label
# ---------------------------------------------------------------------------
11,064h,Diag_Response
11,1E5h,EngineData
11,0A2h,Ignition_Info
11,1BCh,TP_Console
11,333h,ABSdata
11,313h,Door_Left
11,314h,Door_Right
```

```
29,01A54321h,Throttle
29,13A00FA2h,LightState
29,0630ABCDh,Engine_Status
29,03B1C002h,Airbag_Status
29,01234ABCh,NM_Gateway
# ----------------------------------------------------------------------------
```



For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- `BUS<m>:CAN:FRAMe<n>:SYMBol?` on page 1465

12.5.4 CAN and CAN FD Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The binary results of data bytes are displayed MSB first.

For CAN protocol, the endianness setting ("Transfer") is a trigger setting and not considered for decoding.

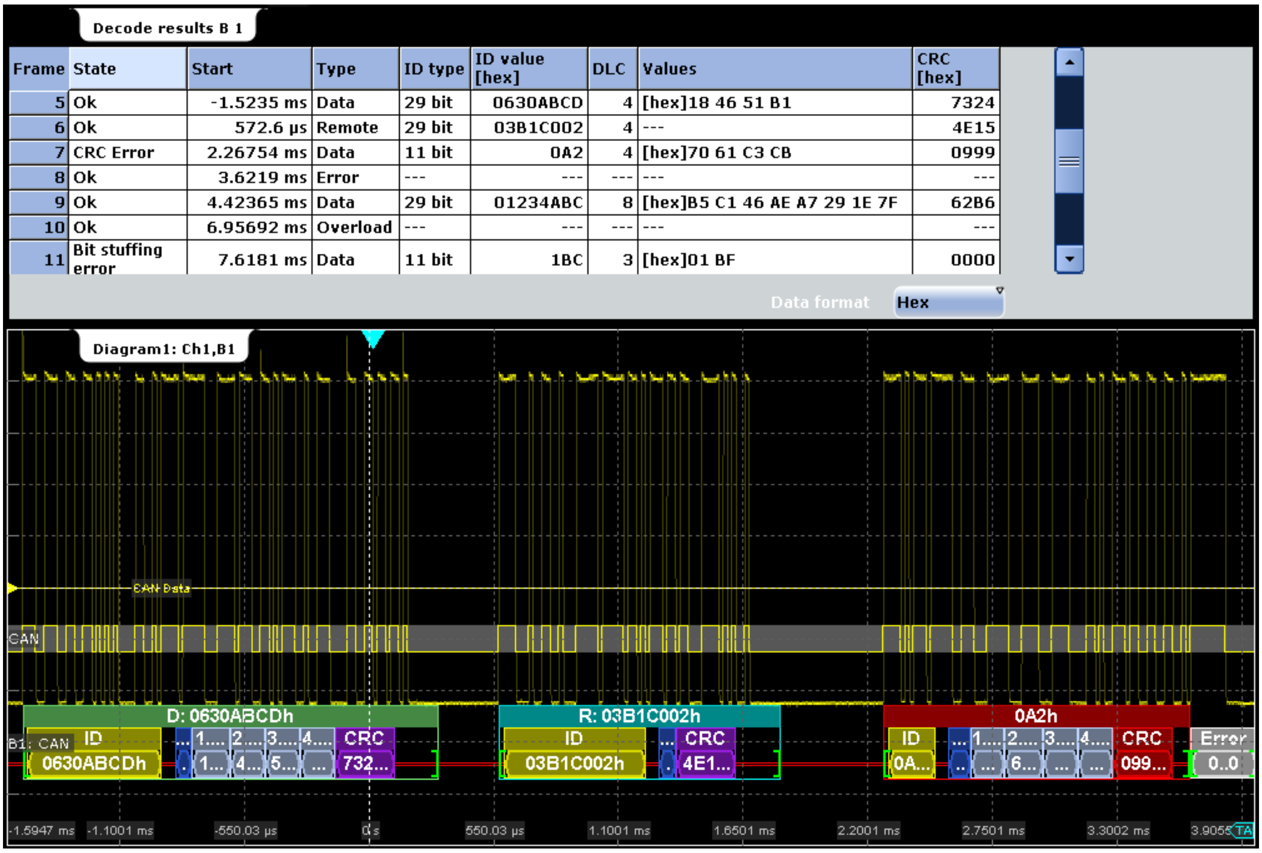


Figure 12-23: Decoded and binary CAN signal, and decode results

green brackets [...] = Start and end of frame
green frame header = Data frame, ok
cyan frame header = Remote frame, ok
magenta frame header = Overload frame, ok
red frame header = Frame contains an error
no frame header = Error frame
yellow = Identifier
blue = DLC
gray-blue = data
purple = CRC (checksum)
gray = Error frame
red = Error occurred

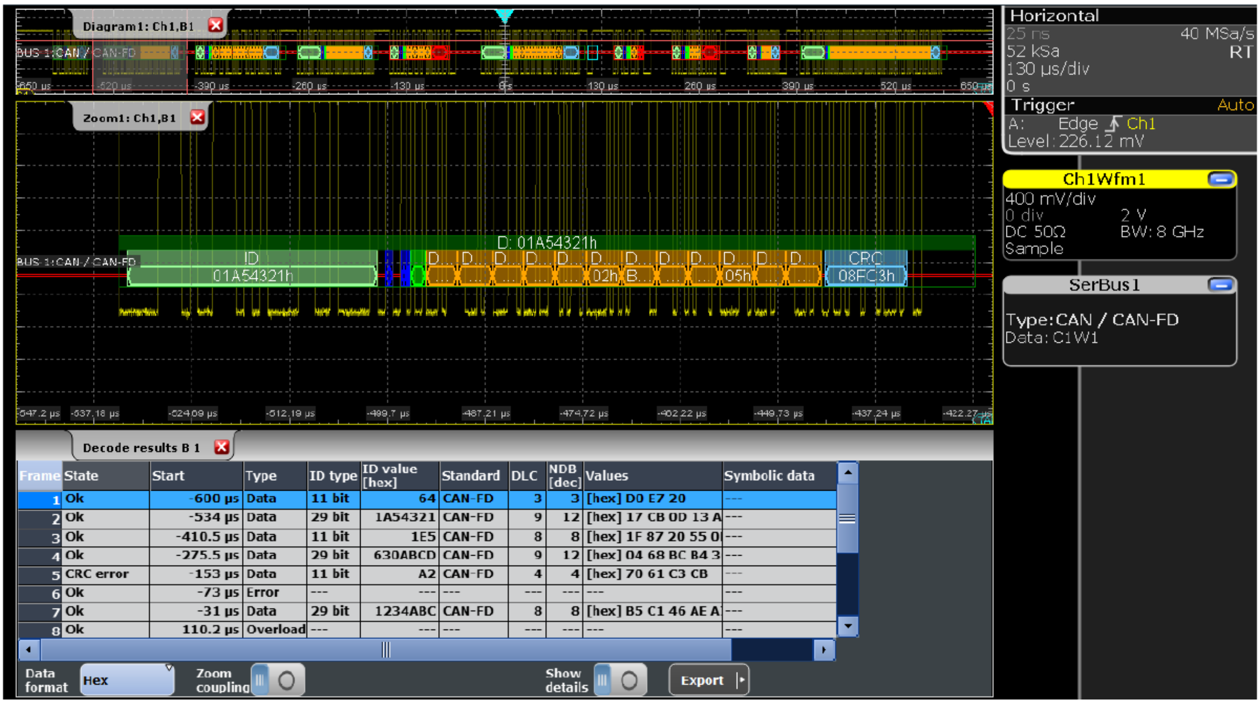


Figure 12-24: Decoded CAN FD signal with data type = CAN_L, arbitration rate = 1 Mbps and data rate = 2 Mbps

You can also load and apply industry-standard DBC files. The symbolic names from the file are applied to the display of the decoded data, see Chapter 12.5.6.2, "Symbolic Decode Waveform", on page 557.

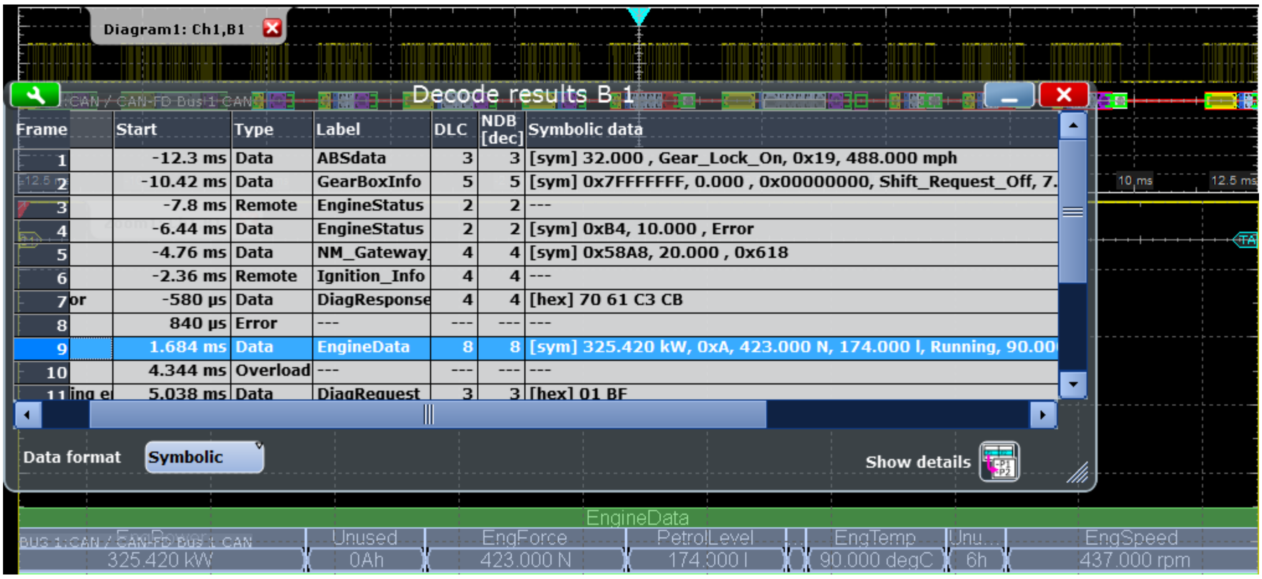
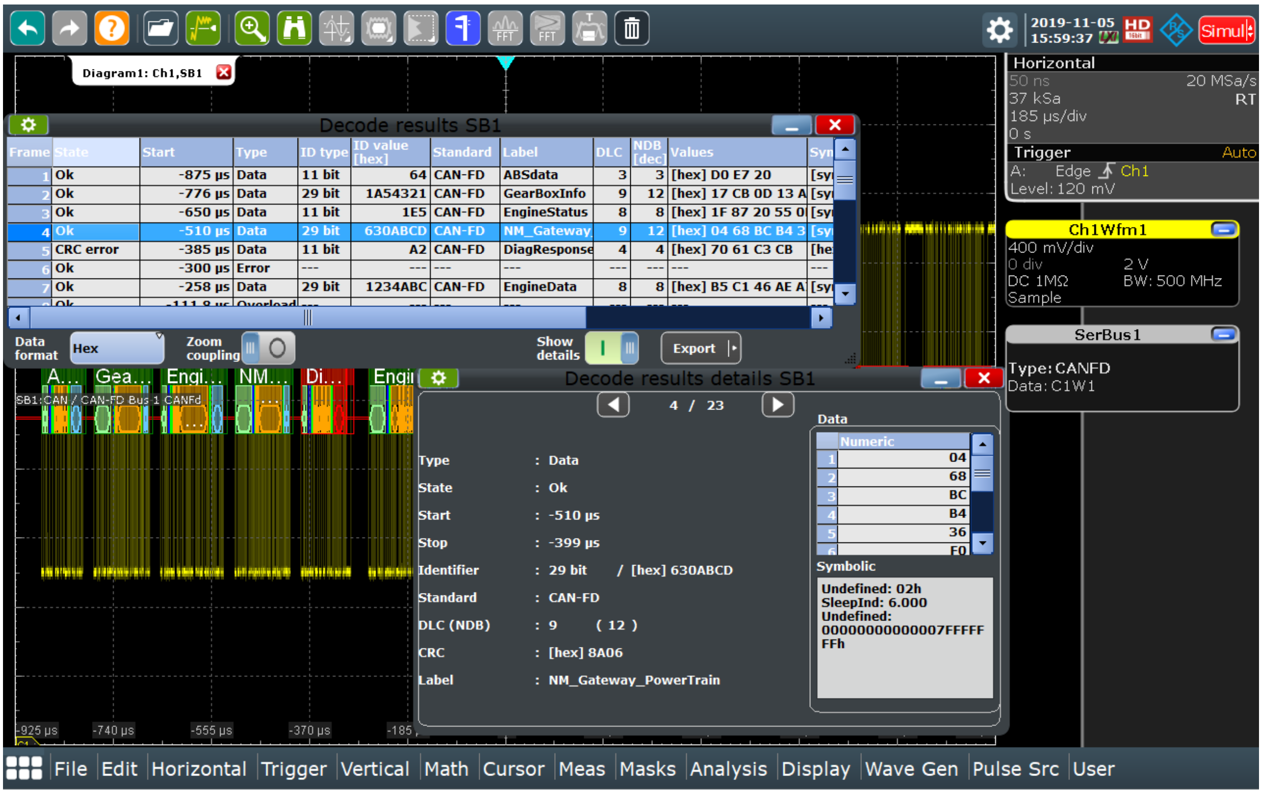


Figure 12-25: Decoded CAN signal with applied DBC file and "Symbolic" data in the result table

Table 12-6: Content of the "Decode results" table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Start	Time of frame start
Type	Frame type: Data, Remote, Error, or Overload
ID type	11-bit standard format or 29-bit extended format
ID value (hex)	Identifier value, hexadecimal value
Standard	Frame format, CAN or CAN FD. Only available in CAN FD option R&S RTE-K9.
Label	Symbolic label name defined in the label list
DLC	Data length code, coded number of data bytes
NDB	Actual number of data bytes
Values	Value of the data frame. The data format is selected below the table. Remote frames do not transmit data, therefore "- - -" is displayed.
Symbolic data	Values of the individual signals that are part of a message. The column is shown instead of the "Values" column, if a DBC file is loaded and the "Data format" is "Symbolic".
SC (dec)	Stuff count value, decimal value. Only available for CAN FD ISO signals, option R&S RTE-K9.
CRC (hex)	Value of the Cyclic Redundance Check (checksum), hexadecimal value
Form error cause	Reason of a form error if a form error occurred

You can also enable "Show details" in the decode table, to display a more detailed analysis of the selected frame. The details include the data, identifier, the label and the symbolic name.



Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.6.3, "Decode Results", on page 1463.

12.5.5 Search on Decoded CAN or CAN FD Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

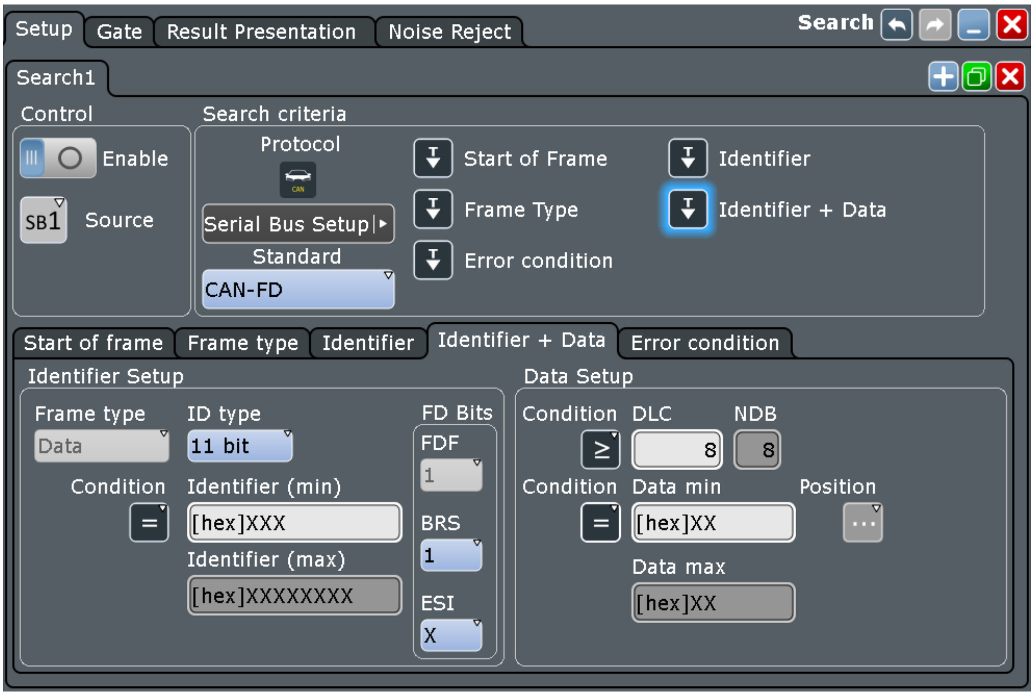
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

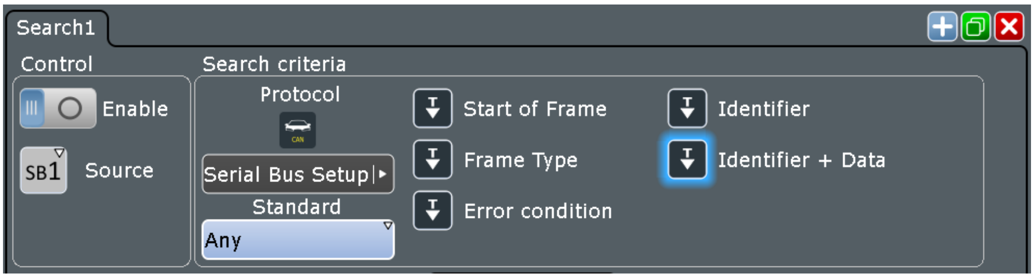
12.5.5.1 Search Settings

Access: [SEARCH] > "Setup" tab



Search Criteria
Sets one criterion or an AND-combination of criteria to be searched for. If more than one criterion is selected, all criteria must be fulfilled by a frame for it to be shown in the search results.

If a DBC label list file is applied, an additional criterion "Symbolic" is provided, see Chapter 12.5.6.3, "Symbolic Search", on page 558.



"Start of frame" Searches for the first edge of the dominant SOF bit (synchronization bit).

"Frame type" Searches for a specified frame type (data, remote, error, or overload). For data and remote frames, also the identifier format is considered.

For details, see:
- "Frame type" on page 549
- "ID type" on page 549

"Identifier" Searches for a specific message identifier or an identifier range. See "Identifier setup: Condition, Identifier min, Identifier max" on page 549.

"Identifier + Data"
Searches for a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern.
The identifier conditions are the same as for the "Identifier" search criteria, see "Identifier setup: Condition, Identifier min, Identifier max" on page 549. Data conditions are set with "Data setup: DLC, NDB, Condition, Data min, Data max" on page 550.

"Error condition"
Identifies various errors in the frame, see "Error conditions" on page 534.

"Symbolic" The "Symbolic" search criteria is available if a DBC label list file is loaded and applied. It allows you to search for specific data messages, or a signal and its value that appears inside the message.
Symbolic search is an alternative to the other search criteria, you can either search for symbolic values or for an AND-combination of the other 5 criteria.
For details, see Chapter 12.5.6.3, "Symbolic Search", on page 558.

Remote command:
`SEARch:TRIGger:CAN[:SSOFrame]` on page 1472
`SEARch:TRIGger:CAN:SFTYpe` on page 1472
`SEARch:TRIGger:CAN:SFIDentifier` on page 1472

SEARch:TRIGger:CAN:SIDData on page 1473
SEARch:TRIGger:CAN:SERRor on page 1473
SEARch:TRIGger:CAN:SSYMbolic on page 1489

Standard

Selects the CAN standard: "CAN", "CAN FD", or "Any".

The setting is available in CAN FD option R&S RTE-K9.

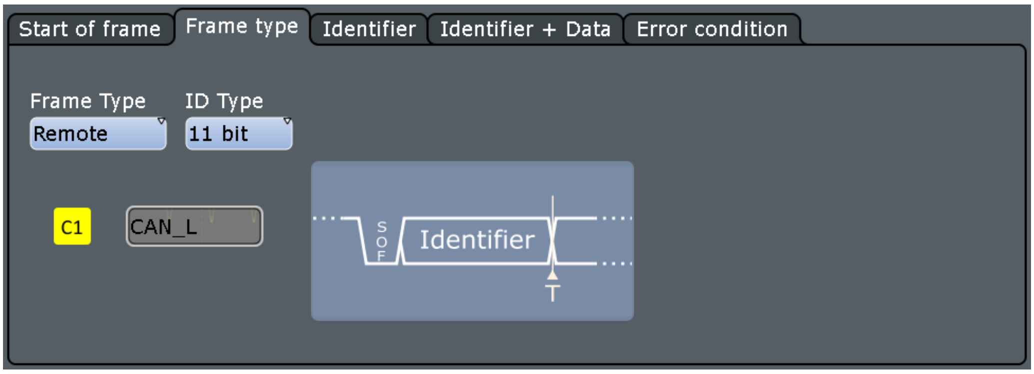
Use "Any" to search for both CAN and CAN-FD frames. In this case, the search configuration provides all possible settings, for CAN as well as for CAN FD.

Remote command:
SEARch:TRIGger:CAN:FDATa:STANdard on page 1473

Frame type

Selects the frame type.

Remote frames are only available in the CAN protocol.



The frame types are the same as in the CAN trigger setup, see "Frame type" on page 530.

Remote command:
SEARch:TRIGger:CAN:FTYPe on page 1473

ID type

Selects the length of the identifier.

"11 bit" Identifier length of the CAN base frame format. The instrument triggers on the sample point of the IDE bit.

"29 bit" Identifier length of the CAN extended frame format. The instrument triggers on the sample point of the RTR bit.

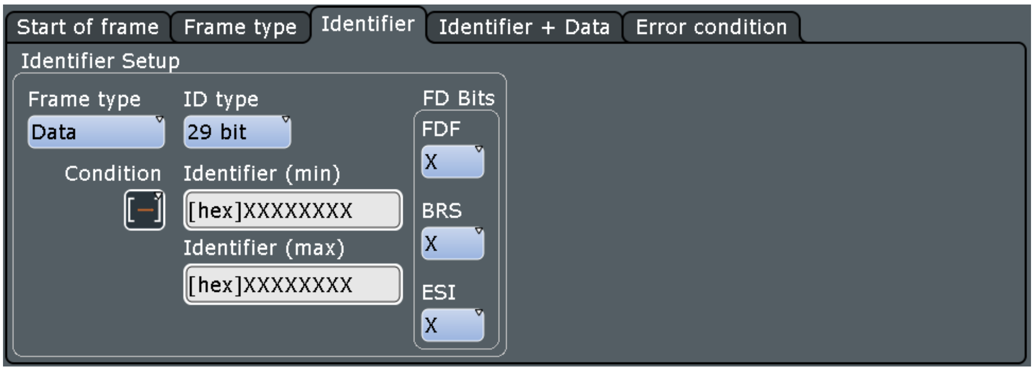
"Any" The ID type is not relevant. If the trigger type is "Identifier + Data", set the "ID type" to "Any" if you want to search only for data.

Remote command:
SEARch:TRIGger:CAN:ITYPe on page 1474

Identifier setup: Condition, Identifier min, Identifier max

The identifier setup consists mainly of the condition and one or two identifier patterns. Additionally, ID type and frame type may qualify the identifier.

The identifier setup settings are the same as in the CAN trigger setup, see "Identifier setup: Condition, Identifier min, Identifier max" on page 531.



FD bits: see "FD bits" on page 550.

Remote command:
`SEARch:TRIGger:CAN:ICONdition` on page 1474
`SEARch:TRIGger:CAN:IMIN` on page 1475
`SEARch:TRIGger:CAN:IMAX` on page 1474

FD bits
For standard settings "CAN FD" and "Any", you can search for CAN FD-specific bits.

For details, see "FD bits" on page 531.

The setting is available in CAN FD option R&S RTE-K9.

Remote command:
`SEARch:TRIGger:CAN:FDATa[:FDF]` on page 1479
`SEARch:TRIGger:CAN:FDATa:BRS` on page 1479
`SEARch:TRIGger:CAN:FDATa:ESI` on page 1480

Data setup: DLC, NDB, Condition, Data min, Data max
The data setup consists of the number of bytes, the condition, and one or two data patterns.

The data setup settings are the same as in the CAN trigger setup, see "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 532.

The data condition setting is also used for symbolic signal search, see Chapter 12.5.6.3, "Symbolic Search", on page 558.

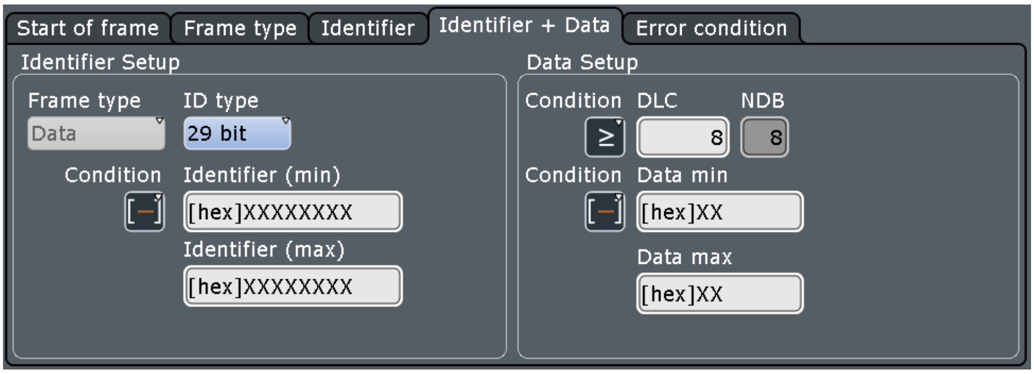


Figure 12-26: Identifier + Data search setup for CAN signals

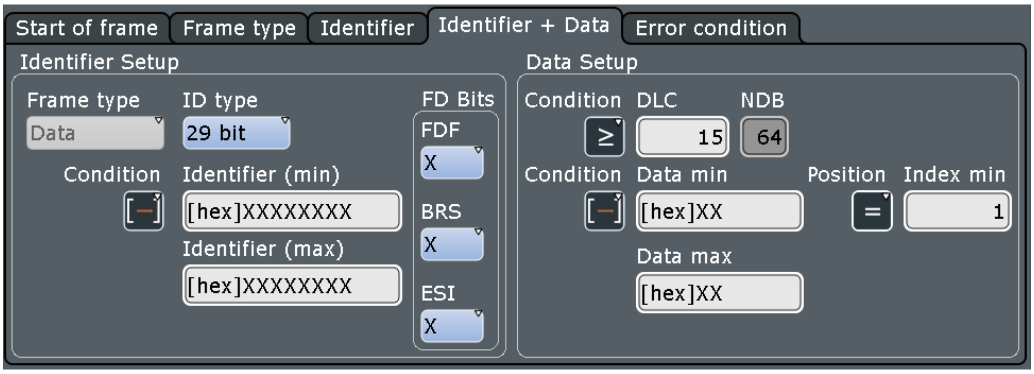


Figure 12-27: Identifier + Data search setup for CAN FD signals

Remote command:
`SEARch:TRIGger:CAN:DCONdition` on page 1475
`SEARch:TRIGger:CAN:DMIN` on page 1475
`SEARch:TRIGger:CAN:DMAX` on page 1476
`SEARch:TRIGger:CAN:DLCCondition` on page 1476
`SEARch:TRIGger:CAN:DLC` on page 1476
`SEARch:RESult:CAN:FRAMe<m>:NDBYtes?` on page 1476

Data position

The data position sets the location in the data field where the instrument looks for the specified data pattern.

The setting is available in CAN FD option R&S RTE-K9.

The position can be defined if the data field of the frame is longer than 8 bytes - if DLC ≥9.

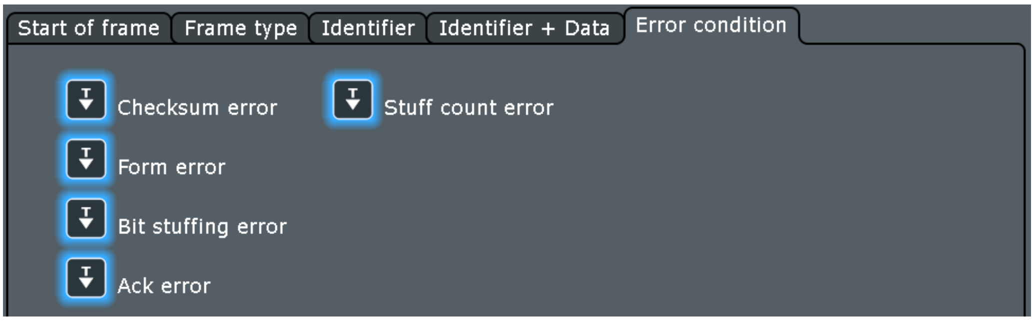
For details, see "Data position" on page 533.

Remote command:
`SEARch:TRIGger:CAN:FDATa:DPOPerator` on page 1477
`SEARch:TRIGger:CAN:FDATa:DPOSition` on page 1477
`SEARch:TRIGger:CAN:FDATa:DPTO` on page 1478

Error Condition

Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the CAN trigger setup, see "Error conditions" on page 534.



Remote command:
SEARch:TRIGger:CAN:CRCerror on page 1478
SEARch:TRIGger:CAN:BITSterror on page 1478
SEARch:TRIGger:CAN:FORMerror on page 1479
SEARch:TRIGger:CAN:ACKerror on page 1478
SEARch:TRIGger:CAN:FDATa:SCERror on page 1479

12.5.5.2 Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- SEARch:RESult:CAN:FCOunt? on page 1481
- SEARch:RESult:CAN:FRAMe<m>:STATus? on page 1485
- SEARch:RESult:CAN:FRAMe<m>:FERCause? on page 1483
- SEARch:RESult:CAN:FRAMe<m>:ACKState? on page 1482
- SEARch:RESult:CAN:FRAMe<m>:ACKValue? on page 1481
- SEARch:RESult:CAN:FRAMe<m>:BSEPosition? on page 1481
- SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe? on page 1481
- SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue? on page 1482
- SEARch:RESult:CAN:FRAMe<m>:CSSTate? on page 1482
- SEARch:RESult:CAN:FRAMe<m>:CSValue? on page 1482

- SEARch:RESult:CAN:FRAMe<m>:DATA? on page 1483
- SEARch:RESult:CAN:FRAMe<m>:DLCState? on page 1482
- SEARch:RESult:CAN:FRAMe<m>:DLCValue? on page 1483
- SEARch:RESult:CAN:FRAMe<m>:IDSTate? on page 1482
- SEARch:RESult:CAN:FRAMe<m>:IDTYpe? on page 1484
- SEARch:RESult:CAN:FRAMe<m>:IDValue? on page 1484
- SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard? on page 1484
- SEARch:RESult:CAN:FRAMe<m>:STARt? on page 1485
- SEARch:RESult:CAN:FRAMe<m>:STOP? on page 1485
- SEARch:RESult:CAN:FRAMe<m>:SYMBol? on page 1486
- SEARch:RESult:CAN:FRAMe<m>:TYPE? on page 1486

12.5.5.3 Searching CAN FD Data

The "Identifier + Data" search supports the search for data bytes of specific value at a specific location in the data field of a frame. The "Data Pattern" field provides 8 data bytes to define the pattern. For data fields longer than 8 byte, you can define the position where the specified pattern starts.

The following example demonstrates how the data pattern and data position is defined.

To set up the search

1. Set the "Source", the signal to be searched: "SerBus". Select the bus that is configured for CAN FD.
2. Set the search criteria:
   a) Select the standard: "CAN FD".
   b) Select the search type: "Identifier + Data".
3. In this example, the identifier does not matter. Set the "ID type = Any".
4. Define the data setup as described in the example.

Example: Searching for a specific byte anywhere in the frame

The CAN FD frame has 8 or more data bytes, containing at least one data byte with value = CB anywhere in the data field.

- Set "DLC ≥ 8".
- Set the data pattern: "= CB".

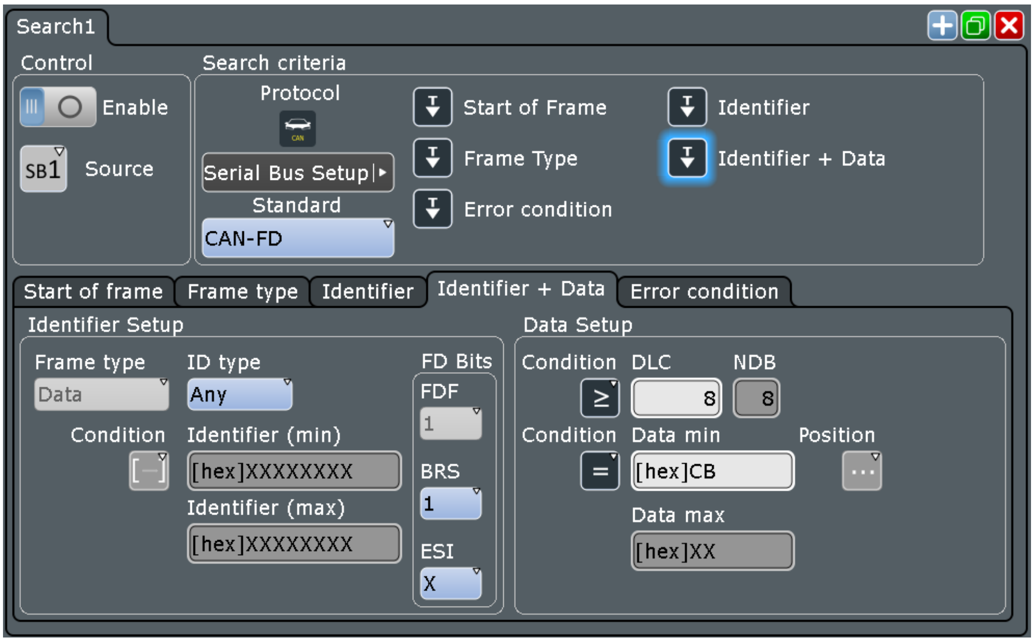


Figure 12-28: Search setup to find all data bytes with value = CB

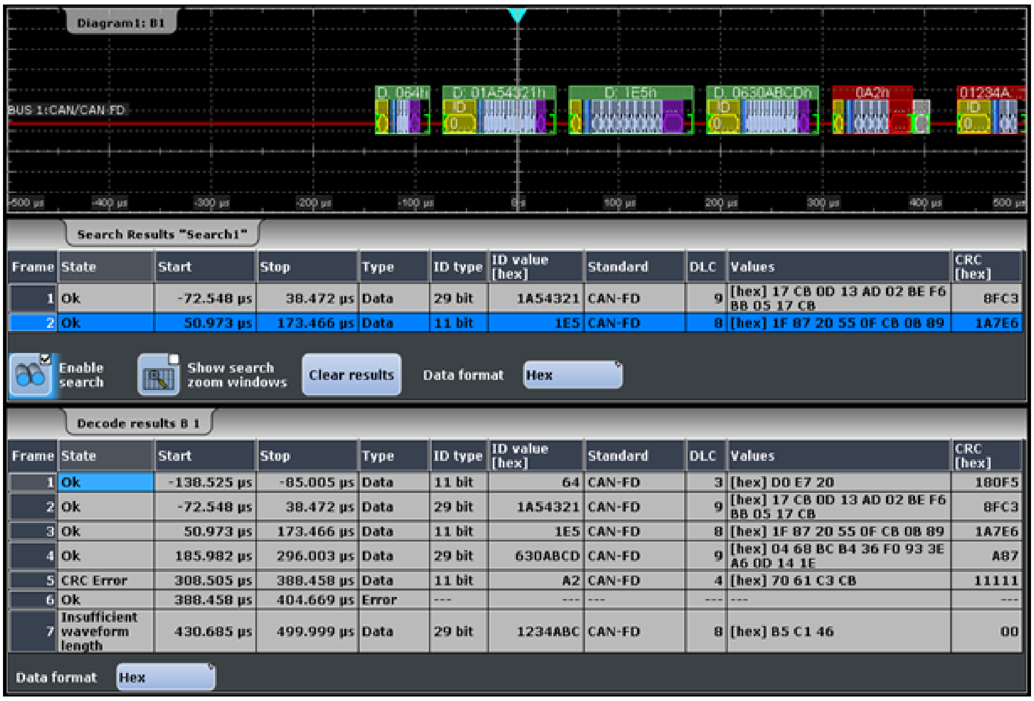


Figure 12-29: Search result

12.5.6 Symbolic Trigger, Decode and Search

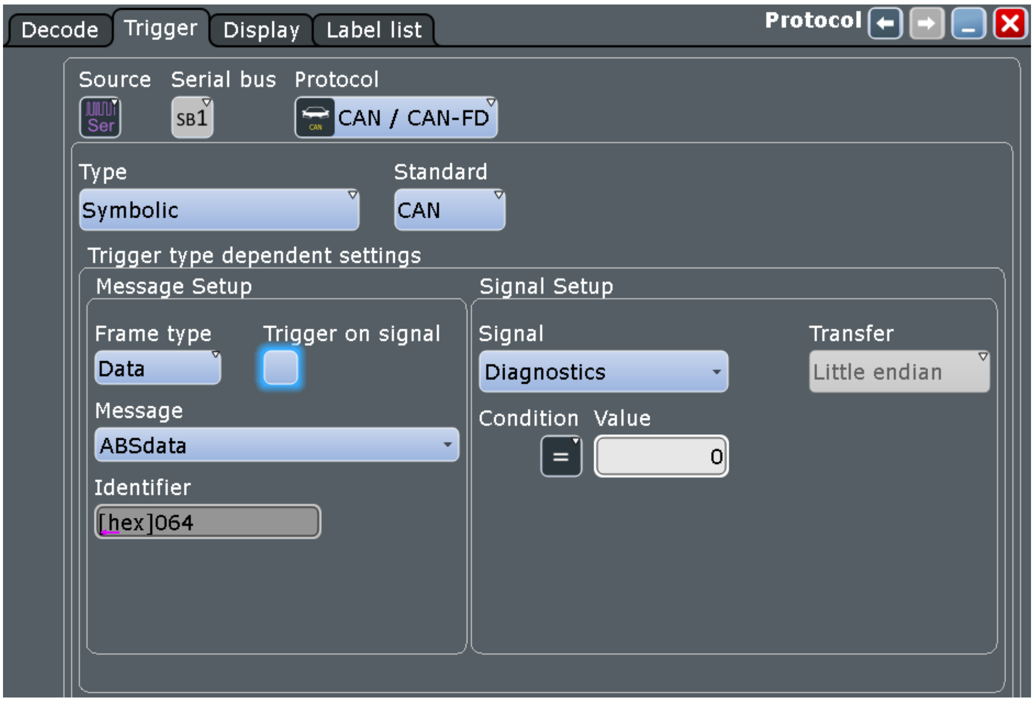
The R&S RTE can read and apply industry-standard DBC files to the decoded signal and provides settings for symbolic triggering and symbolic search.

For a description of DBC files, see Chapter 12.5.3.1, "DBC Files for CAN / CAN FD", on page 540.

12.5.6.1 Symbolic Trigger

The "Symbolic" trigger type is available if a DBC label list file is loaded and applied, see Chapter 12.5.3, "CAN / CAN FD Label List", on page 540. It allows you to trigger on a specific data message, or a signal and its value that appears inside the message.

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = CAN/ CAN-FD" and "Type" = "Symbolic"



The "Frame type" is automatically set to "Data", and the "Identifier" is shown for information.

Specific settings for the symbolic trigger are:

Message..556
Trigger on signal..556
Signal..556
Condition..557
Value, Value min..557
Value max..557

Message
Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

Remote command:
TRIGger<m>:CAN:SYMBolic:MSGValue on page 1487

Trigger on signal
Enables the trigger on a specific signal value that is part of the selected message.

Remote command:
TRIGger<m>:CAN:SYMBolic:TSIGnals on page 1487

Signal
Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

Remote command:
`TRIGger<m>:CAN:SYMBolic:SIGValue` on page 1487

Condition
Sets the operator to set a specific data pattern or symbolic value ("Equal" or "Not equal") or a data range.

Remote command:
`TRIGger<m>:CAN:DCONdition` on page 1459

Value, Value min
Defines the data pattern or selects a symbolic data value.

Remote command:
`TRIGger<m>:CAN:SYMBolic:DMIN` on page 1487
`TRIGger<m>:CAN:SYMBolic:SGEValue` on page 1488

Value max
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CAN:SYMBolic:DMAX` on page 1487

12.5.6.2 Symbolic Decode Waveform

If a DBC file is applied, the symbolic names from the file are applied to the display of the decoded data. The result table lists the signal values and units in the "Symbolic Data" column, and the comb display shows the signal names in addition to the signal values and units.

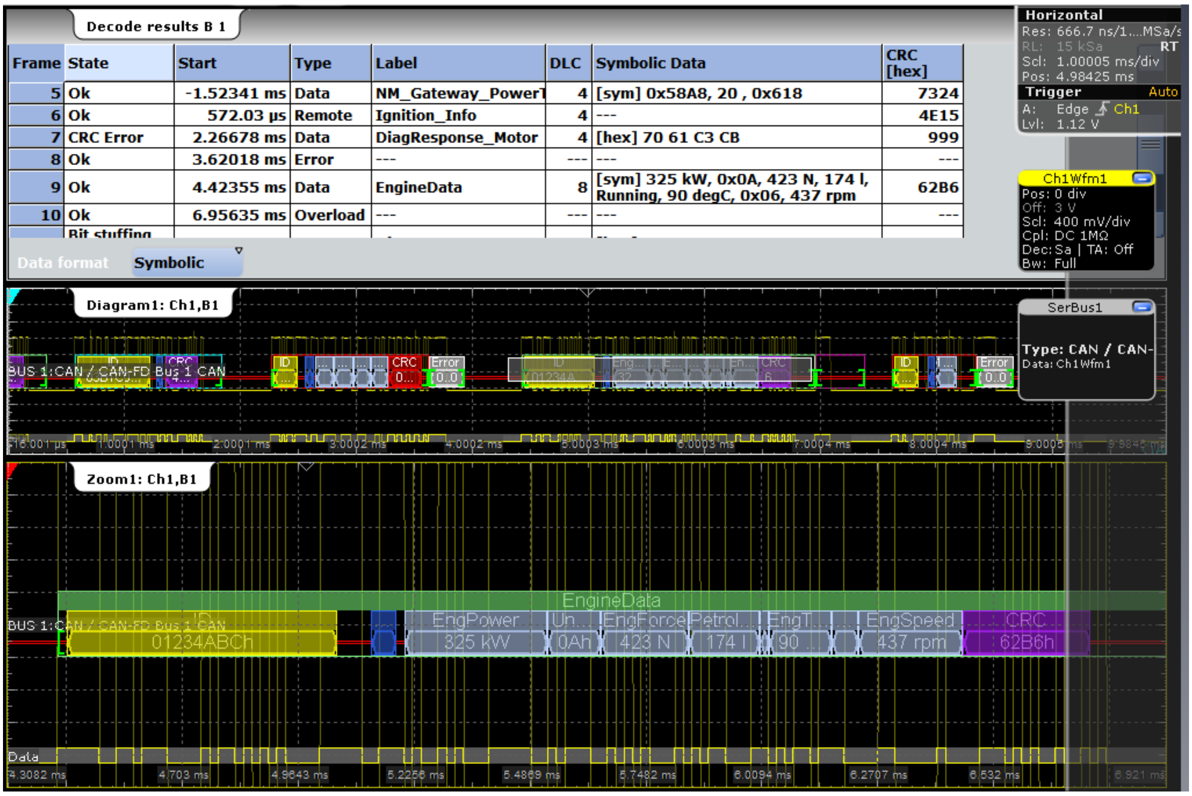


Figure 12-30: Result table and decoded CAN signal with applied DBC file and zoom on EngineData message

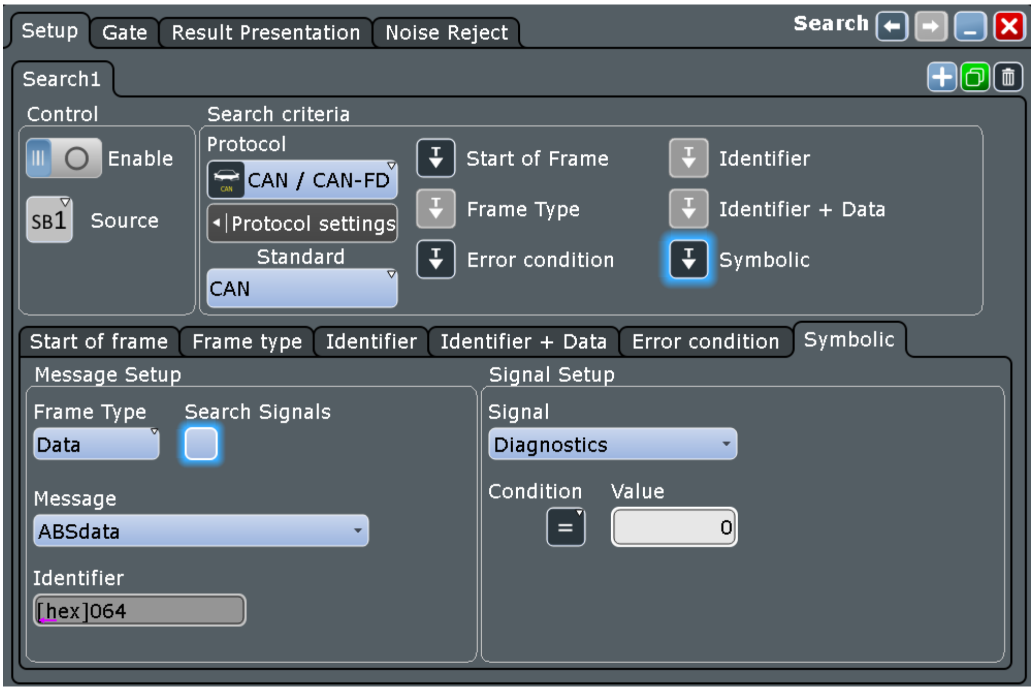
Remote command:

- `BUS<m>:CAN:FRAMe<n>:SDATa?` on page 1488
- `BUS<m>:CAN:FRAMe<n>:SDEXport?` on page 1470

12.5.6.3 Symbolic Search

Access: [SEARCH] > "Setup" tab > "Symbolic" = on

If a DBC file is applied, the symbolic search for messages and signal, which are defined in the DBC file, is available. Symbolic search is an alternative to the other search criteria, you can either search for symbolic values or for an AND-combination of the other 5 criteria.



If symbolic search is active, the "Frame type" is automatically set to "Data", and the "Identifier" is shown for information.

Symbolic
Enables the symbolic search and disables all other search criteria.

Remote command:
`SEARch:TRIGger:CAN:SSYMbolic` on page 1489

Message
Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:MSGValue` on page 1489

Search signals
Enables the search for a specific signal value that is part of the selected message.

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:SSIGnals` on page 1490

Signal
Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

Remote command:
SEARch:TRIGger:CAN:SYMBolic:SIGValue on page 1490

Condition
Sets the operator to set a specific data pattern or symbolic value ("Equal" or "Not equal") or a data range.

This condition is also used to search for data, see "Data setup: DLC, NDB, Condition, Data min, Data max" on page 550.

Remote command:
SEARch:TRIGger:CAN:DCONdition on page 1475

Value, Value (min)
Defines the data pattern or selects a symbolic data value.

Remote command:
SEARch:TRIGger:CAN:SYMBolic:DMIN on page 1490
SEARch:TRIGger:CAN:SYMBolic:SGEValue on page 1491

Value (max)
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:CAN:SYMBolic:DMAX on page 1490

12.5.6.4 Symbolic Search Results

If a DBC file is applied, you can search for symbolic messages and signals as described in Chapter 12.5.6.3, "Symbolic Search", on page 558. As ususal, the search results are shown in a table. You can enable the search zoom window to view the frame with the selected result in more detail.

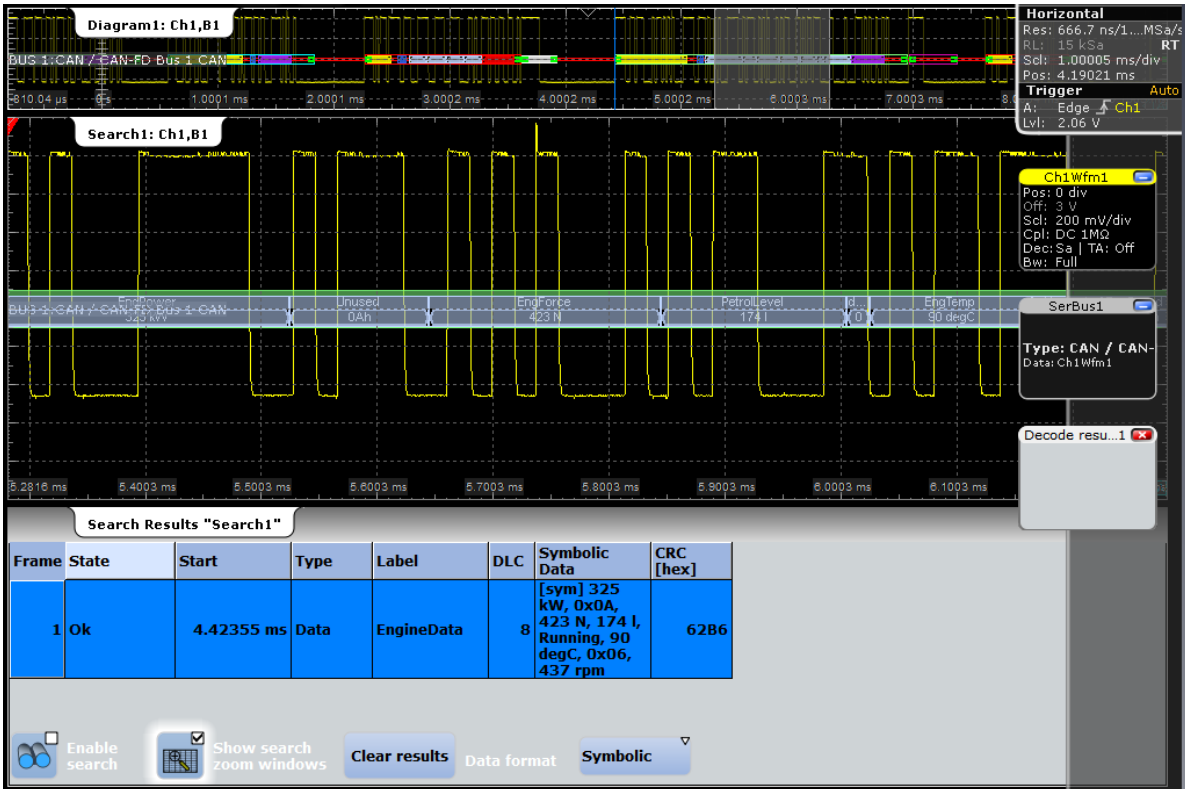


Figure 12-31: Search results table and search zoom window. Search for signal "IdleRunning" with value "Running" (bit value = 0)

The figure shows the result of a search for an "EngineData" message and the signal "IdleRunning = Running" inside the message. The result marker (blue line) is set to the start of the frame that fulfills the search condition. The search zoom window Search1 is active. It has been moved to the right until the "IdleRunning" bit with value 0 is visible in the zoom.

Remote command:

- `SEARch:RESult:CAN:FRAMe<m>:SDATa?` on page 1489

12.6 LIN (Option R&S RTE-K3)

The Local Interconnect Network (LIN) is a simple, low-cost bus system used within automotive network architectures. LIN is usually a subnetwork of a CAN bus. The primary purpose of LIN is the integration of uncritical sensors and actuators with low-bandwidth requirements. Common applications in a motor vehicle are the control of doors, windows, wing mirrors, and wipers.

12.6.1 The LIN Protocol

This chapter provides an overview of protocol characteristics, frame format, identifiers and trigger possibilities. For detailed information, order the LIN specification on http://www.lin-subbus.org/ (free of charge).

LIN characteristics

Main characteristics of LIN are:

- Single-wire serial communications protocol, based on the UART byte-word interface
- Single master, multiple slaves - usually up to 12 nodes
- Master-controlled communication: master coordinates communication with the LIN schedule and sends identifier to the slaves
- Synchronization mechanism for clock recovery by slave nodes without crystal or ceramics resonator

The R&S RTE supports several versions of the LIN standard: v1.3, v2.0, v2.1 and the American SAE J2602.

Data transfer

Basic communication concept of LIN:

- Communication in an active LIN network is always initiated by the master.
- Master sends a message header including the synchronization break, the synchronization byte, and the message identifier.
- The identified node sends the message response: one to eight data bytes and one checksum byte.
- Header and response form the message frame.

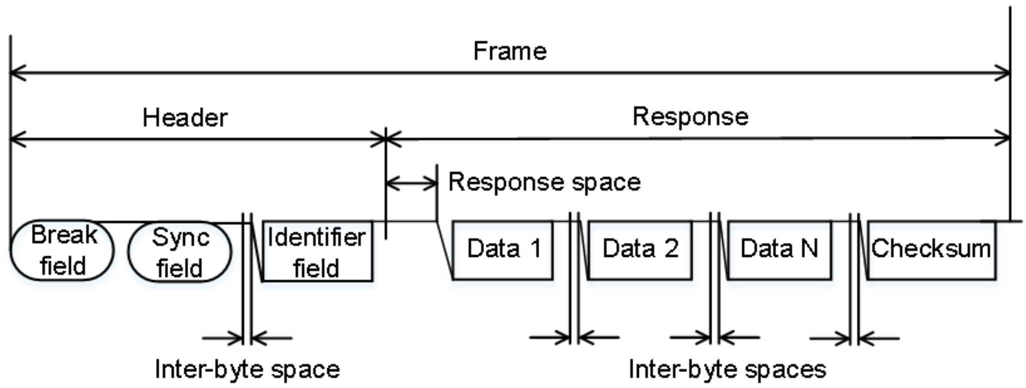


Figure 12-32: LIN frame with header and response

The data is transmitted in bytes using the UART byte-word interface without the parity bit. Each byte consists of a start bit, 8 bits and a stop bit.

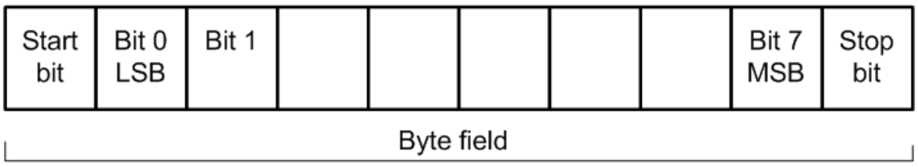


Figure 12-33: Structure of a byte field

Data bytes are transmitted LSB first.

The identifier byte consists of 6 bits for the frame identifier and two parity bits. This combination is known as protected identifier.

Trigger

The R&S RTE can trigger on various parts of LIN frames. The data line must be connected to an input channel, triggering on math and reference waveforms is not possible.

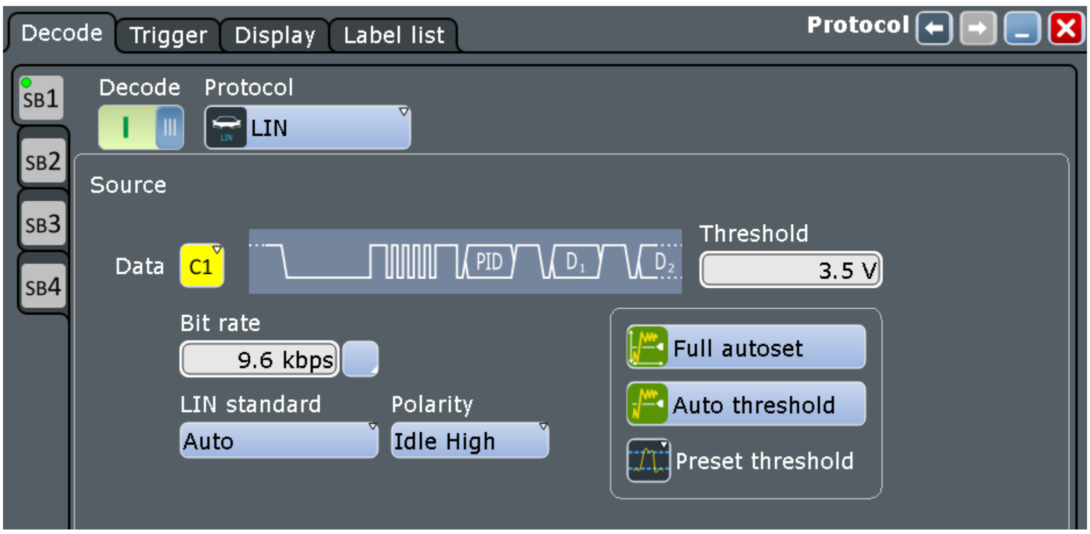
You can trigger on:

- Frame start (synchronization field)
- Specific slave identifier or identifier range
- Data pattern in the message
- Wake up signal
- Checksum error (error in data), parity error (error in identifier)

12.6.2 LIN Configuration

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *LIN*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Data
Sets the source waveform of the data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:LIN:DATA:SOURce` on page 1491

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the Polarity.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Full autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
  See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Preset thresholds"
  Selects the default threshold voltage for various signal technologies from a list. The value is set to "Manual" if the threshold was set with "Auto threshold", or was entered directly.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:LIN:DATA:THReshold` on page 1492
`BUS<m>:LIN:TECHnology` on page 1492
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Bit rate
Sets the number of transmitted bits per second. The maximum bit rate for LIN is 20 kbit/s.

To select a bit rate from list of predefined values, tap the icon beside the "Bit rate" field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

If the "LIN standard" is "J2602", the bit rate is 10.417 kbit/s and cannot be changed.

Remote command:
BUS<m>:LIN:BITRate on page 1493

LIN standard
Selects the version of the LIN standard that is used in the DUT. The setting mainly defines the checksum version used during decoding.

The most common version is LIN 2.x. For mixed networks, or if the standard is unknown, set the LIN standard to "Auto".

Remote command:
BUS<m>:LIN:STANdard on page 1493

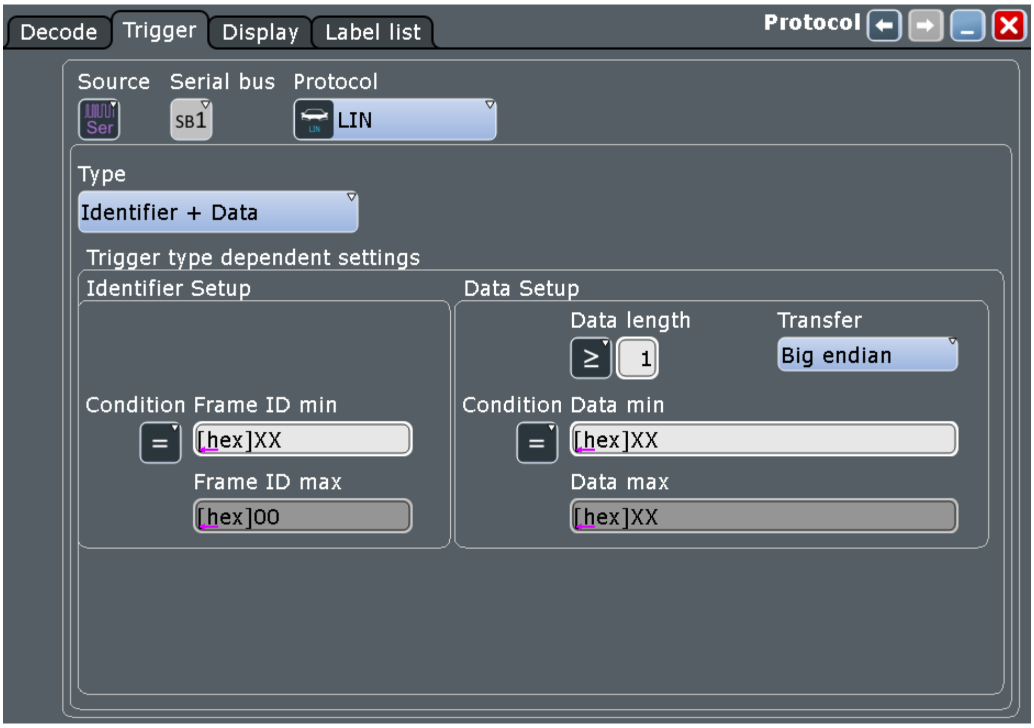
Polarity
Defines the idle state of the bus. The idle state is the recessive state and corresponds to a logic 1.

Remote command:
BUS<m>:LIN:POLarity on page 1493

12.6.3 LIN Trigger

12.6.3.1 LIN Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = LIN"



Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Trigger type
Selects the trigger type for LIN analysis.

"Start of frame (Sync)"
Triggers on the stop bit of the sync field.

"Identifier"
Sets the trigger to one specific identifier or an identifier range. Enter only the 6-bit identifier without parity bits, not the protected identifier.
Description of trigger type specific settings: "Identifier setup: Condition, Frame ID min, Frame ID max" on page 566.

"Identifier OR"
Sets the trigger to a combination of up to four identifiers.
Description of trigger type specific settings: "Identifier OR setup: Monitor, Frame ID" on page 567

"Identifier + Data"
Sets the trigger to a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern.
The identifier conditions are the same as for the "Identifier" trigger type, see Identifier setup: Condition, Frame ID min, Frame ID max.
Data conditions are set with Data setup: Data length, Transfer, Condition, Data min, Data max.

"Wakeup frame"
Triggers after a wakeup frame.

"Error condition"
Identifies various errors in the frame, see "Error conditions" on page 568.

Remote command:
`TRIGger<m>:LIN:TYPE` on page 1494

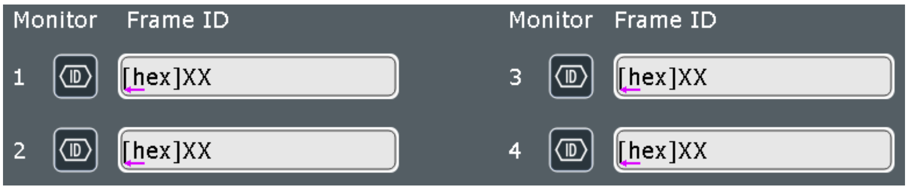
Identifier setup: Condition, Frame ID min, Frame ID max
The identifier setup consists of the condition and one or two identifier pattern.

"Condition"	Defines the operator to set a specific identifier ("Equal" or "Not equal") or an identifier range.
"Frame ID min / Frame ID"	Defines the bit pattern of the slave identifier. Enter only the 6-bit identifier without parity bits, not the protected identifier. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"Frame ID max"	The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:LIN:ICONdition on page 1495
TRIGger<m>:LIN:IMIN on page 1495
TRIGger<m>:LIN:IMAX on page 1495

Identifier OR setup: Monitor, Frame ID

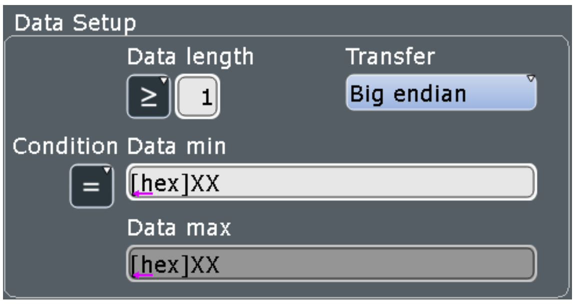
Sets the trigger to a combination of up to four identifiers. Enter the patterns in the "Frame ID" fields. In binary and hex format, characters 1, 0, and X (do not care) are allowed. For each identifier pattern to be triggered on, enable "Monitor".



Remote command:
TRIGger<m>:LIN:IDOR<n>:ENABle on page 1497
TRIGger<m>:LIN:IDOR<n>[:VALue] on page 1498

Data setup: Data length, Transfer, Condition, Data min, Data max

The data setup consists of the transfer direction, the number of bytes, the condition, and one or two data patterns.



"Transfer" Sets the byte order (endianness) of the data transfer. With "Big endian", the data is analyzed and evaluated in the order of reception. With "Little endian", the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.
According to the standard, LIN data is transmitted in little endian transfer order. The "Little endian" setting allows you to enter the required data word directly into "Data min", and the instrument triggers correctly.

"Data length" Sets the length of the bit pattern to be found, in bytes.

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.
Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.

"Condition" Sets the operator to define a specific data pattern ("Equal" or "Not equal") or an data range.

"Data min" Defines the data pattern. The pattern length is adjusted to the data length setting (and vice versa), maximum is 8 bytes.
Enter the pattern MSB first and with big endian byte order, and set the correct "Transfer" direction. The data is compared byte by byte.
In binary format, use the following characters: 1; 0; or X (do not care).
The use of X is restricted to the operators "Equal" and "Not equal".

"Data max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:LIN:BORDer` on page 1496
`TRIGger<m>:LIN:DLECondition` on page 1497
`TRIGger<m>:LIN:DLENgth` on page 1497
`TRIGger<m>:LIN:DCONdition` on page 1495
`TRIGger<m>:LIN:DMIN` on page 1496
`TRIGger<m>:LIN:DMAX` on page 1496

Error conditions
Triggers if one or more of the following errors occur:

- Checksum error
  The checksum verifies the correct data transmission. It is the last byte of the frame response. The checksum includes not only the data but also the protected identifier (PID). To identify checksum errors caused by data, additional settings are required: Enter the bit pattern of the slave identifier ("Frame ID"), the number of data bytes ("Data length"), and select the used "LIN standard". See also: "LIN standard" on page 565.
- Identifier parity error

Parity bits are the bits 6 and 7 of the identifier. They verify the correct transmission of the identifier.
- Sync error
  Synchronization error

Remote command:
TRIGger<m>:LIN:CHKSerror on page 1498
TRIGger<m>:LIN:ERRPattern on page 1499
TRIGger<m>:LIN:CRCDatalen on page 1499
TRIGger<m>:LIN:STANdard on page 1499
TRIGger<m>:LIN:IPERror on page 1498
TRIGger<m>:LIN:SYERror on page 1498

12.6.3.2 Triggering on LIN Signals

Prerequisites: An LIN bus is configured, see Chapter 12.2.2.2, "Configuring I²C Protocol", on page 486.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to LIN.
5. Select the "Trigger type".
6. For more complex trigger types, enter the address and/or data conditions: address, identifier, frame ID.
   For details, see Chapter 12.6.3.1, "LIN Trigger Settings", on page 565.

12.6.4 LIN Label List

Label lists are protocol-specific. Label lists for LIN are available in CSV and PTT format.

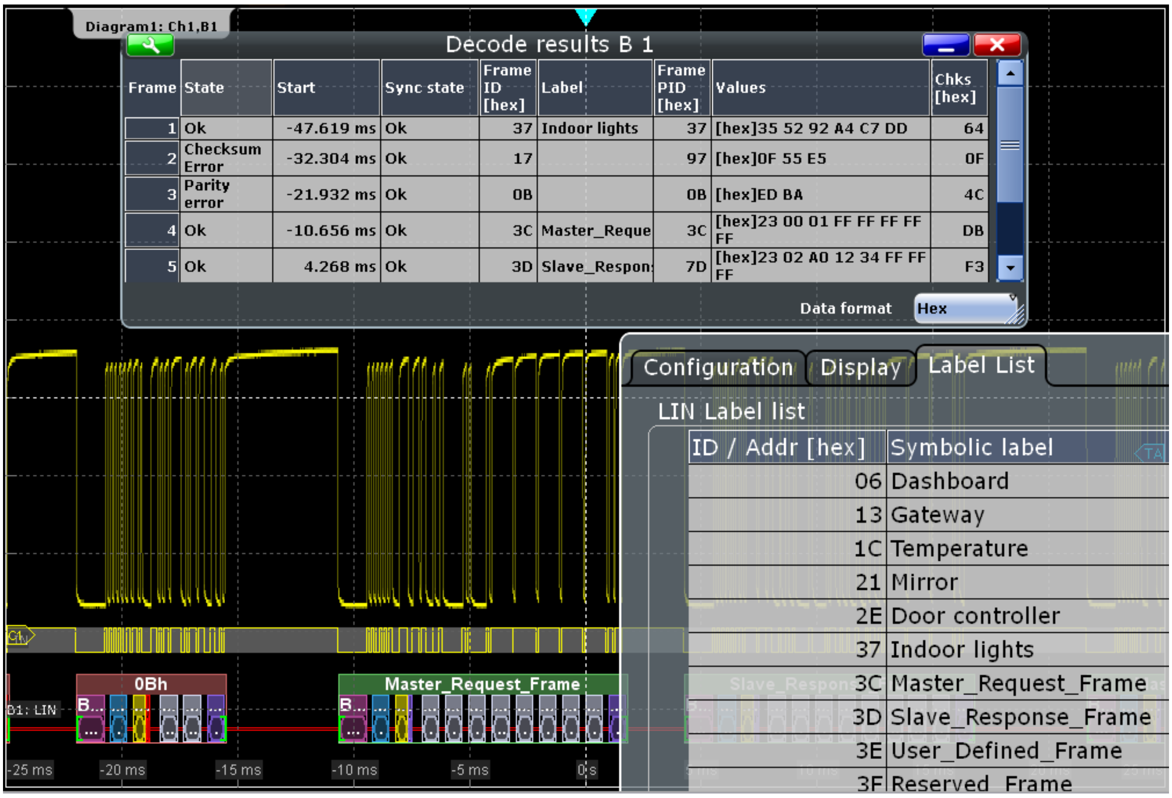
You can also apply an existing LIN description file.

A LIN label file contains two values for each identifier:
- Identifier value
- Symbolic name for the identifier

Example of a LIN PTT file

```
# ----------------------------------------------------------------------------
@FILE_VERSION = 1.0
@PROTOCOL_NAME = lin
# ----------------------------------------------------------------------------
# Labels for LIN protocol
#   Column order: Identifier, Label
# ----------------------------------------------------------------------------
# Labels for standard addresses
0x3F,Temperature
1Ch,Left brake
20h,Right brake
# Following ID is provided as integer
33,Mirror
0x37,Indoor lights
# Labels for reserved addresses
0x3C,Master_Request_Frame
0x3D,Slave_Response_Frame
# ----------------------------------------------------------------------------
```

For general information on label lists, see Chapter 12.1.4, "Label Lists", on page 478.



For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- `BUS<m>:LIN:FRAMe<n>:SYMBol?` on page 1502

The LIN description file (ldf) contains information about the assignment of the LIN frames to the nodes. The format and syntax are defined in the LIN specification.

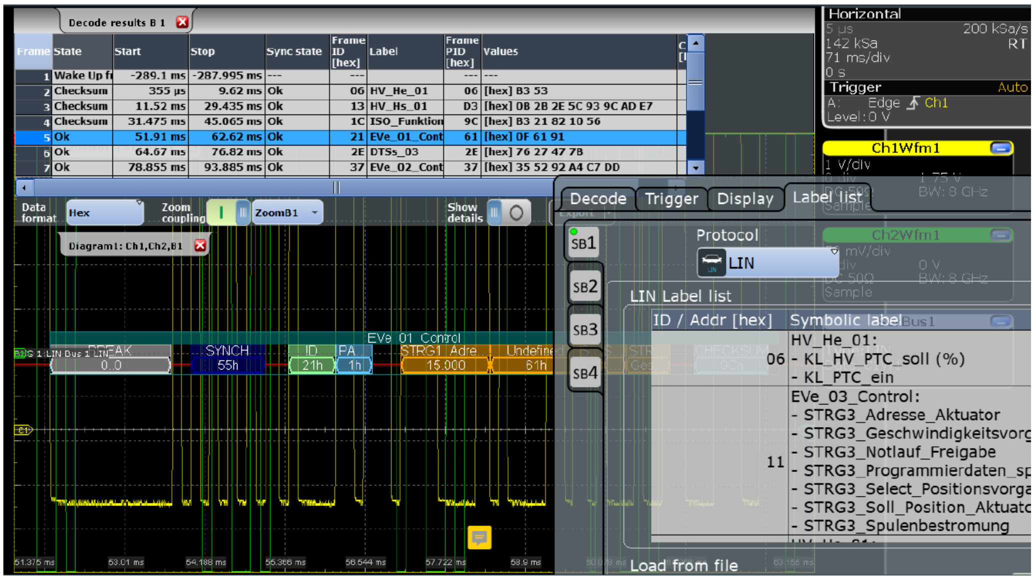
A LIN ldf file contains the following sections:

- Header
- Node section
- Signal section
- Frame section
- Schedule table
- Signal encoding section
- Encoding to signal mapping

Example of a LIN LDF file

```
LIN_description_file;
LIN_protocol_version = "2.0";
LIN_language_version = "2.0";
LIN_speed = 19.2 kbps;

Nodes {
    Master: Master1, 10 ms, 0.1 ms;
    Slaves: Slave_Motor, Slave_Sensor;
}

Signals {
  Sensor1:1,0x0,Slave_Sensor, Master1;
  Temperature1:12,0,Slave_Sensor, Master1;
}

Frames {
  Sensor_Frame: {Error,30;
                 Temperature, 20;
                 Status, 0;}
  Master_Frame: {Error,15;
                 Speed, 10;}
}

Schedule_tables {
  Sensor_Frame delay 10.000 ms;
        Master_Frame delay 15.000 ms;
}

Signal_encoding_types {
  Speed {logical_value, 0, "Motor_off";
         logical_value, 1, "Speed1";
         logical_value, 2, "Speed2";
         logical_value, 3, "Speed3";
    }
}
```

```
Signal_representation {
  Speed1_encoding:Speed1
  Temperature1_encoding:Temperature1
}
```



Remote commands:

- BUS<m>:LIN:FRAMe<n>:SDATa? on page 1501
- BUS<m>:LIN:FRAMe<n>:SDEXport? on page 1502

12.6.5 LIN Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The endianness setting ("Transfer") is a trigger setting and not considered for decoding. The binary results of data bytes are displayed MSB first.

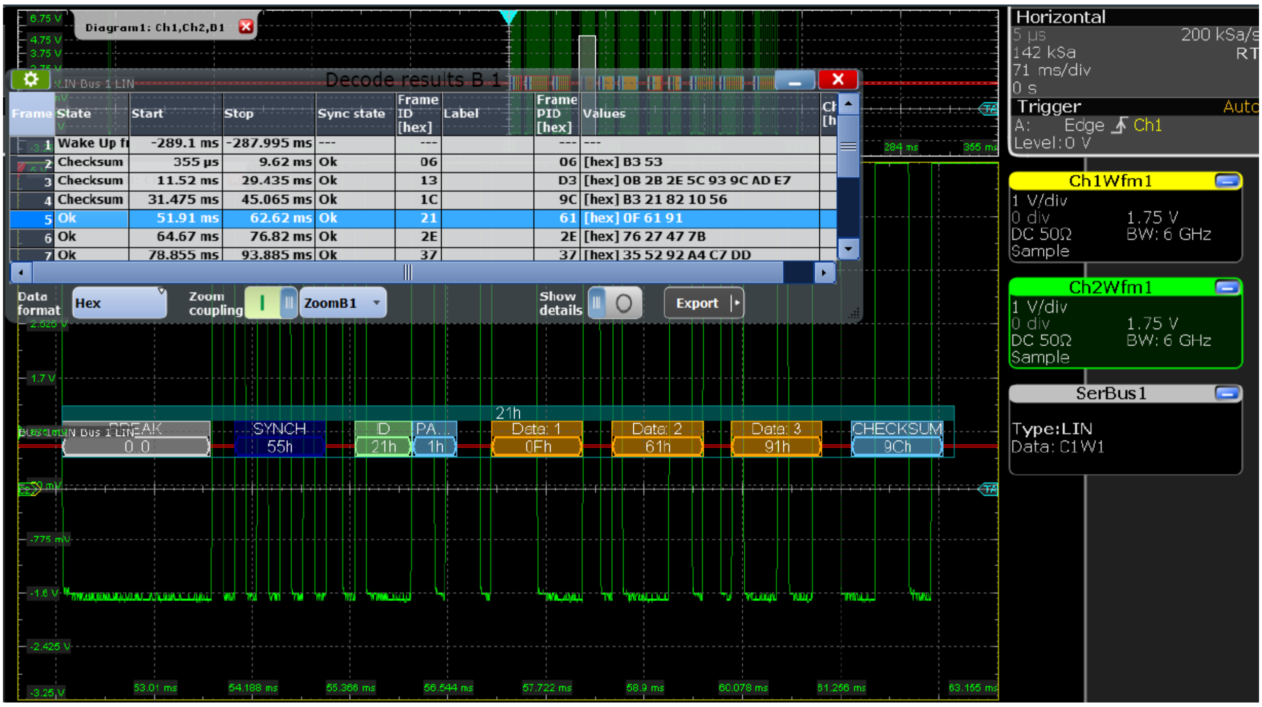


Figure 12-34: Decoded and binary LIN signal, and decode results

green brackets [...]	= start and end of frame
green frame header	= frame state is ok
red frame header	= error in frame
magenta frame header	= wakeup frame
magenta	= break
blue	= sync
yellow	= frame ID ok
gray	= data bytes
purple	= parity bit, or checksum ok
red	= error in frame ID, or checksum, or parity bit

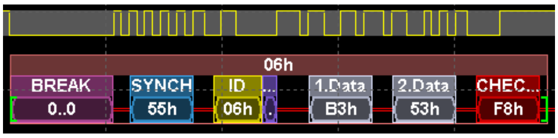


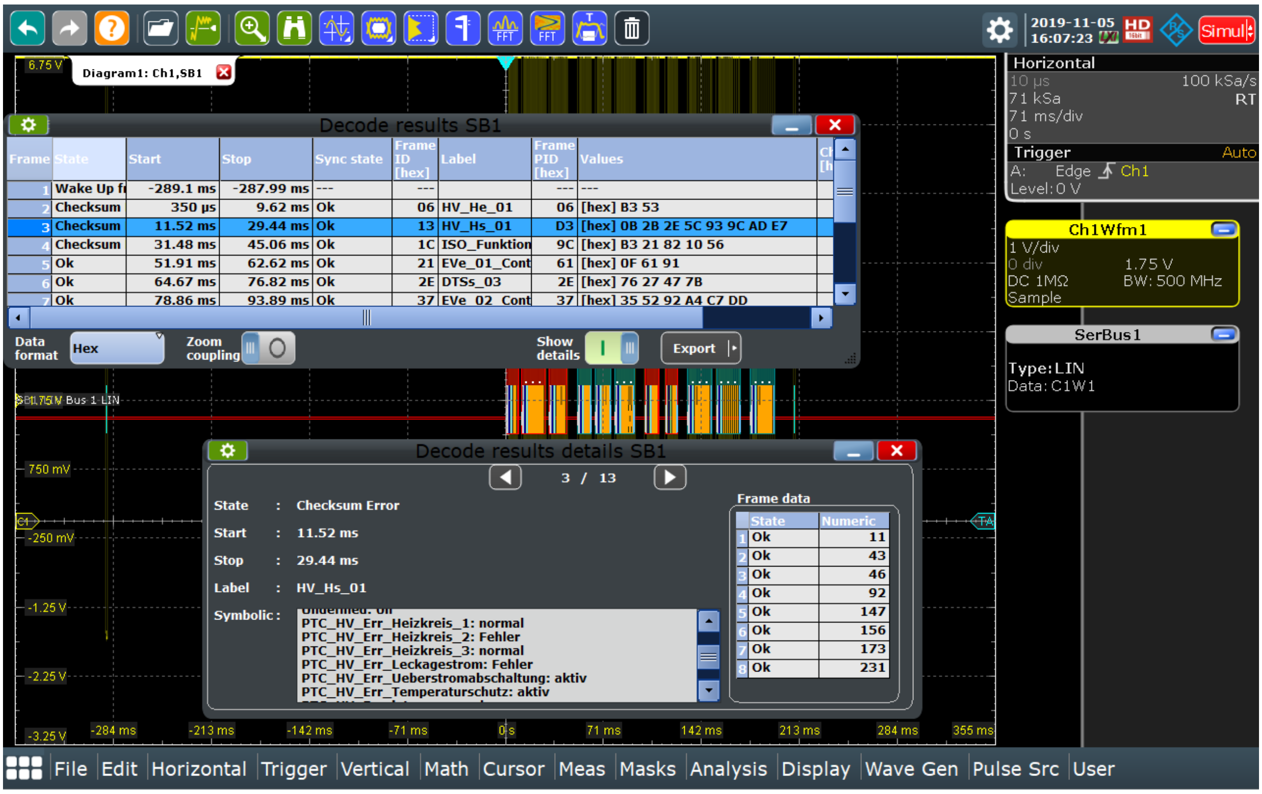
Figure 12-35: Decoded frame with checksum error (frame No 1 in figure above)

Table 12-7: Content of the Decode results table

Column	Description
State	Overall state of the frame.
Start	Time of frame start
Stop	Time of frame stop
Label	Symbolic label name defined in the label list
Sync state	Result of synchronization
Frame ID (hex)	Identifier value

Column	Description
Label	Symbolic label name defined in the label list
Frame PID (hex)	Protected identifier
Values	Value of the data bytes. The data format is selected below the table.
Chks (hex)	Checksum value
Bit rate	Value of the bit rate

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. Details include the frame data, the label and the symbolic name.



Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:
- BUS<m>:ZCOupling on page 1385

For an introduction to zoom settings and setup, see:
- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.7.3, "Decode Results", on page 1500.

12.6.6 Search on Decoded LIN Data

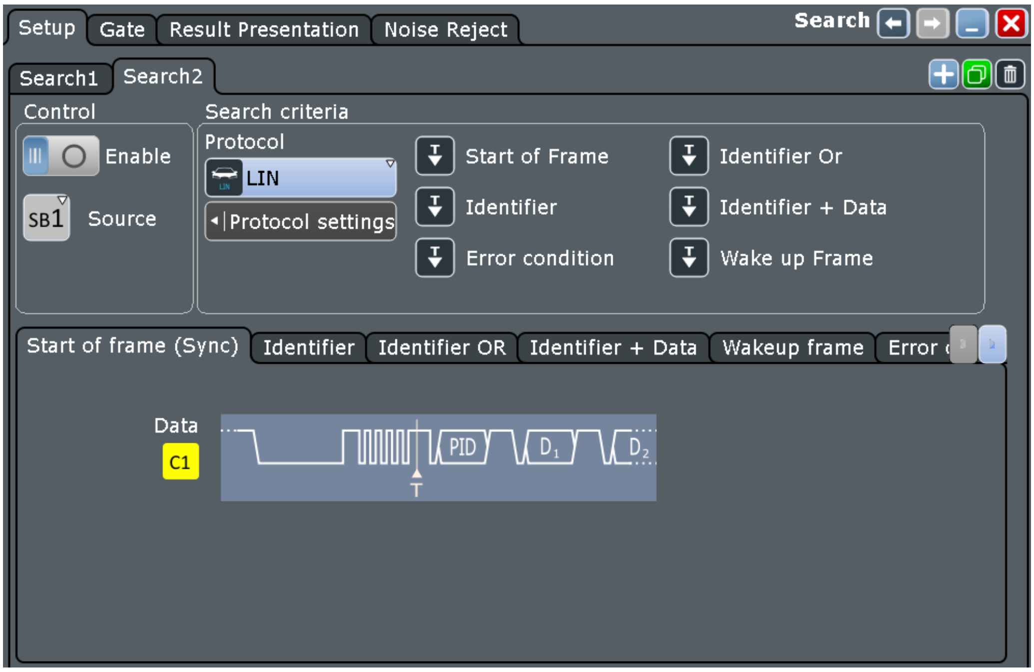
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.6.6.1 LIN Search Setup

Access: [SEARCH] > "Setup" tab



Search criteria

Sets the type to be searched for.

"Start of frame (Sync)"
: Searches for the stop bit of the sync field.

"Identifier"
: Searches for one specific identifier or an identifier range.
See "Identifier setup: Condition, Frame ID min, Frame ID max" on page 577

"Identifier OR"
: Searches for a combination of up to four identifiers.
See "Identifier OR setup: Monitor, Frame ID" on page 577

"Identifier + Data"
: Searches for a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern.
The identifier conditions are the same as for the "Identifier" trigger type, see "Identifier setup: Condition, Frame ID min, Frame ID max" on page 566.
Data conditions are set with "Data setup: Data length, Transfer, Condition, Data min, Data max" on page 567.

"Wakeup frame"
: Searches for wakeup frames.

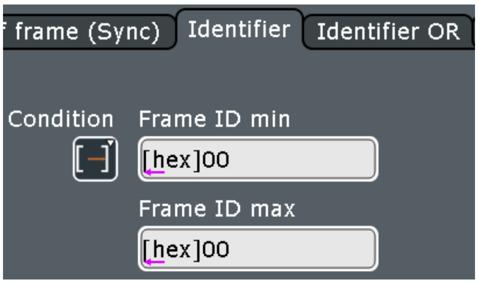
"Error condition"
: Identifies various errors in the frame, see "Error conditions" on page 568.

Remote command:
`SEARch:TRIGger:LIN:SSOFrame` on page 1507
`SEARch:TRIGger:LIN:SFIDentifier` on page 1508
`SEARch:TRIGger:LIN:IDENtifieror` on page 1508
`SEARch:TRIGger:LIN:SIDData` on page 1508

SEARch:TRIGger:LIN:WUFRame on page 1509
SEARch:TRIGger:LIN:SERRor on page 1508

Identifier setup: Condition, Frame ID min, Frame ID max
The identifier setup consists of the condition and one or two frame ID patterns.

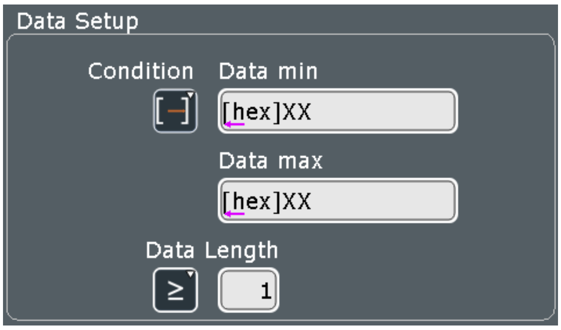
The identifier setup settings are the same as in the LIN trigger setup, see "Identifier setup: Condition, Frame ID min, Frame ID max" on page 566.



Remote command:
SEARch:TRIGger:LIN:ICONdition on page 1509
SEARch:TRIGger:LIN:IMIN on page 1509
SEARch:TRIGger:LIN:IMAX on page 1509

Data setup: Condition, Data min, Data max, Data length, Transfer
The data setup consists of the transfer direction, the data length, the condition, and one or two data patterns.

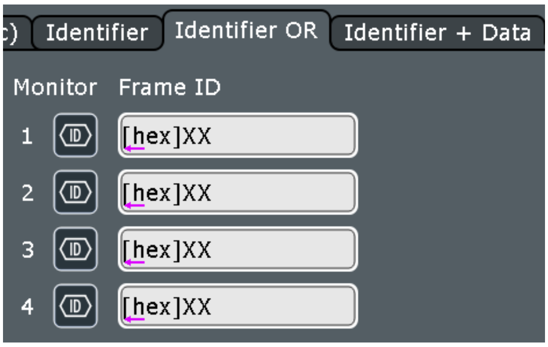
The data setup settings are the same as in the LIN trigger setup, see "Data setup: Data length, Transfer, Condition, Data min, Data max" on page 567.



Remote command:
SEARch:TRIGger:LIN:DCONdition on page 1510
SEARch:TRIGger:LIN:DMIN on page 1511
SEARch:TRIGger:LIN:DMAX on page 1511
SEARch:TRIGger:LIN:DLECondition on page 1512
SEARch:TRIGger:LIN:DLENgth on page 1512
SEARch:TRIGger:LIN:BORDer on page 1511

Identifier OR setup: Monitor, Frame ID
The identifier OR setup consists of the monitor and frame ID.

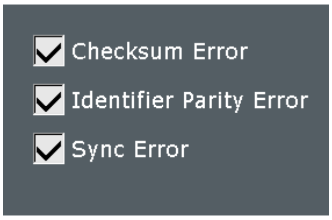
The identifier OR setup settings are the same as in the LIN trigger setup, see "Identifier OR setup: Monitor, Frame ID" on page 567



Remote command:
`SEARch:TRIGger:LIN:IDOR<m>:ENABle` on page 1510
`SEARch:TRIGger:LIN:IDOR<m>[:VALue]` on page 1510

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the LIN trigger setup, see "Error conditions" on page 568.



Remote command:
`SEARch:TRIGger:LIN:IPERror` on page 1512
`SEARch:TRIGger:LIN:SYERror` on page 1513
`SEARch:TRIGger:LIN:CHKSerror` on page 1513
`SEARch:TRIGger:LIN:ERRPattern` on page 1513
`SEARch:TRIGger:LIN:CRCDatalen` on page 1514
`SEARch:TRIGger:LIN:STANdard` on page 1514

12.6.6.2 LIN Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407

- Chapter 10.4, "Result Presentation", on page 424

The columns in the search result table are the same as in the decoding table, see Chapter 12.6.5, "LIN Decode Results", on page 572.

Remote commands:
- SEARch:RESult:LIN:FCOunt? on page 1515
- SEARch:RESult:LIN:FRAMe<m>:STATus? on page 1515
- SEARch:RESult:LIN:FRAMe<m>:STARt? on page 1515
- SEARch:RESult:LIN:FRAMe<m>:STOP? on page 1515
- SEARch:RESult:LIN:FRAMe<m>:DATA? on page 1516
- SEARch:RESult:LIN:FRAMe<m>:CSSTate? on page 1516
- SEARch:RESult:LIN:FRAMe<m>:CSValue? on page 1516
- SEARch:RESult:LIN:FRAMe<m>:IDSTate? on page 1517
- SEARch:RESult:LIN:FRAMe<m>:IDValue? on page 1517
- SEARch:RESult:LIN:FRAMe<m>:IDPValue? on page 1517
- SEARch:RESult:LIN:FRAMe<m>:SYMBol? on page 1518
- SEARch:RESult:LIN:FRAMe<m>:SYSTate? on page 1518
- SEARch:RESult:LIN:FRAMe<m>:VERSion? on page 1518
- SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe? on page 1519
- SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue? on page 1519

12.7 FlexRay (Option R&S RTE-K4)

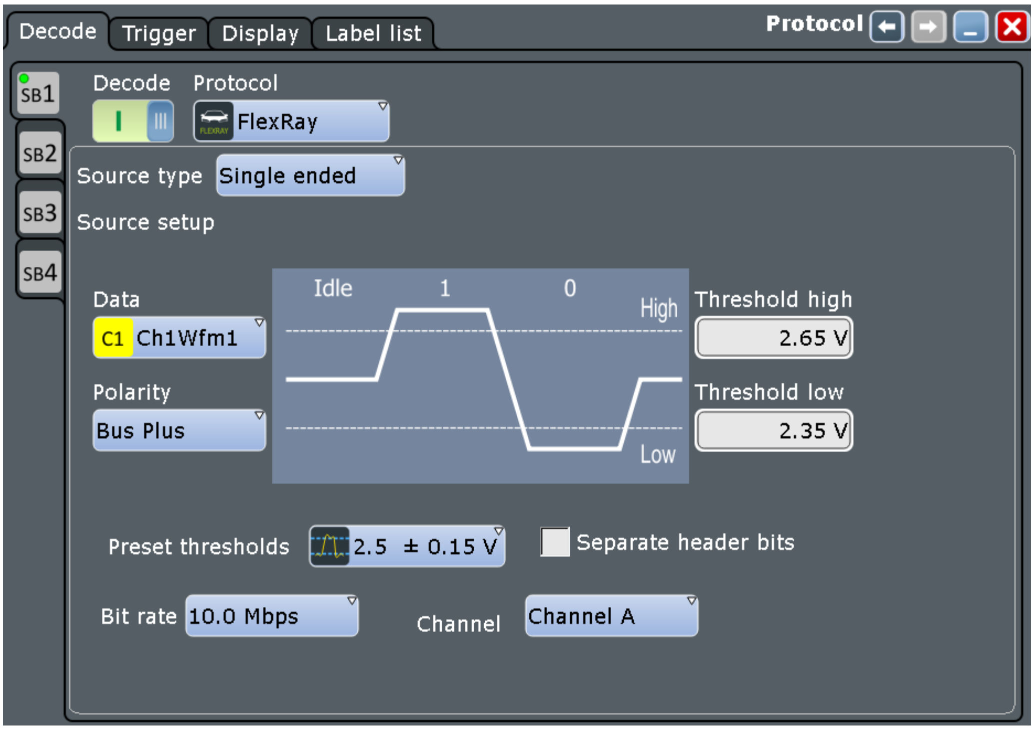
FlexRay is designed for use in safety-related distributed applications in the automotive industry. It is applied in real-time applications when higher data rates and reliable communication are required. In particular, FlexRay supports x-by-wire applications, for example, steer-by-wire or brake-by-wire.

12.7.1 FlexRay Configuration

12.7.1.1 FlexRay Configuration

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *FlexRay*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Source type

Sets the type of measurement. The instrument adjusts the thresholds to the selected source type.

"Single-ended" For measurements with single-ended probes, or single-ended voltage measurements with differential probes on the FlexRay bus. Two thresholds have to be defined as absolute voltage levels.

"Differential" For differential measurements on the FlexRay bus. This is the most common measurement. Two thresholds have to be defined as differential voltages.

"Logic" For measurements of logic signals, for example, of the logic signal inside the FlexRay node, between the communication controller and the bus driver. If MSO option R&S RTE-B1 is installed, you can use digital input channels. It is possible to measure simultaneously on a data line and on the enable line. Each line requires its own threshold.

Remote command:

BUS<m>:FLXRay:SRCType on page 1520

Data

Sets the input channel of the bus signal, or of the data line for a "Logic" source type. Usually, the source is one of the analog channels. Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

If the source type is "Logic", digital channels can be used (MSO option R&S RTE-B1 is required). Digital and analog channels cannot be used at the same time in a bus.

Remote command:
BUS<m>:FLXRay:SOURce<n> on page 1520

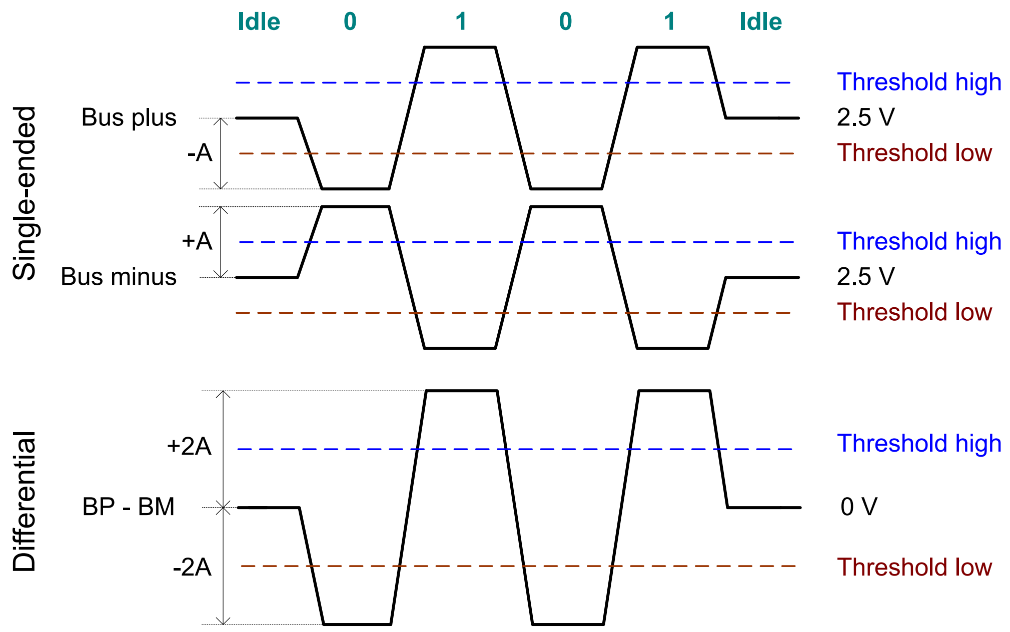
Enable
Sets the input channel of the enable line if there is a "Logic" source type. The enable line transfers the control signal of the bus guardian to the bus driver. None, or one of the analog channels can be used. Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

Alternatively to analog channels, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time in a bus.

Remote command:
BUS<m>:FLXRay:SOURce<n> on page 1520

Thresholds
Threshold values are used for digitization of the signal.

For measurements on a FlexRay bus, two thresholds are required to distinguish the three possible states of the signal - high, low and idle. If the signal value on the line is higher than the upper threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the lower threshold. If the value is between the threshold, the signal is in idle state.



For measurements inside the FlexRay node (with "Source type" = "Logic"), each line requires its threshold level.

There are two ways to set the thresholds: selection of a predefined value, or direct entry of a value.

- "Preset thresholds"

Selects default threshold voltages from a list. The predefined values depend on the selected source type. The value is set to "Manual" if at least one threshold was entered directly.

- "Threshold high" and "Threshold low"
  Upper and lower levels for single-ended or differential source types. You can enter the values directly in the fields.
- "Threshold data" and "Threshold enable"
  Levels for data and enable line if there is logic source type. You can enter the values directly in the fields.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:FLXRay:PRSingle` on page 1522
`BUS<m>:FLXRay:PRDiff` on page 1522
`BUS<m>:FLXRay:PRLogic` on page 1522
`BUS<m>:FLXRay:THReshold<n>` on page 1521
`BUS<m>:FLXRay:THData` on page 1521
`BUS<m>:FLXRay:THENable` on page 1521
`BUS<m>:SETReflevels` on page 1383

Polarity
Selects the wire on which the bus signal is measured if there is "Single-ended" measurement: "Bus plus" or "Bus minus". The setting affects the digitization of the signal.

Remote command:
`BUS<m>:FLXRay:POLarity` on page 1523

Bit rate
Selects the number of transmitted bits per second from a list.

Remote command:
`BUS<m>:FLXRay:BITRate` on page 1523

Channel
Selects the FlexRay channel on which the signal is measured, either channel A or channel B. The setting is considered in the calculation of the frame CRC.

Remote command:
`BUS<m>:FLXRay:CHTYpe` on page 1523

Separate header bits
The setting affects the decoding and its display. If enabled, the leading five indicator bits of the header are decoded as five single bits. Otherwise, the indicator bits are shown as one word with word length 5 bits.

Remote command:
`BUS<m>:FLXRay:SEHB` on page 1524

12.7.1.2 Configuring FlexRay Protocol

The configuration of the FlexRay is simple - assign the two lines to input channels, and set the thresholds.

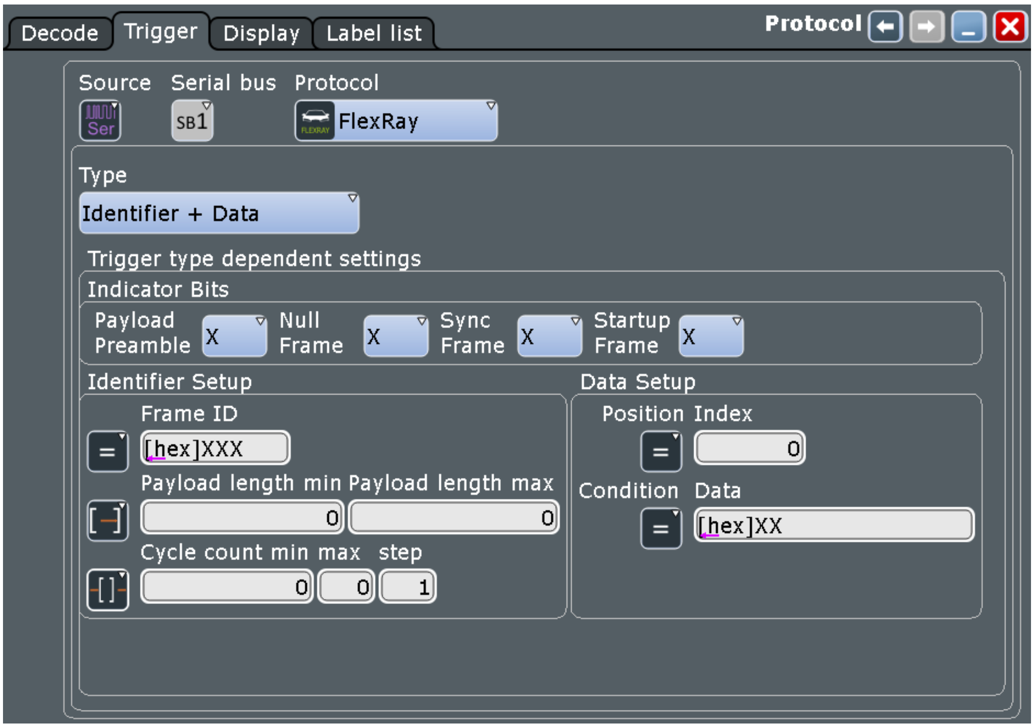
For details on configuration settings, see Chapter 12.7.1.1, "FlexRay Configuration", on page 579.

1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "FlexRay".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Select the "Source type".
7. Select the channels for the sources.
8. Adjust other settings, if necessary.
9. Set the logical thresholds.
10. Enable "Decode", if available.

12.7.2 FlexRay Trigger

12.7.2.1 FlexRay Trigger

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = FlexRay"



Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Trigger type

Selects the trigger type for FlexRay analysis.

"Start of frame" Triggers on the first rising edge after the transmission start sequence (TSS).

"Identifier +data" Triggers on the decoded frame content, on header and payload data:
- Indicator bits, see "Indicator bits" on page 585
- Frame identifier, see "Frame ID (min/max)" on page 585
- Payload length, see "Payload length (min/max)" on page 586
- Cycle count, see "Cycle count (min, max), Step" on page 586
- Data position, see "Position, Index (min, max) - Data setup" on page 587
- Data bit pattern, see "Condition, Data (min, max) - Data setup" on page 587

"Symbol" Triggers on a symbol or wakeup pattern, see "Symbol" on page 587.

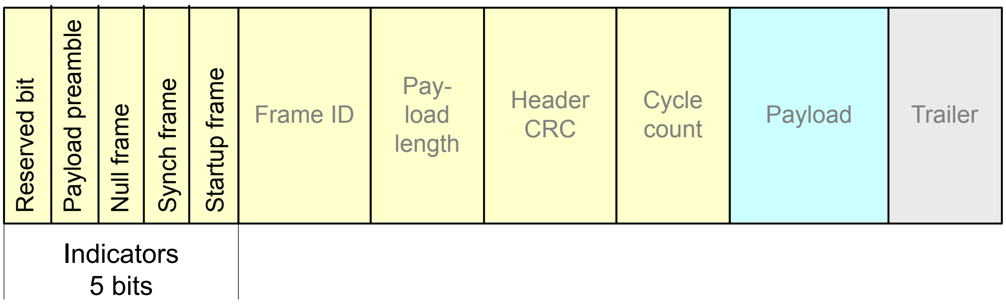
"Error condition" — Triggers on one or more errors that are detected in the decoded data, see "Error conditions" on page 588.

Remote command:
`TRIGger<m>:FLXRay:TYPE` on page 1525

Indicator bits

Triggers on one or more indicator bits at the beginning of the header segment. Each bit can be set to 0, 1, or X (do not care).

Trigger type: "Identifier + data"



"Payload preamble" — Indicates a Network Management Vector in the payload segment. The NMV allows the host processor to send data directly, without processing by the communication controller.

"Null frame" — Indicates a frame without usable data.

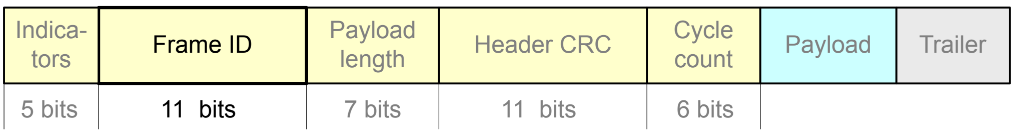
"Sync frame" — Indicates that the frame is used for synchronization of the FlexRay system. Only sync nodes can send this frame type.

"Startup frame" — Indicates a startup frame used for startup of the network. Only specific start nodes can send this frame type.

Remote command:
`TRIGger<m>:FLXRay:PLPReamble` on page 1526
`TRIGger<m>:FLXRay:NUFRame` on page 1526
`TRIGger<m>:FLXRay:SYFRame` on page 1526
`TRIGger<m>:FLXRay:STFRame` on page 1526

Frame ID (min/max)

The frame ID contains the number of the slot in which the frame is transmitted. Each frame ID occurs only once during a FlexRay cycle.



To trigger on a frame ID, you have to define a condition and one or two identifier patterns. The second identifier pattern is required to specify a range with conditions "In range" and "Out of range". In binary format, use the following characters: 1; 0; or X (any bit). The use of X is restricted to the conditions "Equal" and "Not equal". If the identifier is not relevant for the trigger setup, set it to "Off".

The maximum length of the pattern is 11 bit. The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Trigger type: "Identifier + data"

Remote command:
`TRIGger<m>:FLXRay:FCONdition` on page 1527
`TRIGger<m>:FLXRay:FMIN` on page 1527
`TRIGger<m>:FLXRay:FMAX` on page 1527

Payload length (min/max)
The payload length contains the number of words transmitted in the payload segment. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Indica-tors	Frame ID	Payload length	Header CRC	Cycle count	Payload	Trailer
5 bits	11 bits	7 bits	11 bits	6 bits		

To trigger on the payload length, you have to define a condition and one or two numbers of words. The second number is required to specify a range with conditions "In range" and "Out of range". If the payload length is not relevant for the trigger setup, set it to "Off".

Trigger type: "Identifier + data"

Remote command:
`TRIGger<m>:FLXRay:PCONdition` on page 1528
`TRIGger<m>:FLXRay:PMIN` on page 1528
`TRIGger<m>:FLXRay:PMAX` on page 1528

Cycle count (min, max), Step
The cycle count contains the number of the current FlexRay cycle.

Indica-tors	Frame ID	Payload length	Header CRC	Cycle count	Payload	Trailer
5 bits	11 bits	7 bits	11 bits	6 bits		

To trigger on the cycle count, you have to define a condition and one or two numbers. If the condition is a range ("In range" or "Out of range"), a second number "Cycle count max" is required.

Also, you can define a "Step" to trigger on each n-th cycle inside the given range. This allows for specific triggering if slot multiplexing is used.

If the cycle count is not relevant for the trigger setup, set it to "Off".

Trigger type: "Identifier + data"

Remote command:
`TRIGger<m>:FLXRay:CENable` on page 1528
`TRIGger<m>:FLXRay:CMIN` on page 1529
`TRIGger<m>:FLXRay:CMAX` on page 1529
`TRIGger<m>:FLXRay:CSTep` on page 1529

Position, Index (min, max) - Data setup
Sets the position of the first byte of data bit pattern within the payload segment. You can define an exact position, or a position range.

Trigger type: "Identifier + data"

"Position" Operator for the data position. Select "Off", if the position of the required pattern is not relevant for the trigger condition.

"Index" Sets the number of data bytes to be skipped after start of the payload segment if "Position" is "Equal" or "Greater or equal". The index 0 is associated with the first data byte.

"Index min, Index max"
If the "Position" operator defines a range, the indexes of the first and the last byte are defined between which the required bit pattern may start.

Remote command:
`TRIGger<m>:FLXRay:DPOPerator` on page 1530
`TRIGger<m>:FLXRay:DPOSition` on page 1530
`TRIGger<m>:FLXRay:DPTO` on page 1530

Condition, Data (min, max) - Data setup
Specifies the data bit pattern to be found in the payload segment. The starting point of the pattern is defined by "Position, Index (min, max) - Data setup" on page 587. The pattern comparison is byte-aligned, and the instrument triggers at the end of a byte.

"Condition" Sets the operator to set a specific data pattern ("Equal" or "Not equal") or a data range. Select "Off", if the data pattern is not relevant for the trigger condition.

"Data (min/max)" Enter the bytes in msb first bit order. The maximum pattern length is 8 bytes.
In binary format, you can use the following characters: 1; 0; or X (any bit). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Remote command:
`TRIGger<m>:FLXRay:DCONdition` on page 1530
`TRIGger<m>:FLXRay:DMIN` on page 1531
`TRIGger<m>:FLXRay:DMAX` on page 1531

Symbol
Triggers on a symbol or on a wakeup pattern.

Trigger type: "Symbol"

"CAS/MTS" Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.

"Wakeup Pattern" The wakeup pattern is sent to activate the nodes of the system.

Remote command:
`TRIGger<m>:FLXRay:SYMBol` on page 1531

Error conditions

Triggers on one or more errors in the frame.

Trigger type: "Error conditions"

"FSS"	Error in a Frame Start Sequence. FSS follows the Transmission Start Sequence TSS at the beginning of each frame.
"BSS"	Error in a Byte Start Sequence. The BSS is transmitted before each byte.
"FES"	Error in Frame End Sequence. FES indicates the end of each frame.
"Header CRC"	Error in a cyclic redundancy check code of the header data which covers mainly frame ID and payload length.
"Payload CRC"	Error in a cyclic redundancy check code of the complete frame.

Remote command:
`TRIGger<m>:FLXRay:FSSerror` on page 1532
`TRIGger<m>:FLXRay:BSSerror` on page 1531
`TRIGger<m>:FLXRay:FESerror` on page 1532
`TRIGger<m>:FLXRay:HCRCerror` on page 1532
`TRIGger<m>:FLXRay:PCRCerror` on page 1532

12.7.2.2 Triggering on FlexRay Signals

Prerequisites: An FlexRay bus is configured, see .

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to FlexRay.
5. Select the "Trigger type".
6. For more complex trigger types, enter the address and/or data conditions: address, acknowledge bits, R/W bit, and data pattern.
   For details, see Chapter 12.7.2.1, "FlexRay Trigger", on page 583.

12.7.3 FlexRay Label List

Label lists are protocol-specific. A FlexRay label file contains four values for each identifier:

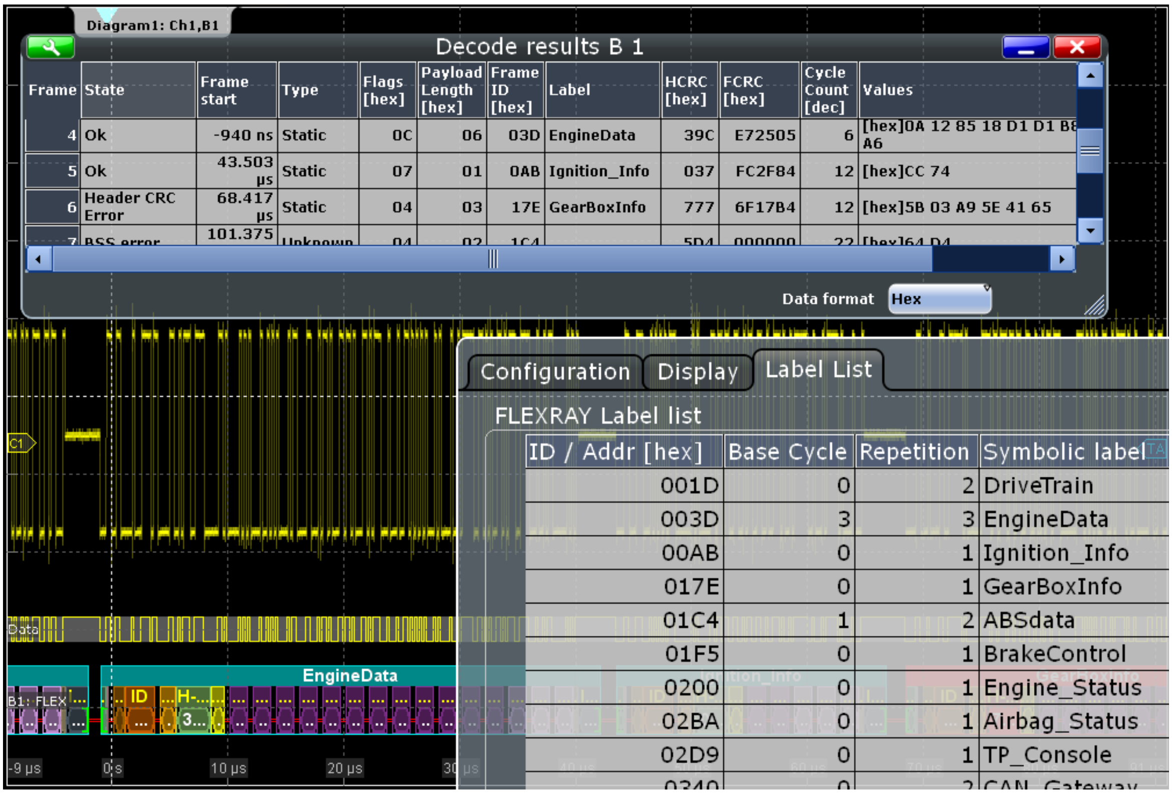
- "ID / Addr": number of the slot in which the frame is transmitted
- "Base cycle" and "Repetition": define the cycle indexes for which the identifier applies. Base cycle defines the first applied cycle.
  There are 64 cycles in a FlexRay communication. The same identifier can be shared by different devices, and each device uses the identifier at different cycles.
  For example:
  0x0AB,0,2,Ignition_Info: uses cycles 0,2,4,6,...,62

0x0AB,1,2,GearBoxInfo: uses cycles 1,3,5,7,...,63

- "Symbolic label": symbolic name of the identifier, specifying the device function.

Example: FlexRay PTT file

```
# ---------------------------------------------------------------------------
@FILE_VERSION = 1.0
@PROTOCOL_NAME = flexray
# ---------------------------------------------------------------------------
# Labels for FlexRay protocol
#   Column order: Identifier, Base cycle, Cycle repetition, Label
# ---------------------------------------------------------------------------
# ----Definition----
0x01D,0,2,DriveTrain
0x03D,3,3,EngineData
0x0AB,0,2,Ignition_Info
0x0AB,1,2,GearBoxInfo
0x1C4,1,2,ABSdata
0x1F5,0,1,BrakeControl
0x200,0,1,Engine_Status
0x2BA,0,1,Airbag_Status
0x2D9,0,1,TP_Console
0x340,0,2,CAN_Gateway
0x38B,55,1,MOST_Gateway
0x3EA,0,1,PressureInfo
# ---------------------------------------------------------------------------
```



For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- BUS<m>:FLXRay:FRAMe<n>:SYMBol? on page 1534

12.7.4 FlexRay Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

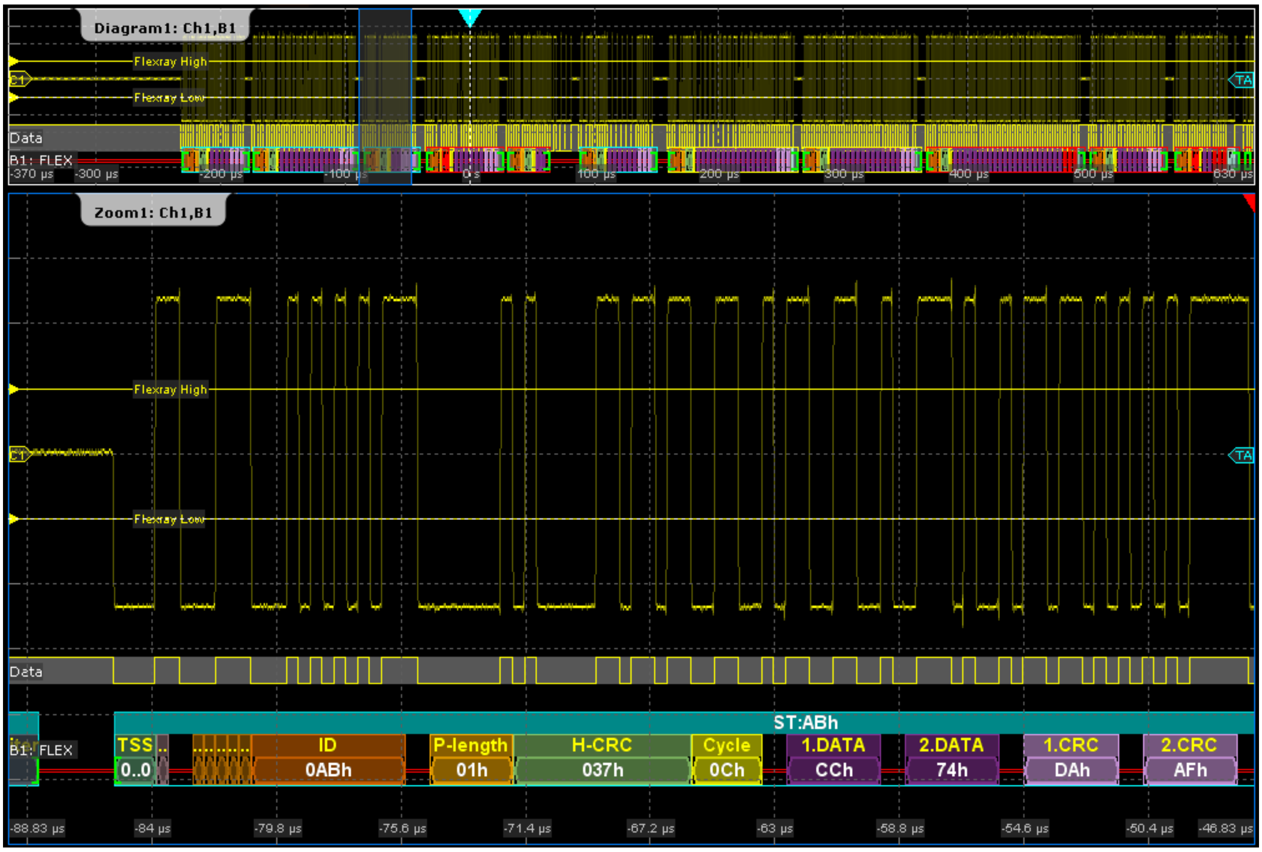


Figure 12-36: FlexRay - decoded static slot

Data is decoded and displayed in the order of its reception. The "Decode results" box shows the detailed decoded data for each frame as it is received.

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

Figure 12-37: FlexRay - decoded dynamic slot and results table

Table 12-8: Content of the "Decode results" table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Frame start	Time of frame start
Type	Frame type: Frame of the static segment, frame of the dynamic segment, wakeup frame, symbol in the frame
Flags	State of indicator bits
Payload length	Number of data words in the payload segment.
Frame ID	Value of the frame ID (slot number)
Label	Symbolic label name defined in the label list
HCRC	Value of the header CRC
FCRC	Value of the frame CRC

Column	Description
Cycle count	Number of the current FlexRay cycle
Values	Value of the data bytes. The data format is selected below the table. Wakeup and symbol frames do not transmit data, therefore "- - -" is displayed.

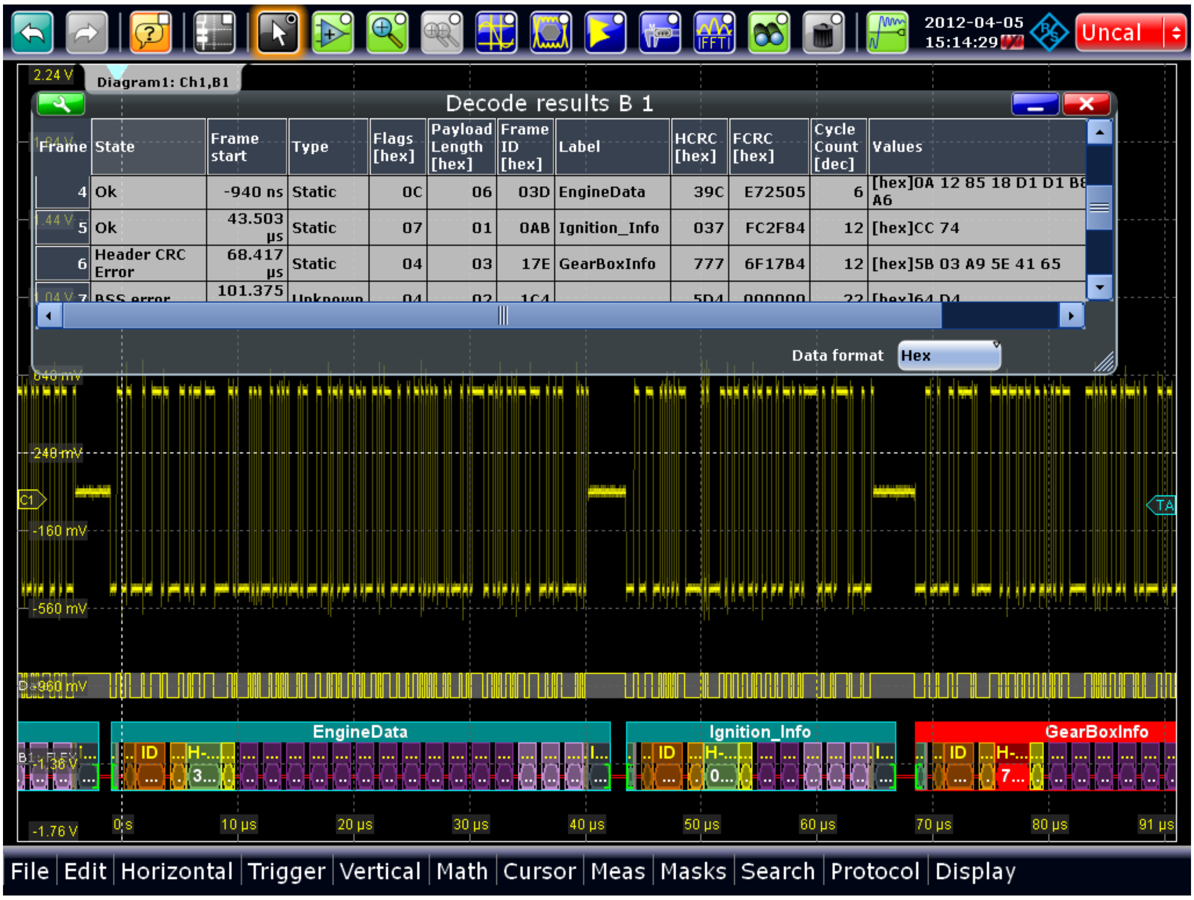


Figure 12-38: FlexRay - decode results with applied label list

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- BUS<m>:FORMat on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- BUS<m>:ZCOupling on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.8.3, "Decode Results", on page 1532.

12.7.5 Search on Decoded FlexRay Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

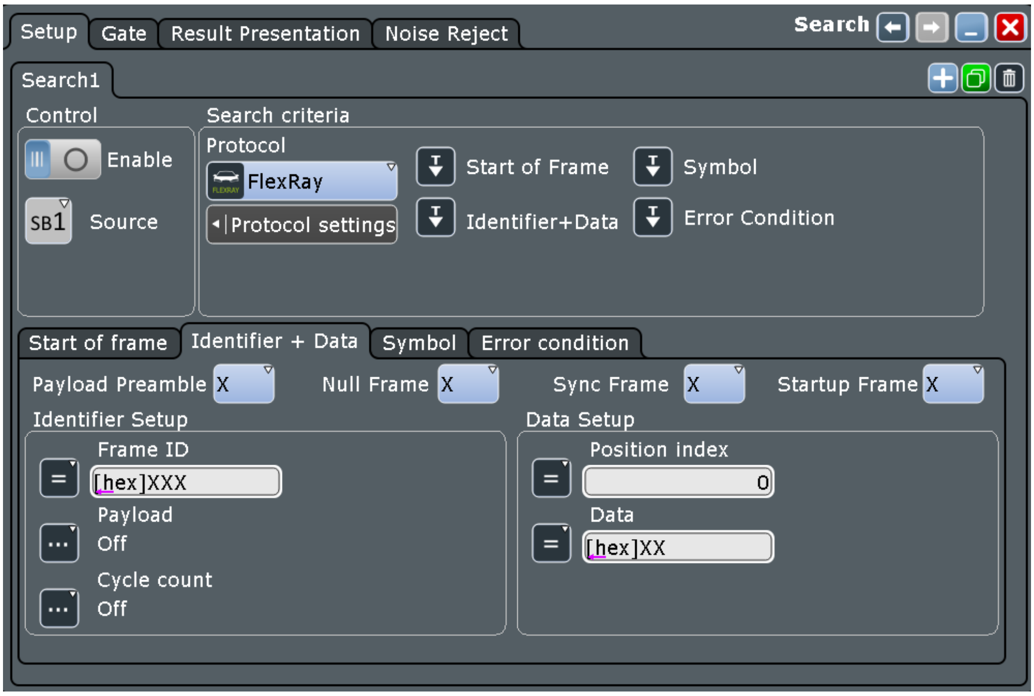
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

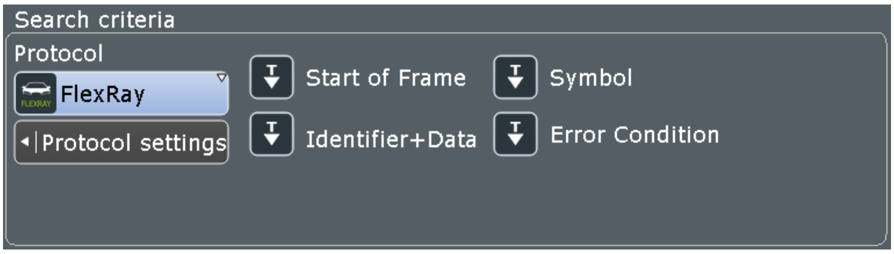
12.7.5.1 FlexRay Search Setup

Access: [SEARCH] > "Setup" tab



Search Criteria

Sets one criterion or an AND-combination of criteria to be searched for. If more than one criterion is selected, all criteria must be fulfilled by a frame for it to be shown in the search results.



"Start of frame" Searches for the first rising edge after the transmission start sequence (TSS).

"Identifier+data"

Searches for the decoded frame content, on header and payload data:

- Indicator bits, see "Indicator bits" on page 595
- Frame identifier, see "Frame ID (min/max)" on page 595
- Payload length, see "Payload length (min/max)" on page 596
- Cycle count, see "Cycle count (min, max), Step" on page 596
- Data position, see "Position, Index (min, max) - Data setup" on page 596
- Data bit pattern, see "Condition, Data (min, max) - Data setup" on page 596

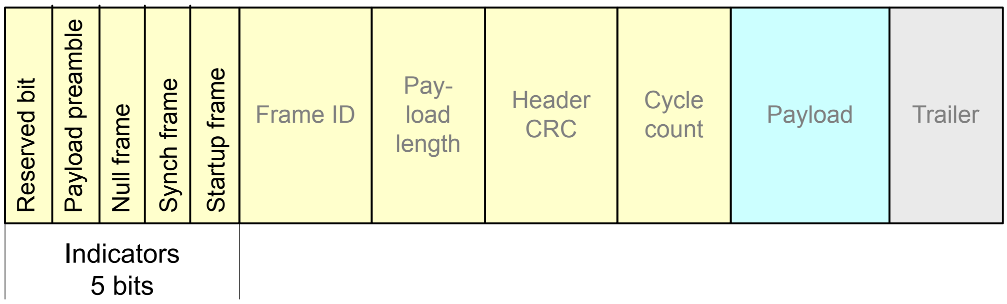
"Symbol" Searches for a symbol or wakeup pattern, see "Symbol" on page 596.

"Error condition"
Searches for one or more errors that are detected in the decoded data, see "Error Condition" on page 597.

Remote command:
`SEARch:TRIGger:FLXRay[:SSOFrame]` on page 1539
`SEARch:TRIGger:FLXRay:SSYMbol` on page 1540
`SEARch:TRIGger:FLXRay:SIDData` on page 1539
`SEARch:TRIGger:FLXRay:SERRor` on page 1539

Indicator bits

Searches for one or more indicator bits at the beginning of the header segment. Each bit can be set to 0, 1, or X (do not care).



"Payload pre-amble"	Indicates a Network Management Vector in the payload segment. The NMV allows the host processor to send data directly, without processing by the communication controller.
"Null frame"	Indicates a frame without usable data.
"Sync frame"	Indicates that the frame is used for synchronization of the FlexRay system. Only sync nodes can send this frame type.
"Startup frame"	Indicates a startup frame used for startup of the network. Only specific start nodes can send this frame type.

Remote command:
`SEARch:TRIGger:FLXRay:PLPReamble` on page 1545
`SEARch:TRIGger:FLXRay:NUFRame` on page 1544
`SEARch:TRIGger:FLXRay:SYFRame` on page 1546
`SEARch:TRIGger:FLXRay:STFRame` on page 1546

Frame ID (min/max)

The frame ID contains the number of the slot in which the frame is transmitted. Each frame ID occurs only once during a FlexRay cycle.

The setup conditions are the same as in the FlexRay trigger setup, see "Frame ID (min/max)" on page 585.

Remote command:
`SEARch:TRIGger:FLXRay:FCONdition` on page 1543
`SEARch:TRIGger:FLXRay:FMIN` on page 1544
`SEARch:TRIGger:FLXRay:FMAX` on page 1544

Payload length (min/max)
The payload length contains the number of words transmitted in the payload segment. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

The setup conditions are the same as in the FlexRay trigger setup, see "Payload length (min/max)" on page 586.

Remote command:
`SEARch:TRIGger:FLXRay:PCONdition` on page 1545
`SEARch:TRIGger:FLXRay:PMIN` on page 1546
`SEARch:TRIGger:FLXRay:PMAX` on page 1545

Cycle count (min, max), Step
The cycle count contains the number of the current FlexRay cycle.

The setup conditions are the same as in the FlexRay trigger setup, see "Cycle count (min, max), Step" on page 586.

Remote command:
`SEARch:TRIGger:FLXRay:CENable` on page 1540
`SEARch:TRIGger:FLXRay:CMIN` on page 1541
`SEARch:TRIGger:FLXRay:CMAX` on page 1540
`SEARch:TRIGger:FLXRay:CSTep` on page 1541

Position, Index (min, max) - Data setup
Sets the position of the first byte of data bit pattern within the payload segment. You can define an exact position, or a position range.

The setup conditions are the same as in the FlexRay trigger setup, see "Position, Index (min, max) - Data setup" on page 587.

Remote command:
`SEARch:TRIGger:FLXRay:DPOPerator` on page 1542
`SEARch:TRIGger:FLXRay:DPOSition` on page 1543
`SEARch:TRIGger:FLXRay:DPTO` on page 1543

Condition, Data (min, max) - Data setup
Specifies the data bit pattern to be found in the payload segment. The starting point of the pattern is defined by "Position, Index (min, max) - Data setup" on page 596. The pattern comparison is byte-aligned, and the instrument triggers at the end of a byte.

The setup conditions are the same as in the FlexRay trigger setup, see "Condition, Data (min, max) - Data setup" on page 587.

Remote command:
`SEARch:TRIGger:FLXRay:DCONdition` on page 1541
`SEARch:TRIGger:FLXRay:DMIN` on page 1542
`SEARch:TRIGger:FLXRay:DMAX` on page 1542

Symbol
Searches for a symbol or a wakeup pattern.

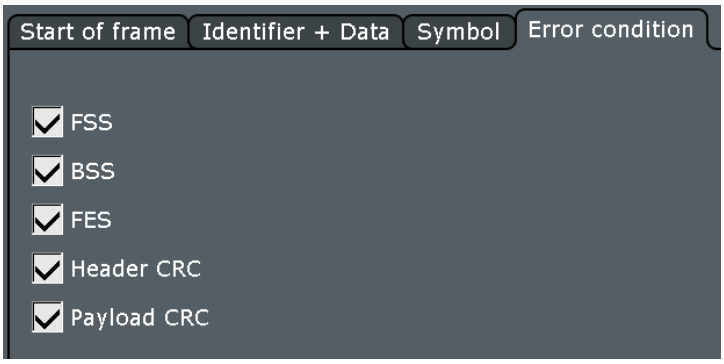
"CAS/MTS"	Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.
"Wakeup Pattern"	The wakeup pattern is sent to activate the nodes of the system.

Remote command:
`SEARch:TRIGger:FLXRay:SYMBol` on page 1547

Error Condition

Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the FlexRay trigger setup, see "Error conditions" on page 588.



Remote command:
`SEARch:TRIGger:FLXRay:BSSerror` on page 1547
`SEARch:TRIGger:FLXRay:FESerror` on page 1547
`SEARch:TRIGger:FLXRay:FSSerror` on page 1547
`SEARch:TRIGger:FLXRay:HCRCerror` on page 1548
`SEARch:TRIGger:FLXRay:PCRCerror` on page 1548

12.7.5.2 FlexRay Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- `SEARch:RESult:FLXRay:FCOunt?` on page 1549
- `SEARch:RESult:FLXRay:FRAMe<m>:ADID?` on page 1549

- SEARch:RESult:FLXRay:FRAMe<m>:CSSTate? on page 1549
- SEARch:RESult:FLXRay:FRAMe<m>:CSValue? on page 1550
- SEARch:RESult:FLXRay:FRAMe<m>:CYCount? on page 1550
- SEARch:RESult:FLXRay:FRAMe<m>:DATA? on page 1550
- SEARch:RESult:FLXRay:FRAMe<m>:FCSTate? on page 1550
- SEARch:RESult:FLXRay:FRAMe<m>:FCValue? on page 1551
- SEARch:RESult:FLXRay:FRAMe<m>:FLAGs? on page 1551
- SEARch:RESult:FLXRay:FRAMe<m>:PAYLength? on page 1551
- SEARch:RESult:FLXRay:FRAMe<m>:STATus? on page 1552
- SEARch:RESult:FLXRay:FRAMe<m>:STARt? on page 1552
- SEARch:RESult:FLXRay:FRAMe<m>:STOP? on page 1553
- SEARch:RESult:FLXRay:FRAMe<m>:SYMBol? on page 1553
- SEARch:RESult:FLXRay:FRAMe<m>:TYPE? on page 1553

12.8 Audio Signals (Option R&S RTE-K5)

The R&S RTE can analyze several standard and de-facto industry standard signals: I²S Inter-IC Sound standard audio format, left justified and right justified data formats and Time Division Multiplexed (TDM) audio format.

- Audio Protocols....................598
- Audio Configuration....................600
- Audio Trigger....................605
- Audio Decode Results....................610
- Track....................612
- Trend....................617

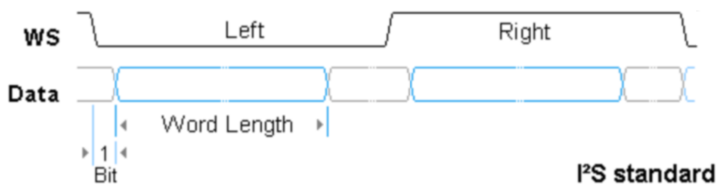
12.8.1 Audio Protocols

All audio protocols use 3 lines:

- The clock line generates the bit clock.
- The word select line (WS, also known as word clock) defines the frame start and the maximum length of the data word.
  For pulse code modulated signals (I²S standard, left and right justified data formats), the level of the WS signal assigns the data words to the left and right channels.
  TDM uses frame synchronization pulses on the WS line to identify the beginning of a frame.
- The data line transmits the audio data in time-multiplexed data channels.

12.8.1.1 I²S Standard

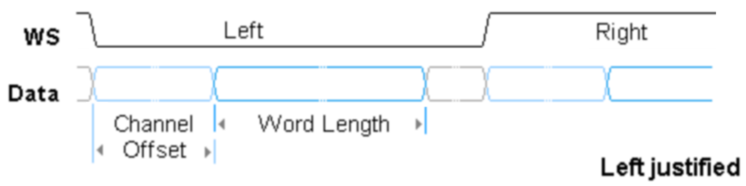
I²S standard interfaces transmit two PCM coded audio channels. The WS line selects the channel being transmitted - left or right channel. Usually, 32 bits are transmitted on each channel. The data word can be shorter than the channel length, and the receiver ignores the remaining bits. The first byte of the audio word is delayed one clock period from the leading edge of the word select pulse. The R&S RTE can decode I²S standard signals with MSBF and LSBF bit order.



12.8.1.2 Left Justified Data Format

The left justified data format is very similar to the I²S standard, but the first byte of the audio word is aligned with the leading edge of the word select pulse. Thus the audio word is left justified within the frame. The data word can be shorter than the channel length.

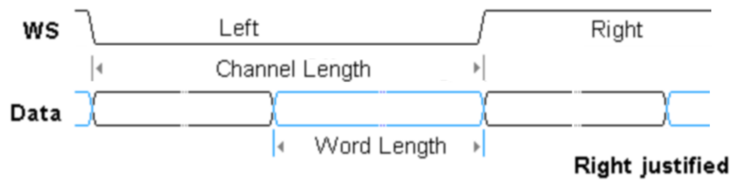
In addition to the standard configuration, the R&S RTE can analyze also left justified data formats which send the data word with offset to the WS edge. The bit order can be MSBF or LSBF.



12.8.1.3 Right Justified Data Format

The right-justified data format is similar to the left-justified, but the last byte of the word in the frame is aligned with the trailing edge of the word select pulse. Thus the audio word is right-aligned within the frame.

The R&S RTE can decode right justified signals with MSBF and LSBF bit order.



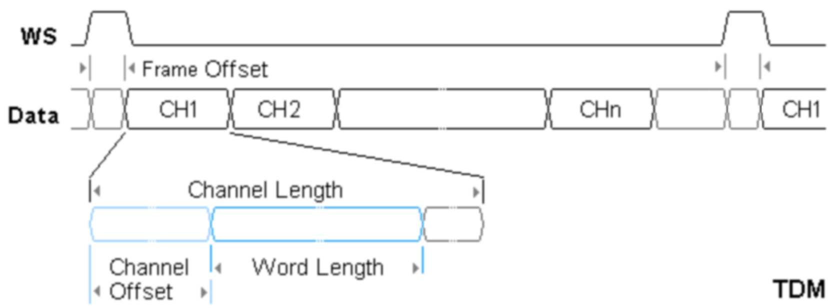
12.8.1.4 TDM

The Time Division Multiplexed (TDM) audio format is not standardized and provides high flexibility for transfer of more than two audio data channels on one line. On the word select line, it uses frame synchronization pulses to identify the beginning of a frame. On the data line, channel blocks of a defined length are transmitted. Each block contains an audio word that can be shorter than the channel length.

Each frame can start with frame offset bits, which preceed the first channel. Inside the channel, the audio word also can have an offset to the channel start.

Channel length, channel offset and word length are dependent values:

Channel length ≥ Word length + Channel offset



12.8.2 Audio Configuration

12.8.2.1 Audio Signal Configuration

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *Audio*

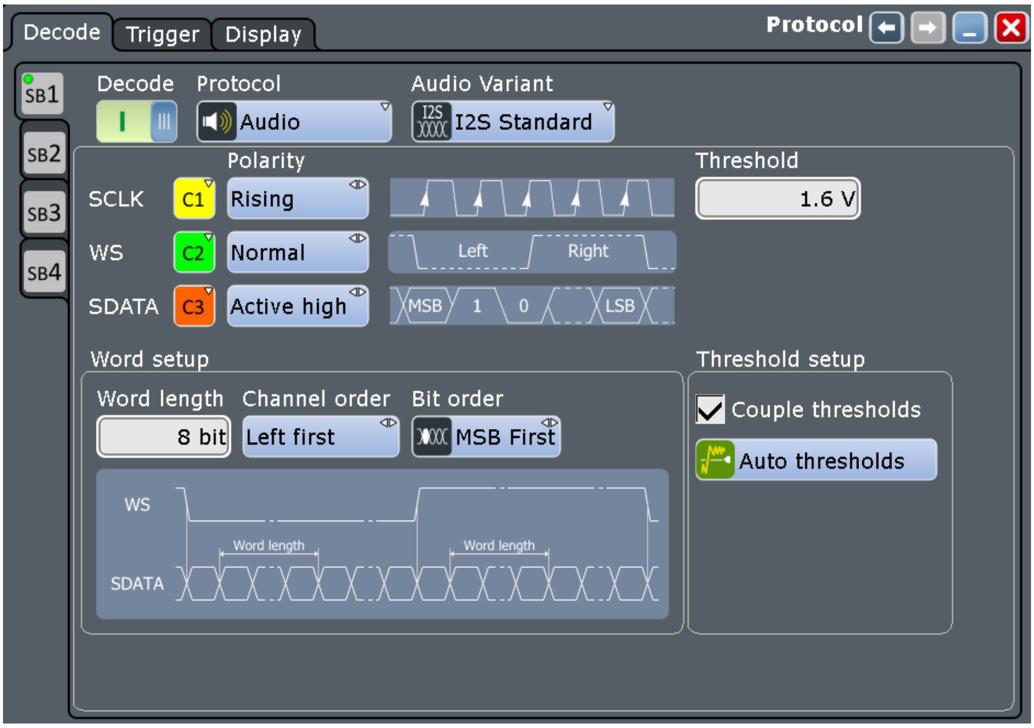
Make sure that the tab of the correct serial bus is selected on the left side.

In the "Decode" tab you configure the audio signal. Several audio signal variants are available: the I²S standard signal, the left- and right-justified data formats, and the TDM interface.

For all audio signal variants, you define the line sources and their polarities. Also, if coupling is active, one threshold for all sources; if coupling is not enabled, three thresholds for each source.

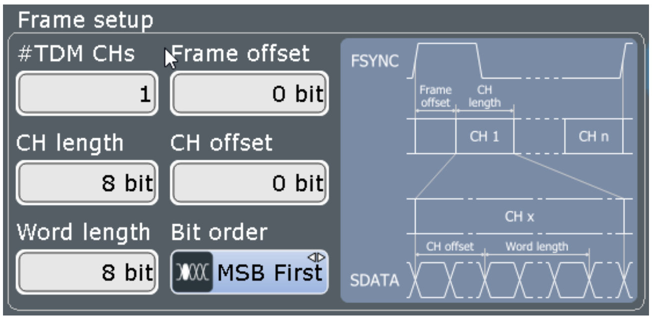
Specific settings for I²S standard signals are:

- "Channel order" on page 604
- "Word length" on page 604
- "Bit order" on page 604
- For left-justified data: "Channel offset" on page 604
- For right-justified data: "CH length" on page 604



Specific settings for TDM audio signals are:

- "Word length" on page 604
- "Bit order" on page 604
- "Channel offset" on page 604
- "#TDM CHs" on page 604
- "Frame offset" on page 604
- "CH length" on page 604



Make sure that the tab of the correct serial bus is selected on the left side.

Audio Variant
Selects the protocol variant of the audio signal. The configuration possibilities exceed the definitions of the standards.

"I2S Standard" Inter-IC Sound standard audio format.
It uses the SCLK, WS and SDATA lines. The first byte of the audio word is delayed one clock period from the leading edge of the word select pulse.

"Left justified" The left-justified data format uses the same lines as I²S standard. The first byte of the audio word is aligned with the leading edge of the word select pulse, or left justified within the frame. The format is word-length independent.

"Right justified" The right-justified data format is similar to the left-justified, but the last byte of the last word in the frame is aligned with the trailing edge of the word select pulse, or right-aligned within the frame. This format is not word-length independent.

"TDM" The Time Division Multiplexed audio format is not standardized and provides high flexibility for transfer of up to 8 audio data channels on one line. Instead of words select, it uses frame synchronization pulses to identify the beginning of a frame. On the data line, channel blocks of a defined length are transmitted. Each block contains an audio word followed by several zero bits to complete the block.

Remote command:
`BUS<m>:I2S:AVARiant` on page 1554

SCLK
Selects the source of the clock line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:I2S:CLOCk:SOURce` on page 1554

SCLK Polarity
Sets th polarity of the clock signal, that is the edge at which the instrument samples the data on the data line. Usually, the rising edge is used. The R&S RTE can also analyze the converse setup.

Remote command:
`BUS<m>:I2S:CLOCk:POLarity` on page 1555

WS / FSYNC
Selects the source of the word select line for I²S standard, left- and right-justified data formats, or the source of the frame synchronization pulse for TDM audio signals. The same waveforms as for SCLK are available, and the same restrictions are applied.

Remote command:
`BUS<m>:I2S:WSELect:SOURce` on page 1555

WS / FSYNC Polarity
For a word select line, the polarity defines the word select values assigned to the left and right channels.
- "Normal": usually, 0 indicates the left channel, and 1 indicates the right channel.
- "Inverted": 0 indicates the right channel, and 1 the left channel.

For an FSYNC line (TDM), the polarity defines the edge of the FSYNC pulse that identifies the beginning of a frame. The frame starts exactly at the next clock edge following the selected FSYNC edge.
- "Normal": usually, the frame begins with a rising edge.
- "Inverted": the frame begins with a falling edge.

Remote command:
`BUS<m>:I2S:WSELect:POLarity` on page 1555

SDATA
Selects the source of the audio data line. The same waveforms as for SCLK are available, and the same restrictions are applied.

Remote command:
`BUS<m>:I2S:DATA:SOURce` on page 1556

SDATA Polarity
Defines the interpretation of high and low signal states.
- "Active high": HIGH (signal level above the threshold level) = 1 and LOW (signal level below the threshold level) = 0
- "Active low": HIGH = 0 and LOW = 1

Remote command:
`BUS<m>:I2S:DATA:POLarity` on page 1556

Threshold setup
Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the polarity.

There are three ways to set the threshold:
- "Threshold"
  Enter individual values for each line directly in the fields.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
  This option is only available for analog sources.
- "Couple thresholds"
  Sets all thresholds to the same value. Enter the value in the "Threshold" field.

Remote command:
`BUS<m>:I2S:TCOupling` on page 1556
`BUS<m>:I2S:CLOCk:THReshold` on page 1557
`BUS<m>:I2S:DATA:THReshold` on page 1557

`BUS<m>:I2S:WSELect:THReshold` on page 1557
`BUS<m>:SETReflevels` on page 1383

Channel order
Defines if the left or the right channel is the first channel in the frame.

The setting is not available for TDM audio signals.

Remote command:
`BUS<m>:I2S:CHANnel:ORDer` on page 1557

Word length
Defines the number of bits in an audio data word. The minimum length is 4 bit, the maximum is 32 bit.

Remote command:
`BUS<m>:I2S:WLENgth` on page 1558

Bit order
Sets the bit order in the audio data words. Usually, the MSB is transmitted first.

Remote command:
`BUS<m>:I2S:BORDer` on page 1558

Channel offset
Sets the number of bits between the channel start and the start of the audio word. The setting is available for left-justified data format and TDM audio signals.

For TDM, possible values depend on the channel size and the word size. The maximum delay is *Channel length - Word length*.

Remote command:
`BUS<m>:I2S:CHANnel:OFFSet` on page 1558

#TDM CHs
Sets the number of channels transmitted on the TDM audio line.

Remote command:
`BUS<m>:I2S:CHANnel:TDMCount` on page 1559

Frame offset
Sets the number of bits between the frame start and the start of the first channel of a TDM audio line. The maximum offset is 256 bit. Each FSYNC edge restarts the offset count.

Remote command:
`BUS<m>:I2S:FOFFset` on page 1559

CH length
Sets the number of bits in a channel block for right-justified data format and TDM audio signals.

Remote command:
`BUS<m>:I2S:CHANnel:LENGth` on page 1559

12.8.2.2 Configuring Audio Protocol

The configuration of the Audio is simple - assign the two lines to input channels, and set the thresholds.

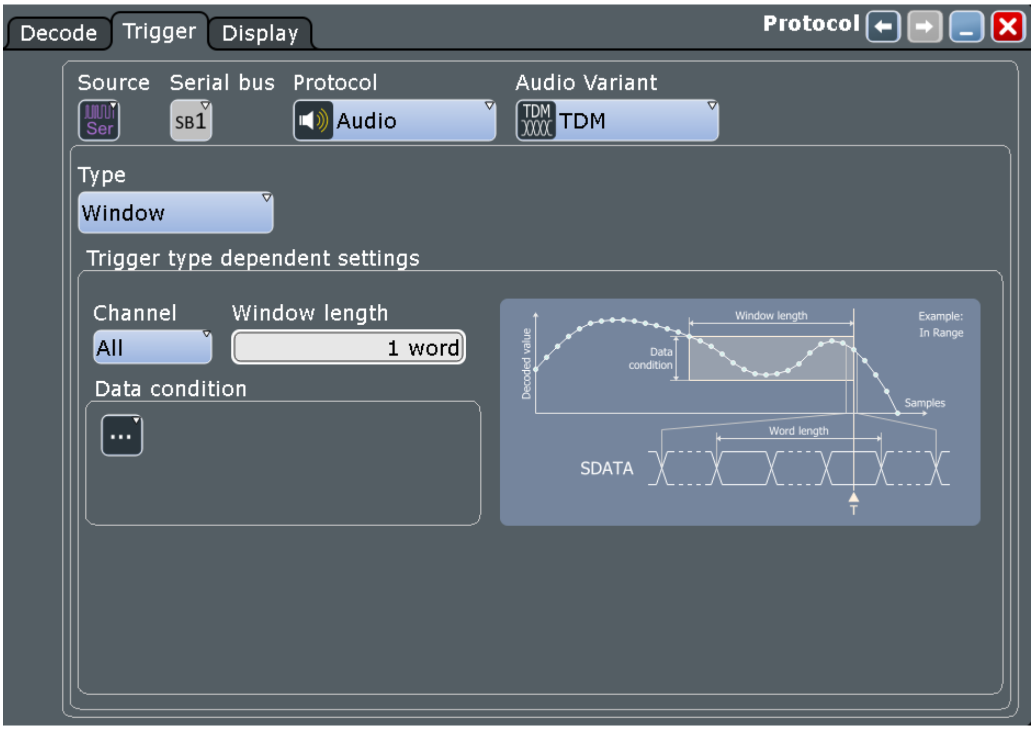
For details on configuration settings, see Chapter 12.8.2.1, "Audio Signal Configuration", on page 600.

1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "Audio".
5. Select the "Audio Variant".
6. Set the parameters, as required according to the selected "Audio Variant".
7. Optionally, you can enter a "Bus label" on the "Display" tab.
8. Enable "Decode", if available.

12.8.3 Audio Trigger

12.8.3.1 Audio Trigger

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = Audio"



Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:
`TRIGger<m>:SOURce:SBSelect` on page 1387

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1382

Type
Selects the trigger type.

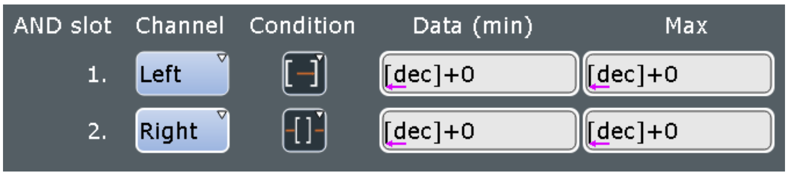
"Data"
: Sets the trigger on a data word or a data range that occurs on a specified channel or on any channel. The instrument triggers on the last bit of the specified data pattern.

Description of specific trigger type settings:
- "Channel" on page 608
- "Data condition" on page 608

"Window"
: This trigger checks if the decoded data values stay inside a "window" that is formed by a data range and a time specified by a number of subsequent words. It considers a selected channel or all channels. The instrument triggers at the end of the last word. Thus, for example, you can trigger on a pause.

Description of specific trigger type settings:
- "Channel" on page 608
- "Data condition" on page 608
- "Window length" on page 609

"Frame condition"
: Sets the trigger on an AND combination of data conditions on different channels. The instrument triggers if all conditions are met inside one frame.



Description of specific trigger type settings:
- "Channel" on page 608
- "Data condition" on page 608

Description of specific trigger type settings: "Channel" on page 608 and .

"Word select"
: Triggers on the selected edge of the WS line, that is, on the beginning of the left or right channel (I²S, left- and right-justified). For TDM signals, it triggers on the selected edge of the FSYNC line - on the beginning of a TDM frame.
The trigger time is the first clock edge after the selected WS/FSYNC edge.
Description of specific trigger type settings: "Word select: Slope" on page 609.

"Error condition" The oscilloscope uses the WS or FSYNC line to monitor the channel and frame length. An error is detected when two consecutive frames have different length. The instrument triggers on the first clock edge after error detection.

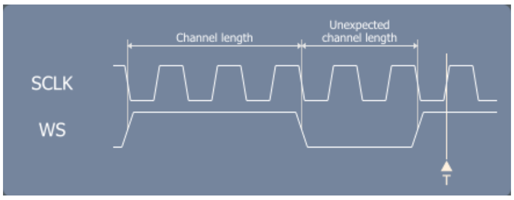


Figure 12-39: Trigger on errors in I²S standard signal with clock polarity "Rising"

Remote command:
TRIGger<m>:I2S:TYPE on page 1560

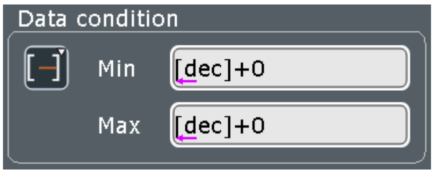
Channel
Selects the audio channel on which the instrument looks for the specified data condition.

The setting is relevant for trigger types Data, Window and Frame condition.

Note: For TDM signals. the number of available channels depends on the configuration of the audio bus, see "#TDM CHs" on page 604.

Remote command:
TRIGger<m>:I2S:TCONdition<n>:CHANnel on page 1561

Data condition
The data condition setup consists of the operator and one or two data patterns.



The settings are relevant for trigger types Data, Window and Frame condition.

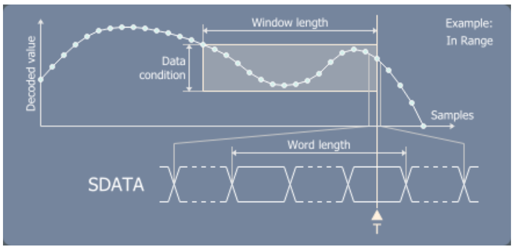
"Operator" Defines the operator to set a specific data word ("Equal" or "Not equal") or a data range.

"Min data" Defines the data pattern. The data length is limited to the word length. Enter the pattern using the bit order defined in the signal configuration. X (don't care) is not allowed.
Usually, audio words are signed numbers in 2's complement format. The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Max data" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:I2S:TCONdition<n>:CONDtion on page 1561
TRIGger<m>:I2S:TCONdition<n>:DMIN on page 1562
TRIGger<m>:I2S:TCONdition<n>:DMAX on page 1562

Window length
Sets the number of words that is used as time limit for the "Window" trigger type. The instrument triggers if the data condition is fulfilled on the same channel for the given number of subsequent frames.



Remote command:
`TRIGger<m>:I2S:SOWords` on page 1563

Word select: Slope
Sets the edge of the WS or FSYNC signal as trigger condition. The instrument triggers on the first clock edge after the specified edge.

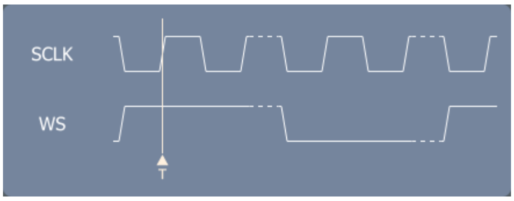


Figure 12-40: Word select trigger on I²S standard signal with clock polarity "Rising" and "Normal" WS polarity (left = 0)

The WS edge indicates the start of the left or right channel. The FSYNC edge indicates the frame start. Consider the WS / FSYNC Polarity setting in the "Protocol Configuration" dialog box.

Remote command:
`TRIGger<m>:I2S:WSSLope` on page 1563

12.8.3.2 Triggering on Audio Signals

Prerequisites: An Audio bus is configured, see Chapter 12.8.2.2, "Configuring Audio Protocol", on page 605.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to Audio.
5. Select the "Trigger type".
6. For more complex trigger types, enter the data, frame or word select conditions.

For details, see Chapter 12.8.3.1, "Audio Trigger", on page 605

12.8.4 Audio Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The decoding process considers the "Bit order" configuration setting of the signal and displays the binary result MSB first. Binary values in the combs of the decoded signal also consider the "Binary bit order" setting in the "Display" tab. Thus, you can read the bits of an LSB first signal in LSB first order in the combs while the results table displays the correct values MSB first.

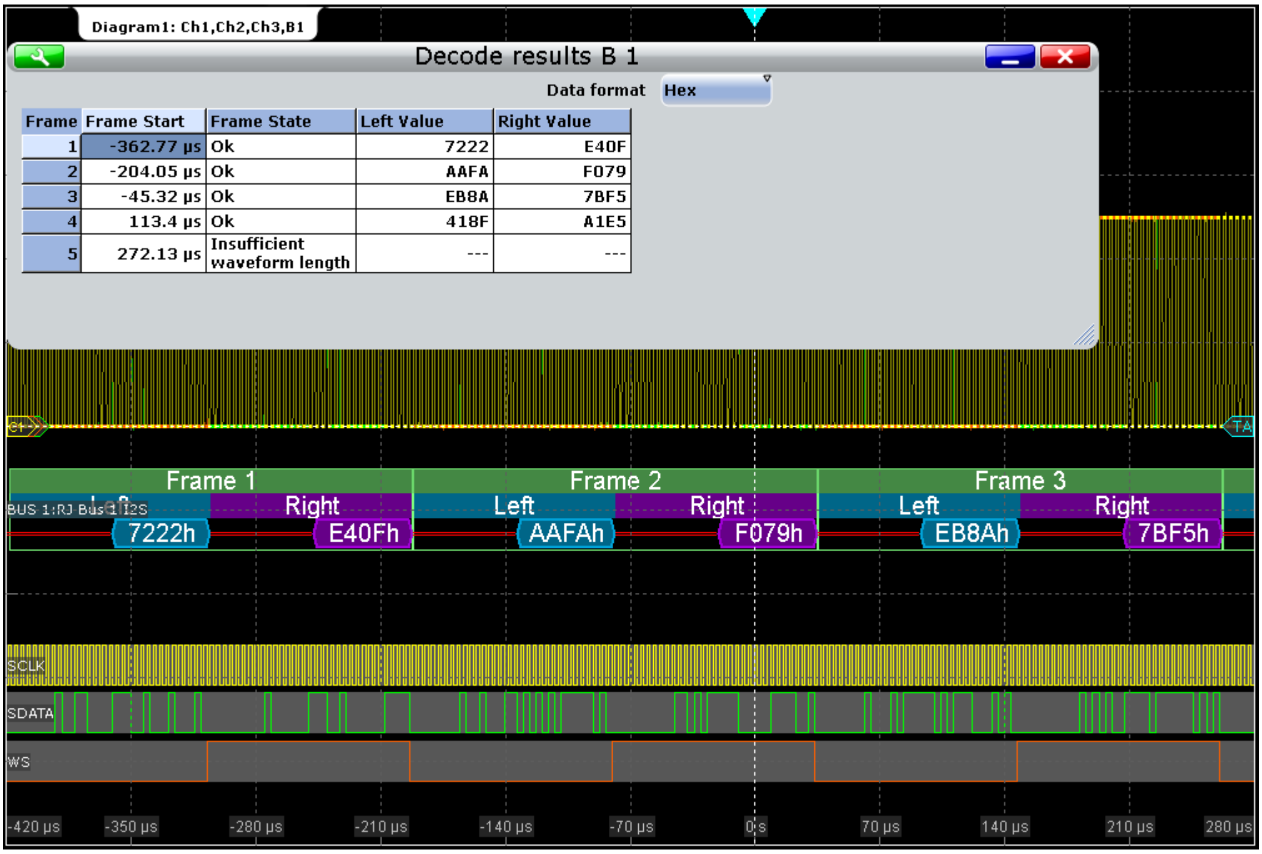


Figure 12-41: Decoded audio signal, right-justified data format

green = frame
blue = left channel
violet = right channel
orange = frame/channel is not completely contained in the acquisition
red = error

Table 12-9: Content of the "Decode results" table

Column	Description
Frame Start	Time of the frame start
Frame State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition.
Left Value	Data value of the left channel
Right Value	Data value of the right channel

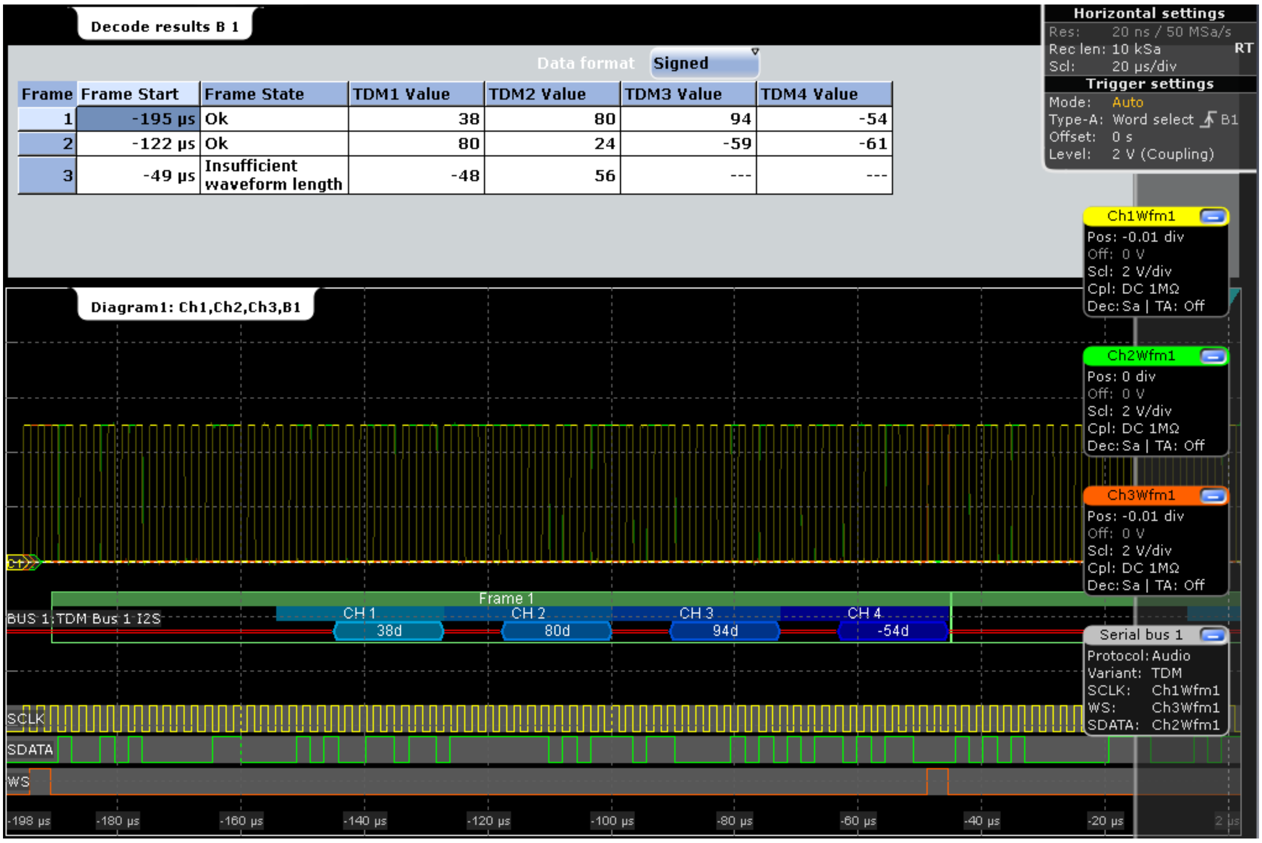


Figure 12-42: Decoded TDM signal with 4 channels, frame offset 16 bit, channel offset 4 bit, word length 8 bit and inverted FSYNC and SDATA polarity

Table 12-10: Content of the "Decode results" table

Column	Description
Frame Start	Time of the frame start
Frame State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition.
TDM<x> Value	Data value of the TDM channel

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.9.3, "Decode Results", on page 1563.

12.8.5 Track

The track is a waveform that shows measurement values in time-correlation to the audio signal. It is the graphical interpretation of all measurement values of a single acquisition. For audio signals, the measurement values on the vertical axis are the decoded values of the audio channels, the time scale is equivalent to the scale of the source waveforms.

You can display the values of several channels in one track, or create one track for each channel and display them in parallel.

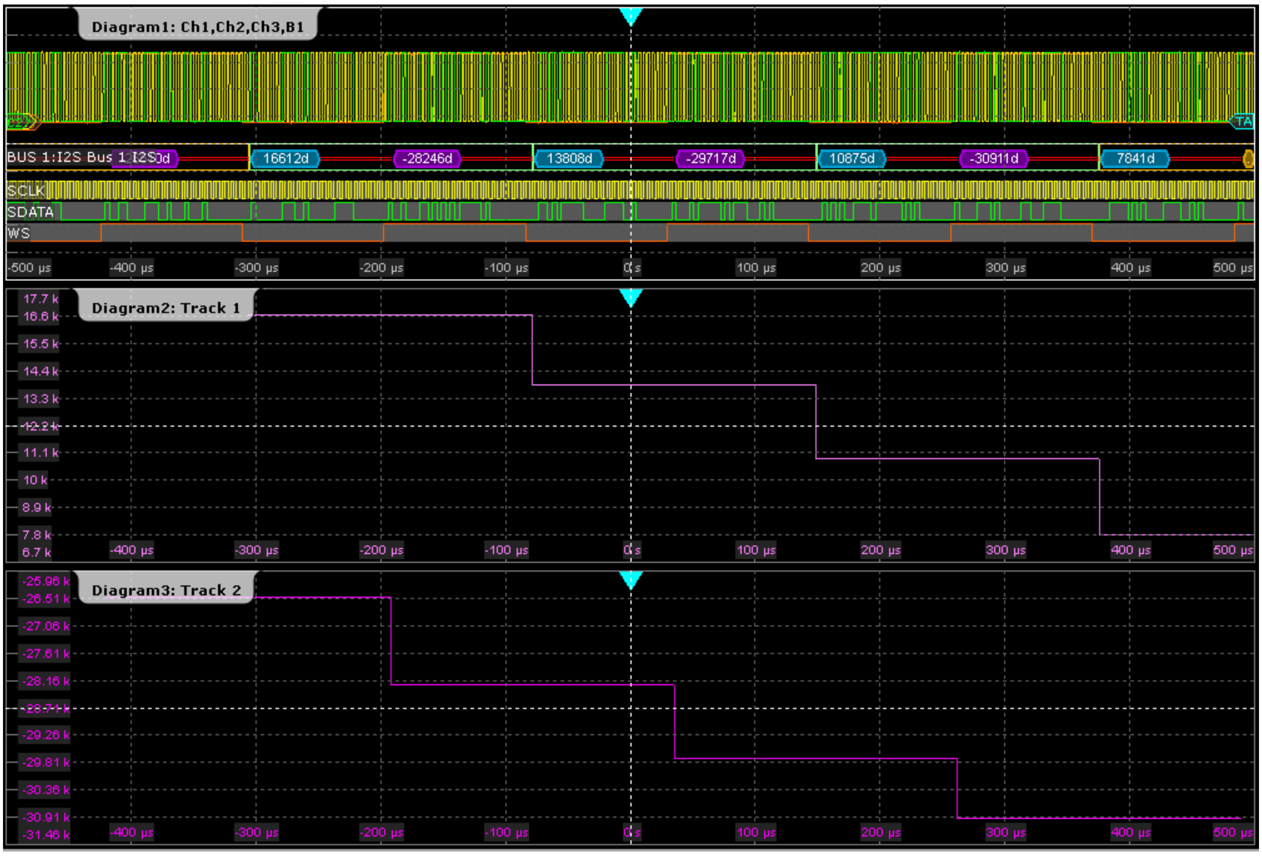


Figure 12-43: Tracks of left and right channel of I²S standard signal

The track is a special measurement waveform, so it can be used for further analysis like cursor measurements and zoom.

The instrument uses the bus data format to interpret audio data ("Display" tab > "Data format"). ASCII format is not supported.

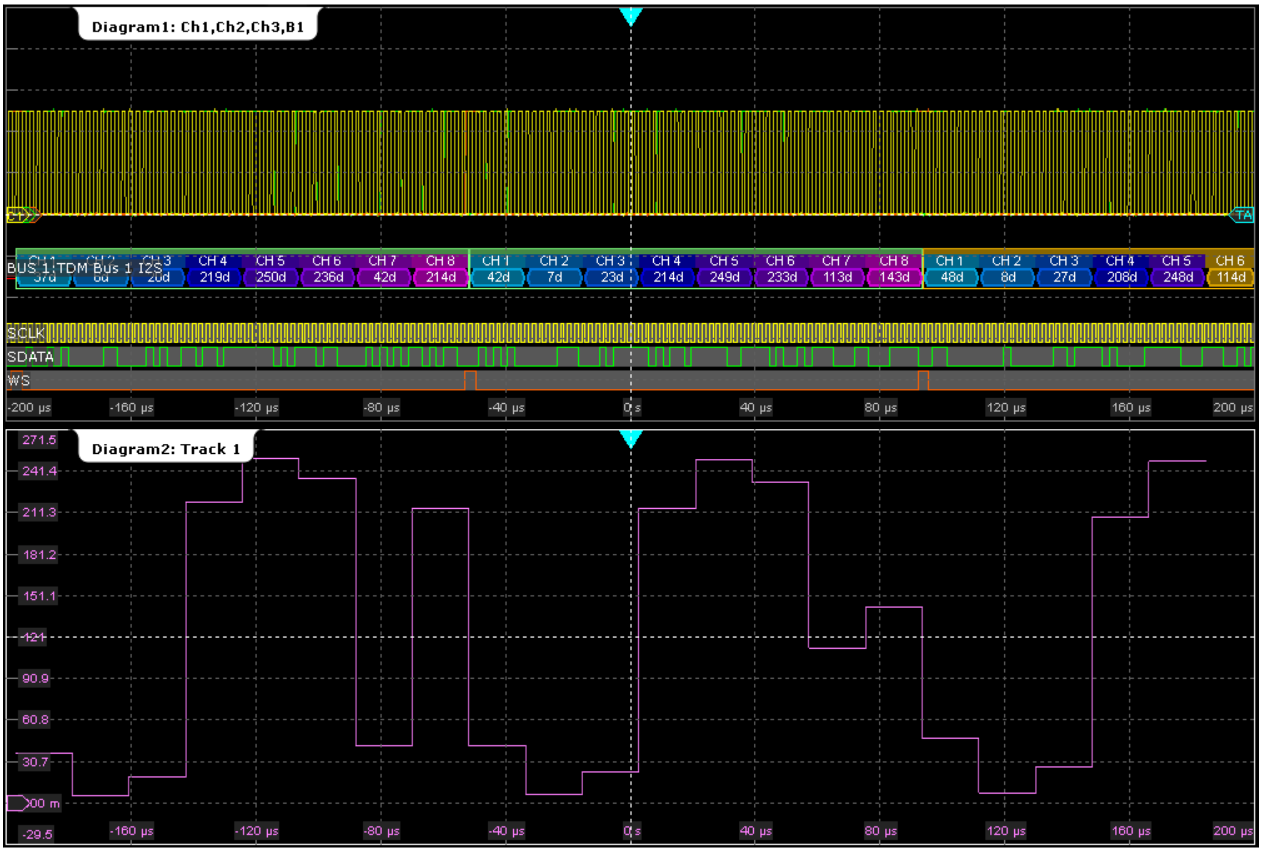


Figure 12-44: Track of all 8 channels of a TDM signal

12.8.5.1 Displaying and Configuring a Track

To get a first impression of the track, you can display it quickly. For further analysis, some configuration settings are available.

1. Press the [PROTOCOL] key.
2. Select the "Display" tab.
3. Tap the "Show Track Waveform(s)" button.
4. Select the "Audio channel" to be tracked.

   The track waveform with default settings is enabled and displayed.

   Tip: Alternatively, you can enable the track in the "Measurements" dialog box, on the "Result Analysis" tab if the protocol measurement is selected on the "Setup" tab.
5. If you want to change the track settings, tap "Track Setup".
6. By default, the track is displayed using "Continuous auto scale". If you want to change the scaling, proceed as follows:

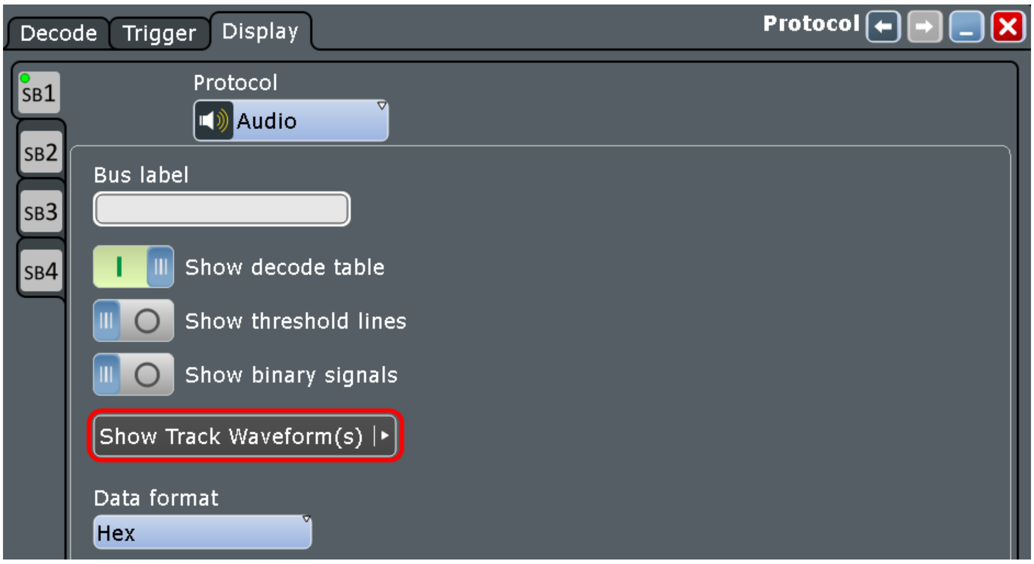
   a) Select the "Result Analysis" tab.

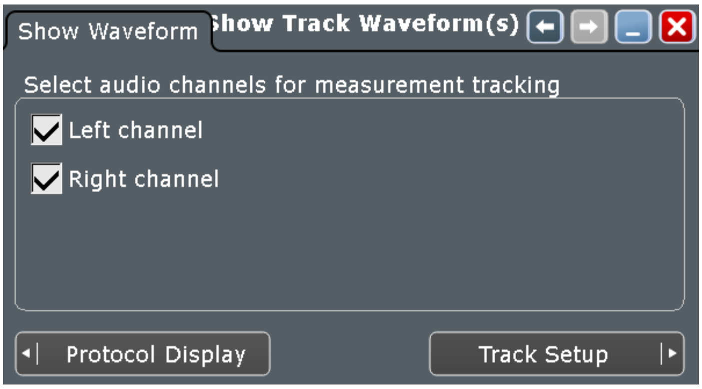
b) Disable "Continuous auto scale".
c) Adjust "Vertical scale" and "Vertical offset".

12.8.5.2 Track Settings in Protocol Setup

You can enable the track waveforms in the protocol display settings. As the track is based on measurement, you can use the "Measurement" dialog box alternatively, see Chapter 12.8.5.4, "Track Enabling in Measurement Setup", on page 616.

Access: [PROTOCOL] > "Display" tab





To set the vertical scale of the track waveform, use the measurement scale settings on the "Measurements" > "Result Analysis" tab, see "Analysis scale" on page 350.

Show Track Waveform(s)
Enables and displays the tracks for the selected channels of the decoded bus using continuous auto scale.

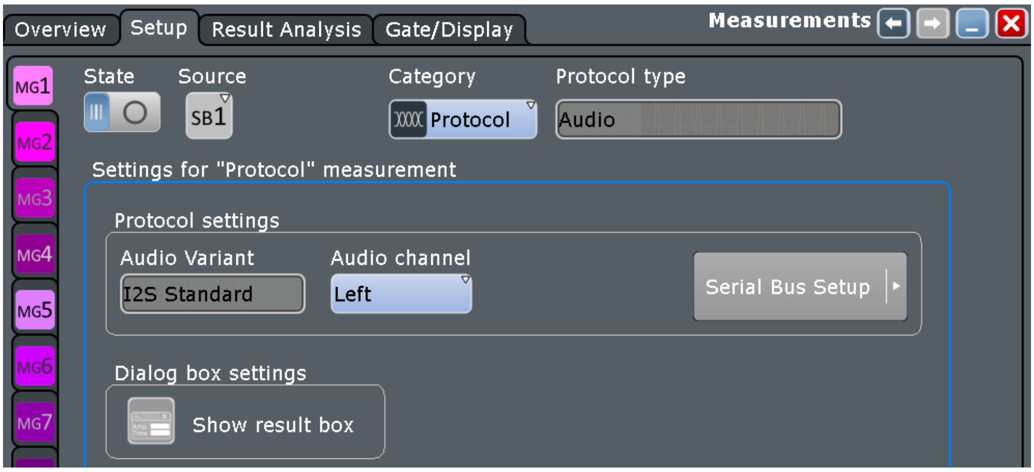
See also: Chapter 12.8.5, "Track", on page 612.

Remote command:
BUS<m>:I2S:TRACk:LEFT on page 1567
BUS<m>:I2S:TRACk:RIGHt on page 1567
BUS<m>:I2S:TRACk:TD1Ch on page 1567
BUS<m>:I2S:TRACk:TD2Ch on page 1567
BUS<m>:I2S:TRACk:TD3Ch on page 1567
BUS<m>:I2S:TRACk:TD4Ch on page 1567
BUS<m>:I2S:TRACk:TD5Ch on page 1567
BUS<m>:I2S:TRACk:TD6Ch on page 1567
BUS<m>:I2S:TRACk:TD7Ch on page 1567
BUS<m>:I2S:TRACk:TD8Ch on page 1567

12.8.5.3 Track Settings in Measurement Setup

As track and trend are based on measurements, the main settings are available in the "Measurement" dialog box.

Access: "Meas" menu > "Setup" > "Protocol" category



Audio variant
Shows the current audio protocol for information.

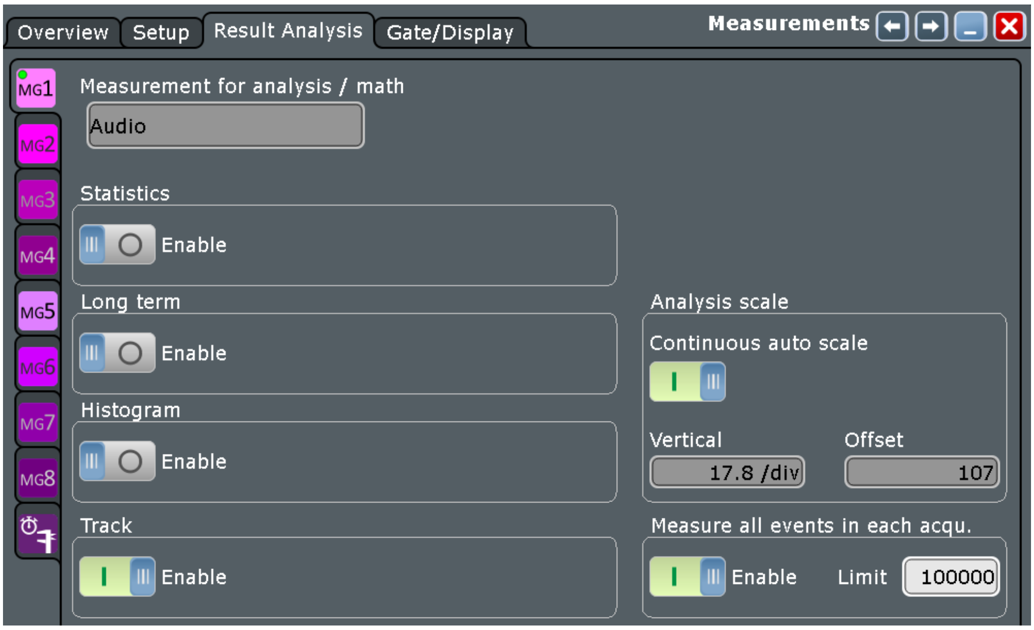
Audio channel
Selects the channel that is shown in the track and trend waveforms.

Show result box
Hides or shows the measurement result box. For track and trend, no numerical results are available in the result box, so you can hide it.

12.8.5.4 Track Enabling in Measurement Setup

As the track is based on measurement, it can be set up in the "Result Analysis" tab of the "Measurement" dialog box. Alternatively, you can enable the track in the protocol display settings, see Chapter 12.8.5.2, "Track Settings in Protocol Setup", on page 615.

Access: "Meas" menu > "Result Analysis"



To set the vertical scale of the track waveform, use analysis scale settings, see "Analysis scale" on page 350.

Enable (Track)
Enables the track of measurement results over time and displays the track waveform. It is the graphical interpretation of all measurement values of a single acquisition.

The track is available for most amplitude/time measurements (except for High, Low, Amplitude, Max, Min, Peak to peak, Mean, RMS, S-dev, Pos. and Neg. overshoot, and Area), and for jitter measurements.

Enabling the track enables also the Continuous auto scale and Measure all events in each acquisition.

Before you can enable the track, activate the appropriate measurement.

With option R&S RTE-K5 I²S audio signals, you can use the track for protocol measurements on decoded audio buses, see Chapter 12.8.5, "Track", on page 612.

Remote command:
`MEASurement<m>:TRACk[:STATe]` on page 1264

12.8.6 Trend

The trend is a special long-term measurement that shows the evolution of measurement values in a running continuous acquisitions. For audio signals, each decoded channel value is a measurement result that creates a point on the trend curve. You can configure the number of points that builds the complete trend curve.

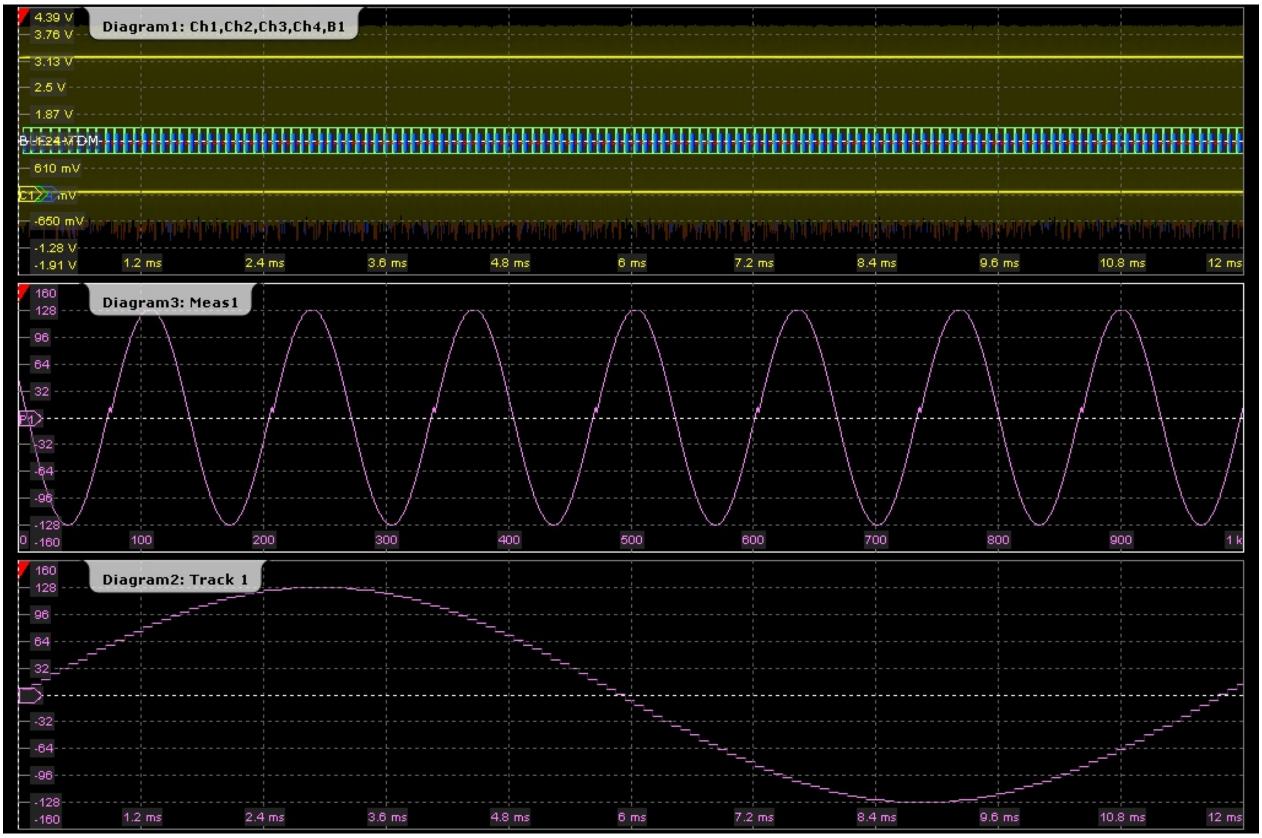
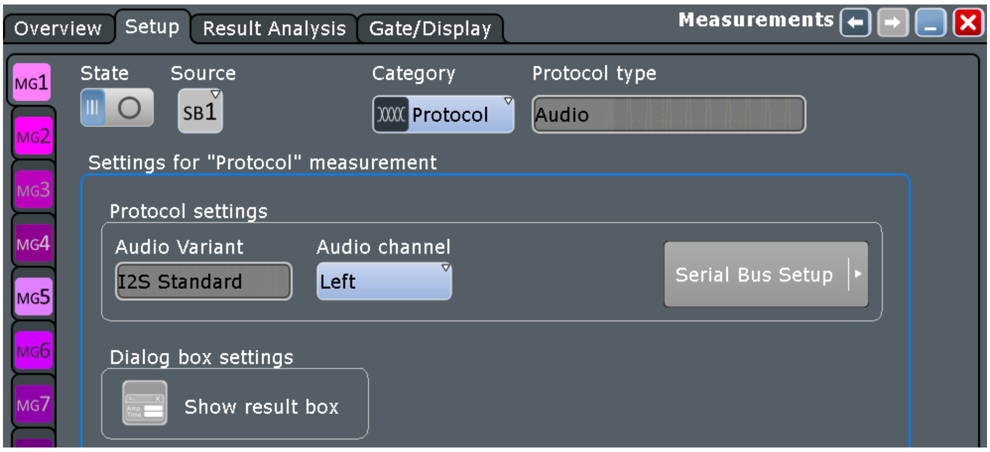


Figure 12-45: Trend (Diagram3) and track of an audio signal

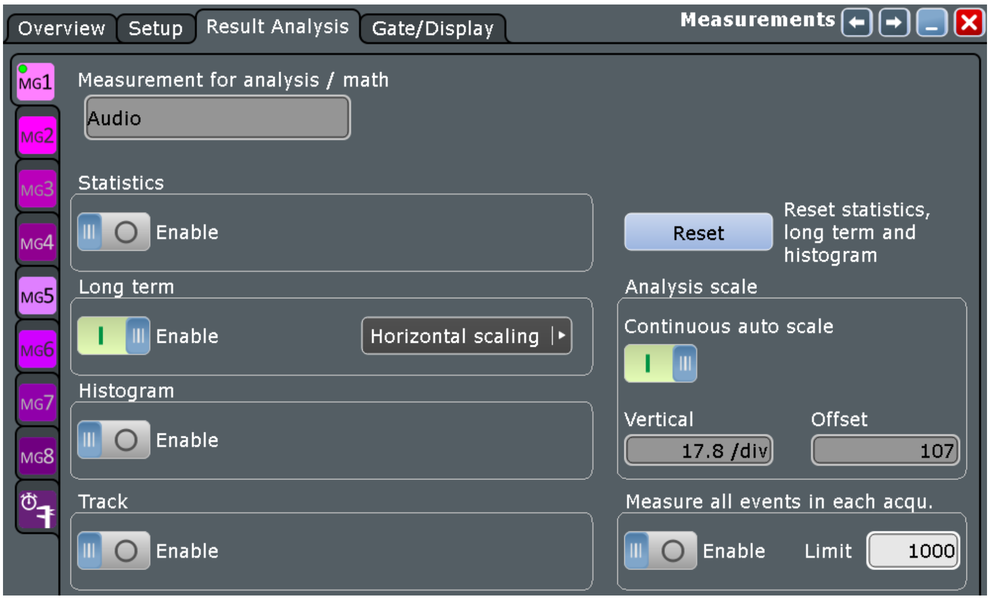
12.8.6.1 Displaying and Configuring the Trend

If an audio bus is configured, you can set up the trend measurement for audio channels. The following procedure describes the complete trend setup using the "Meas" menu.

1. On the "Meas" menu, select "Setup".
2. Select the "Category" = "Protocol".
3. Select the "Source" of the measurement: "Serial bus". Select the bus configured for the audio signal.
4. Enable "State".
5. Select the "Audio channel" for which you want to analyze the trend.

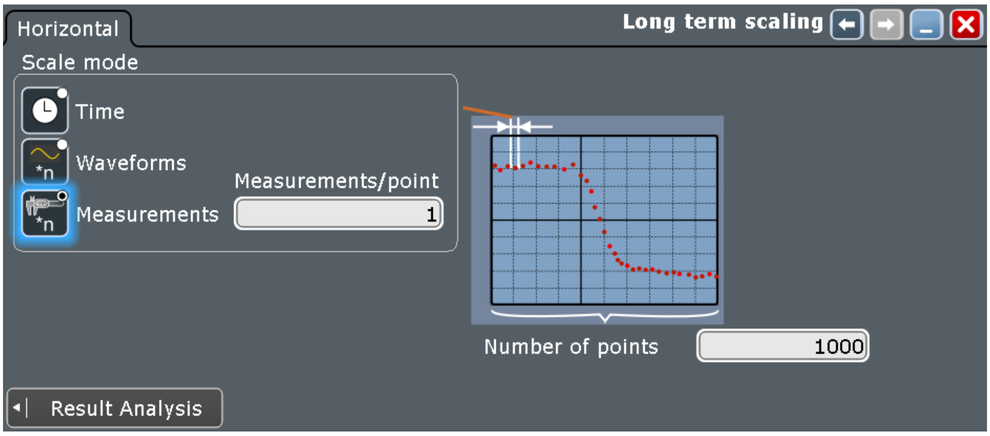


6. Select the "Result Analysis" tab.
7. Under "Long term", select "Enable".



An empty trend diagram is displayed.

8. Tap "Horizontal scaling".
9. Set the scale mode to "Measurements" and the number of "Measurements/point" to *1*.



10. Tap "Result Analysis".
11. Set the "Vertical scaling" to "Continuous auto scale".
12. Start continuous acquisition.

    The trend fills up with measurement points from left to right.
13. If you want to change the vertical scaling of the trend curve, disable "Continuous auto scale" and adjust "Vertical scale" and "Vertical offset".

12.9 MIL-1553 (Option R&S RTE-K6)

12.9.1 The MIL-STD-1553

The MIL-STD-1553 specification defines the characteristics of a serial data bus originally designed for use in the military avionics. Nowadays it is also used in spacecraft on-board data handling.

The bus is a 2-wire bus that uses differential signals.

A MIL-STD-1553 system consists of the following components:

- Bus Controller (BC): initiates and coordinates the data flow in the system.
- Remote Terminal (RT): interfaces various subsystems with the data bus. A system consists of up to 31 RTs and each RT can have 31 subaddresses.
  The subaddresses 0 and 31 refer to a mode code command.
- Bus Monitor (BM) (optional): listens to all messages and can record selected data for real-time or off-line analysis.

The information is transmitted over the bus in defined series of words using Manchester code, where each bit is transmitted as high-low for a logical 1 or a low-high for a logical 0. There are three types of words: command, data and status.

Command Word

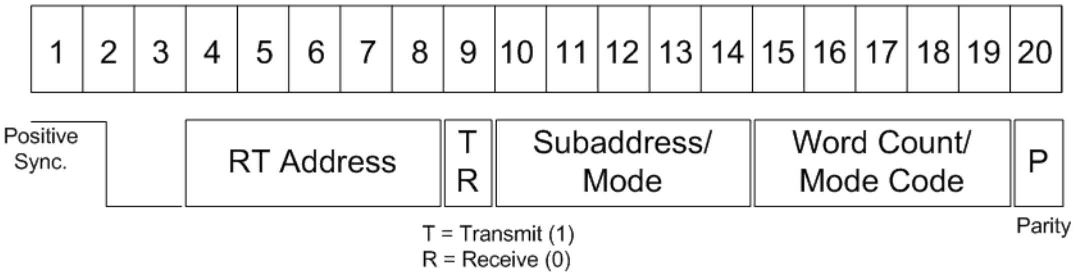


Figure 12-46: Structure of a command word

The format of a command word consists of the following parts (see Figure 12-46):

- Sync: an invalid Manchester waveform.
- Remote Terminal (RT) Address: the unique address of the corresponding RT.
- Transmit/Receive (T/R): indicates the action required from the RT.
- Subaddress/Mode Code: indicates the RT subaddress. The subaddresses 0 and 31 signalize the transmission of a mode code.
- Data Word Count /Mode Code: indicates the number of words that are sent/ received by the RT. A maximum of 32 words is allowed. This field may be used for the transmission of the mode code value.
- Parity: checks if there are bit errors during the transmission. The total number of logic 1 bits for the word (sync bits not included) shall be odd.

Data Word

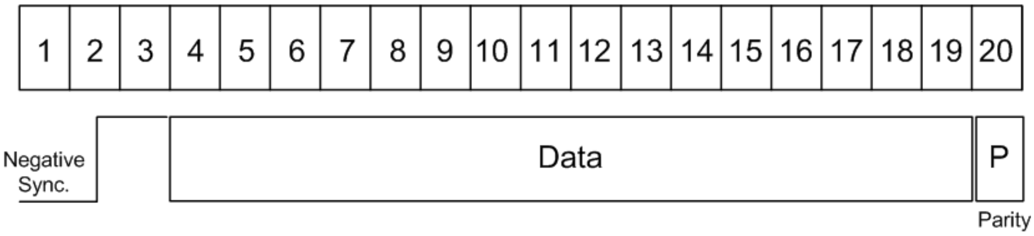


Figure 12-47: Structure of a data word

The format of a data word consists of the following parts (see Figure 12-47):

- Sync: an invalid Manchester waveform.
- Data: the transferred information (16 bit).
- Parity: checks if there are bit errors during the transmission. The total number of logic 1 bits for the word (sync bits not included) shall be odd.

Status Word

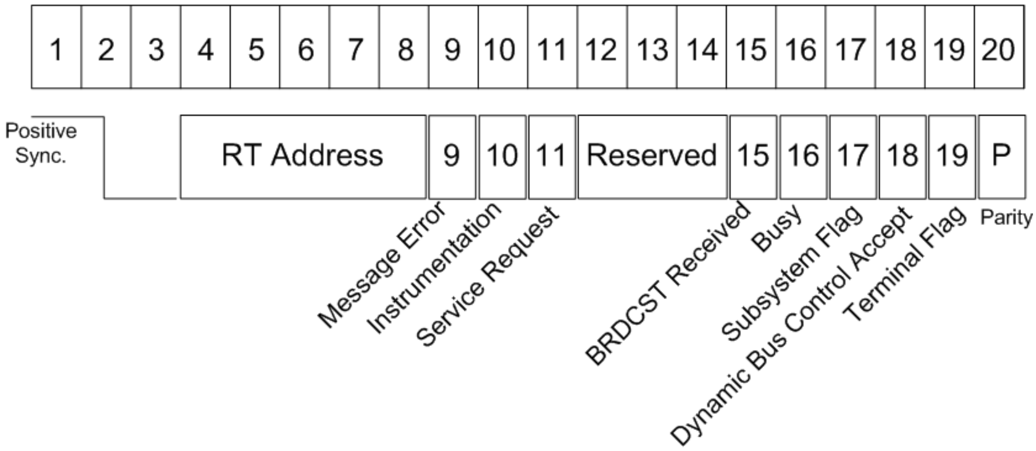


Figure 12-48: Structure of a status word

The format of a status word consists of the following parts (see Figure 12-48):

- Sync: an invalid Manchester waveform.
- Remote Terminal (RT) Address: the unique address of the corresponding RT.
- Message error: indicates an error in the command/data word transmission from the BC. A logic 1 indicates presence of a message error and a logic 0 indicates its absence.
- Instrumentation: helps to distinguish between a status word and a command word. The logic state of this bit shall be 0.
- Service Request: indicates that the RT requires service. A logic 1 indicates a presence of a service request and logic 0 indicates its absence.
- Reserved: bits reserved for future uses.
- Broadcast Command: a logic 1 indicates that the preceding valid command word was a broadcast command and a logic 0 that it was not.
- Busy: a busy state indicates that the RT or the subsystem is not able to transfer data. A logic 1 indicates a presence of a busy condition and logic 0 indicates its absence.
- Subsystem Flag: flags a subsystem fault. A logic 1 indicates a presence of a flag and logic 0 indicates its absence.
- Dynamic Bus Control Acceptance: a logic 1 indicates acceptance of a dynamic bus control and a logic 0 a rejection.
- Terminal Flag: flags an RT fault condition. A logic 1 indicates a presence of a flag and logic 0 indicates its absence.
- Parity: checks if there are bit errors during the transmission. The total number of logic 1 bits for the word (sync bits not included) shall be odd.

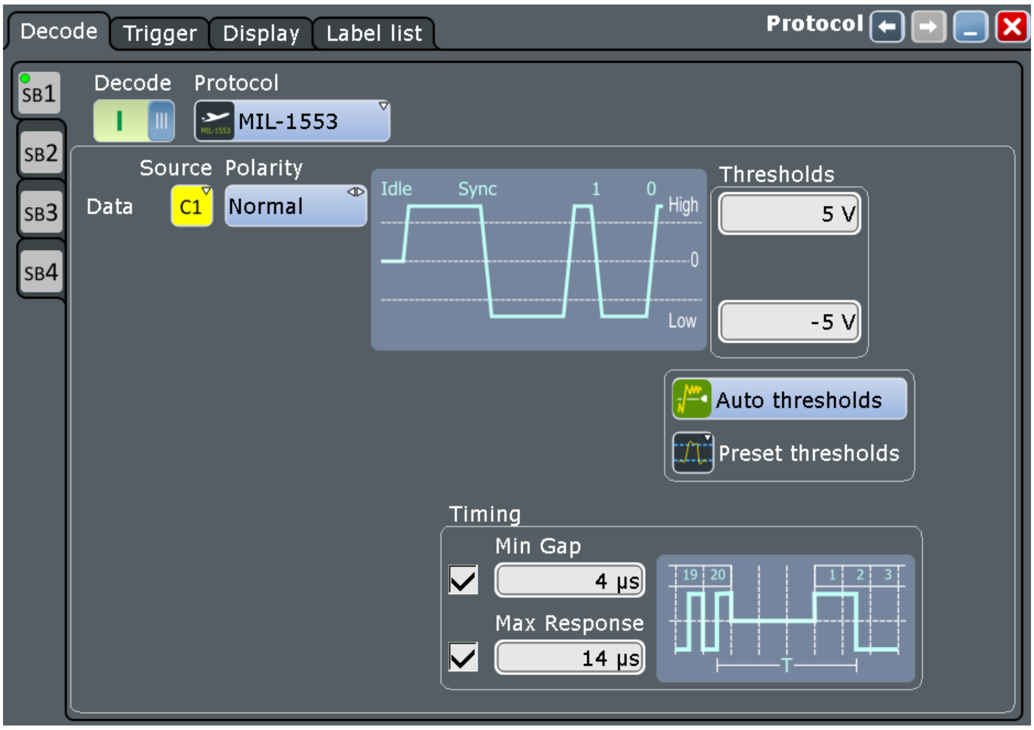
For comfortable analysis, you can load an editable label list, to interpret transferred numeric values as meaningful text labels.

12.9.2 MIL-STD-1553 Configuration

12.9.2.1 MIL-STD-1553 Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *MIL-STD-1553*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Data

Sets the source of the selected data line. Analog channels, math waveforms, and reference waveforms can be used.

For triggering on a serial bus, a channel signal is required.

Remote command:
`BUS<m>:MILStd:SOURce` on page 1569

Polarity

Selects the wire on which the bus signal is measured : "Normal" or "Inverted". The setting affects the digitization of the signal.

Remote command:
`BUS<m>:MILStd:POLarity` on page 1570

Thresholds
Threshold values are used for digitization of the signal.

Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the threshold.

There are three ways to set the thresholds:
- "High" and "Low"
  Upper and lower threshold levels. You can enter the values directly in the fields.
- "Preset thresholds"
  Selects the default threshold voltage from a list. The value is set to "Manual" if the threshold was set with "Auto thresholds", or was entered directly.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:MILStd:THReshold:HIGH` on page 1570
`BUS<m>:MILStd:THReshold:LOW` on page 1571
`BUS<m>:MILStd:PRESet` on page 1570
`BUS<m>:SETReflevels` on page 1383

Min Gap
Selects and sets a value for the intermessage gap beween the last bit of a message and the following command word sync. The time is measured between the mid bit zero crossings. According to the standard, the minimum idle time is 4 µs.

The minimum gap time is relevant for protocol configuration and error trigger.

If "Min Gap" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shrter than specified.

Remote command:
`BUS<m>:MILStd:MINGap:BITS` on page 1570
`BUS<m>:MILStd:MINGap:SELect` on page 1569
`TRIGger<m>:MILStd:MINGap:BITS` on page 1579
`TRIGger<m>:MILStd:MINGap:SELect` on page 1579

Max Response
Selects and sets a value for the maximum response time between the last bit of a word and the following status word sync. The time is measured between the mid bit zero crossings. According to the standard, the RT shall respond to a valid command word within the time period of 4 to 12 µs.

The max response time is relevant for protocol configuration and error trigger.

If "Max response" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the response time is longer than specified.

Remote command:
`BUS<m>:MILStd:MAXResponse:BITS` on page 1569
`BUS<m>:MILStd:MAXResponse:SELect` on page 1569
`TRIGger<m>:MILStd:MAXResponse:BITS` on page 1578
`TRIGger<m>:MILStd:MAXResponse:SELect` on page 1579

12.9.2.2 Configuring MIL-STD-1553

For configuration assign the line to the input channel, set the threshold and the timing conditions.

For details on configuration settings, see Chapter 12.9.2.1, "MIL-STD-1553 Configuration Settings", on page 623.

To display the decoded signal, option R&S RTE-K6 is required.

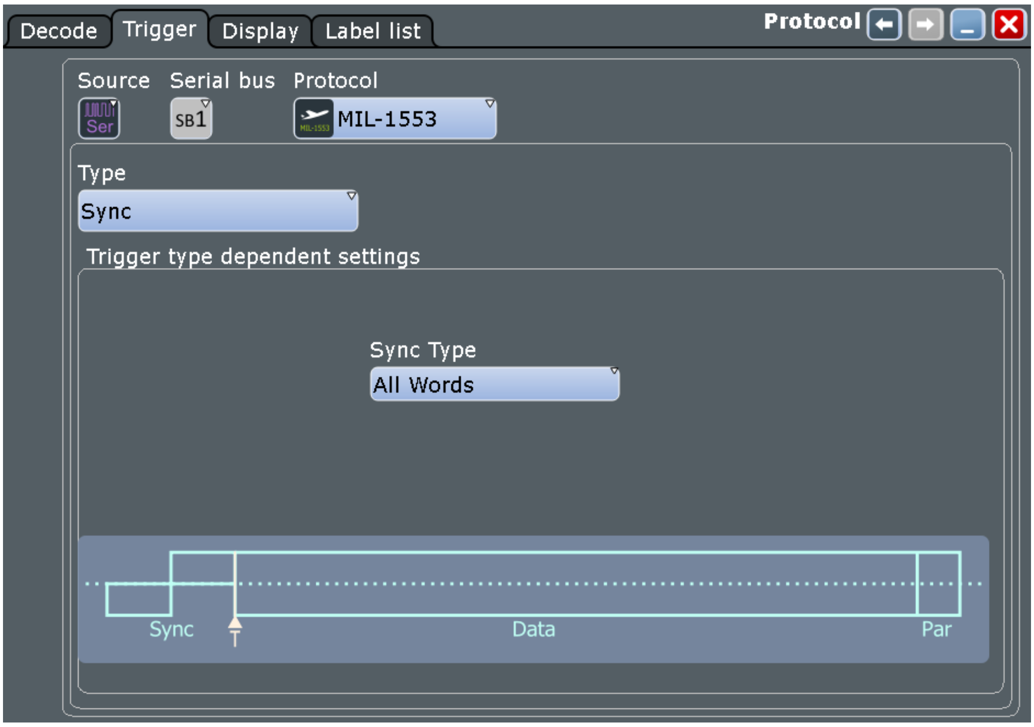
1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "MIL-STD-1553".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Tap the "Polarity" button, and select the waveform of the data line.
7. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to the middle reference levels with "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.
8. If necessary, tap the "Min Gap" button to select it and set the minimum gap time.
9. If necessary, tap the "Max Response" button to select it and set the maximum response time.

12.9.3 MIL-STD-1553 Trigger

12.9.3.1 Trigger Settings MIL-STD-1553

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = MIL-1553"

Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.



Trigger Type

Selects the trigger type for MIL-STD-1553 analysis.

"Sync" Triggers on a sync impulse.

"Word" Triggers on the selected word type.

"Data Word" Triggers on a specified data word or data word range.

"Command/ Status Word"
Triggers on a specified command word or on a status word.

"Command Word"
Triggers on a specified command word.

"Status Word" Triggers on a specified status word.

"Error Condition" Triggers on any combination of protocol errors.

Remote command:
`TRIGger<m>:MILStd:TYPE` on page 1572

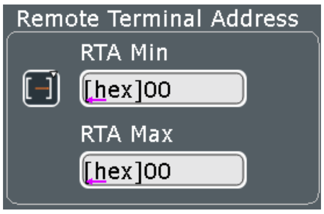
Sync Type / Word Type

Triggers on a sync impulse or word type. You can select to trigger on "Command/ Status", on "All" or on "Data" sync pulses / word types.

Remote command:
`TRIGger<m>:MILStd:TPSPecifier` on page 1581

Remote Terminal Address

The RTA setup consists of the condition and one or two RTA patterns.

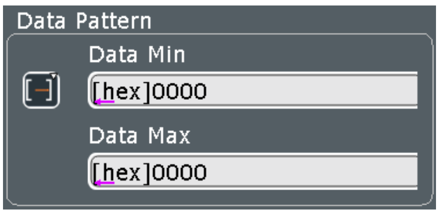


"Condition"	Defines the operator to set a specific RTA ("Equal" or "Not equal") or an RTA range.
"RTA Min/ RTA"	Defines the bit pattern of the RTA. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"RTA Max"	The second RTA pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:MILStd:CDST:RCONdition` on page 1573
`TRIGger<m>:MILStd:CDST:RMAX` on page 1573
`TRIGger<m>:MILStd:CDST:RMIN` on page 1573
`TRIGger<m>:MILStd:CMD:RCONdition` on page 1573
`TRIGger<m>:MILStd:CMD:RMAX` on page 1573
`TRIGger<m>:MILStd:CMD:RMIN` on page 1573
`TRIGger<m>:MILStd:DATA:RCONdition` on page 1573
`TRIGger<m>:MILStd:DATA:RMAX` on page 1573
`TRIGger<m>:MILStd:DATA:RMIN` on page 1573

Data Pattern

The data pattern setup consists of the condition and one or two data patterns.



"Condition"	Defines the operator to set a specific data pattern ("Equal" or "Not equal") or a data patter range.

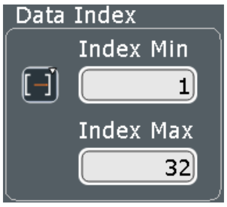
"Data Min/ Data" Defines the bit pattern of the data pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:MILStd:DATA:DCONdition` on page 1576
`TRIGger<m>:MILStd:DATA:DMAX` on page 1577
`TRIGger<m>:MILStd:DATA:DMIN` on page 1577

Data Index

The MIL-MIL-STD-1553 standard defines the length of a message to a series of up to 32 words. Data index sets the range within this series of the data words that is considered for the analysis. The data index setup consists of the condition and one or two data index values.



"Condition" Defines the operator to set a specific data ("Equal") or a data range.

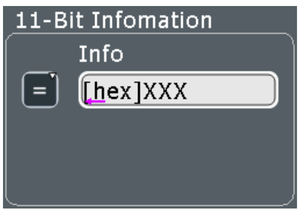
"Index Min/Index" Defines the minimum index.

"Index Max" The second data pattern is required to specify a range with conditions "In range".

Remote command:
`TRIGger<m>:MILStd:DATA:ICONdition` on page 1577
`TRIGger<m>:MILStd:DATA:IMAX` on page 1577
`TRIGger<m>:MILStd:DATA:IMIN` on page 1578

11-Bit Information

The 11-Bit information sets bits 9 to 19 if there is a command or status word. The 11-Bit information consists of the condition and one or two 11-Bit information patterns.



"Condition" Defines the operator to set a specific info ("Equal" or "Not equal") or an info range.

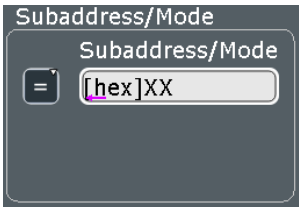
"Info Min/Info" Defines the bit pattern of the 11-Bit information.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Info Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:MILStd:CDST:ICONdition on page 1576
TRIGger<m>:MILStd:CDST:IMAX on page 1576
TRIGger<m>:MILStd:CDST:IMIN on page 1576

Subaddress/ Mode

The subaddress/mode setup consists of the condition and one or two subaddress/mode patterns.



"Condition" Defines the operator to set a specific subaddress/mode ("Equal" or "Not equal") or a subaddress range.

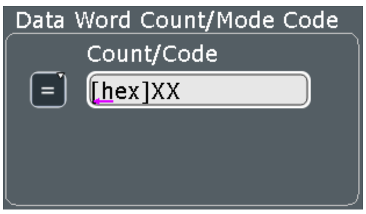
"Subaddress Min / Subaddress/Mode"
Defines the bit pattern of the subaddress/mode.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Subaddress Max" The second subaddress/mode pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:MILStd:CMD:SCONdition on page 1574
TRIGger<m>:MILStd:CMD:SMAX on page 1575
TRIGger<m>:MILStd:CMD:SMIN on page 1575

Data Word Count/Mode Code

The data word count/ mode code setup consists of the condition and one or two patterns.



"Condition" Defines the operator to set a specific data word count/ mode code ("Equal" or "Not equal") or a range.

"Word Count Min/ Count Code"
: Defines the bit pattern of the data.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Word Count Max"
: The second data pattern is required to specify a range with conditions "In range" and "Out of range".

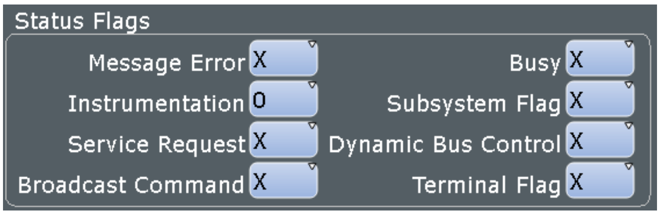
Remote command:
TRIGger<m>:MILStd:CMD:CCONdition on page 1574
TRIGger<m>:MILStd:CMD:CMAX on page 1574
TRIGger<m>:MILStd:CMD:CMIN on page 1574

T/R (Transmit/receive)
Toggles the data direction of the selected command: 1 (transmit), 0 (receive), or X (either).

Remote command:
TRIGger<m>:MILStd:CMD:TR on page 1575

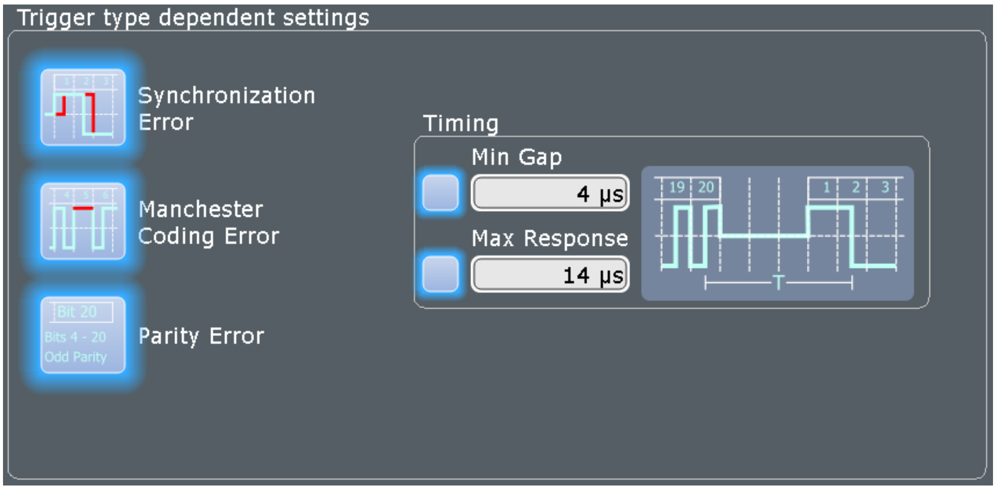
Status Flags
Specifies the values of the status flags. You can use the following characters: 1; 0; or X (do not care).

For details, see "Status Word" on page 622.



Remote command:
TRIGger<m>:MILStd:STATus:BCReceived on page 1579
TRIGger<m>:MILStd:STATus:BUSY on page 1580
TRIGger<m>:MILStd:STATus:DBCaccept on page 1580
TRIGger<m>:MILStd:STATus:INSTrument on page 1580
TRIGger<m>:MILStd:STATus:MERRor on page 1580
TRIGger<m>:MILStd:STATus:SREQuest on page 1580
TRIGger<m>:MILStd:STATus:SUBSystem on page 1581
TRIGger<m>:MILStd:STATus:TERMinal on page 1581

Error Condition
Specify the error conditions to be triggered on.



Synchronization Error ← Error Condition
Triggers if a sync impulse does not fulfill the technical requirements or when the transmission is not valid.

Remote command:
TRIGger<m>:MILStd:ERRor:SYNC on page 1578

Manchester Coding Error ← Error Condition
Triggers if there is an error in the Manchester coding of the signal.

Remote command:
TRIGger<m>:MILStd:ERRor:MANChester on page 1578

Parity Error ← Error Condition
Checks the parity of every word and triggers if the parity is even.

Remote command:
TRIGger<m>:MILStd:ERRor:PARity on page 1578

Min Gap ← Error Condition
Selects and sets a value for the intermessage gap beween the last bit of a message and the following command word sync. The time is measured between the mid bit zero crossings. According to the standard, the minimum idle time is 4 µs.

The minimum gap time is relevant for protocol configuration and error trigger.

If "Min Gap" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shrter than specified.

Remote command:
BUS<m>:MILStd:MINGap:BITS on page 1570
BUS<m>:MILStd:MINGap:SELect on page 1569
TRIGger<m>:MILStd:MINGap:BITS on page 1579
TRIGger<m>:MILStd:MINGap:SELect on page 1579

Max Response ← Error Condition
Selects and sets a value for the maximum response time between the last bit of a word and the following status word sync. The time is measured between the mid bit zero crossings. According to the standard, the RT shall respond to a valid command word within the time period of 4 to 12 µs.

The max response time is relevant for protocol configuration and error trigger.

If "Max response" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the response time is longer than specified.

Remote command:
`BUS<m>:MILStd:MAXResponse:BITS` on page 1569
`BUS<m>:MILStd:MAXResponse:SELect` on page 1569
`TRIGger<m>:MILStd:MAXResponse:BITS` on page 1578
`TRIGger<m>:MILStd:MAXResponse:SELect` on page 1579

12.9.3.2 Triggering on MIL-STD-1553

Prerequisites: A bus is configured for the MIL-STD-1553 signal to be analyzed.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to MIL-STD-1553.
5. Select the "Trigger type".
6. For more complex trigger types, enter the data pattern conditions.
   For details, see Chapter 12.9.3.1, "Trigger Settings MIL-STD-1553", on page 625.

12.9.4 MIL-STD-1553 Label List

Label lists are protocol-specific. A MIL-STD-1553 label file contains four values for each identifier:

- "RTA": hexadecimal remote terminal address value
- "Sub Addr": hexadecimal sub address value
- "Sub Address Label Name": the label name corresponding to the value of the subaddress.
- "Symbolic label": symbolic name of addressed device, specifying the device function, and the label of the sub address.

Example: MIL PTT file

```
# ------------------------------------------------------------
# Labels for MIL.1553 protocol
# Column order: RT address, RT label, Subaddress, Subaddress Label
# ----------------------------------------------------------------------
```

```
@PROTOCOL_NAME = mil1553
0Ah,Engine,01h,Thrust
03h,Main panel,07h,Altimeter
03h,Main panel,01h,Speed
0Eh,Only RTA
```

MIL-1553 Label List

RTA	Sub Addr	Sub address Label Name	Symbolic Label
[hex] 03	*		Main panel
[hex] 03	1	Speed	Main panel - Speed
[hex] 03	7	Altimeter	Main panel - Altimeter
[hex] 0A	1	Thrust	Engine - Thrust
[hex] 0E	*		Only RTA

For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- BUS<m>:MILStd:WORD<n>:SYMBol? on page 1584

12.9.5 MIL-STD-1553 Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

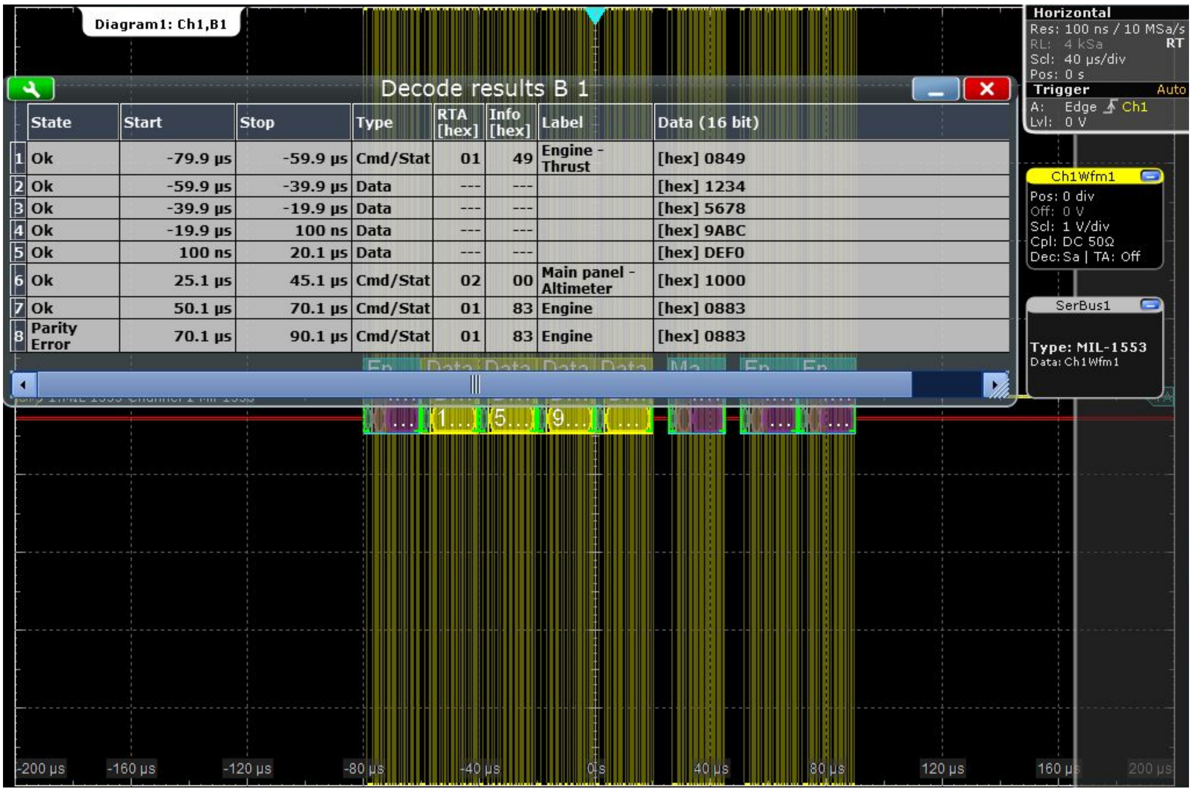


Figure 12-49: Decoded MIL-1553 signal with applied label list and results table. The last frame contains an error.

Table 12-11: Content of the decode result table

Column	Description
State	Overall state of the word
Start	Time of word start in relation to the trigger point
Stop	Time of word stop in relation to the trigger point
Type	Word type
RTA	Remote terminal address
Info	The hexadecimal value of the 9th to 1th bit of a command/status word
Label	Symbolic label name defined in the label list
Data	The values of the data bytes

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Export of decode results

1. In the protocol decode table, press "Export".
   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.10.3, "Decode Results", on page 1581.

12.9.6 Search on Decoded MIL Data

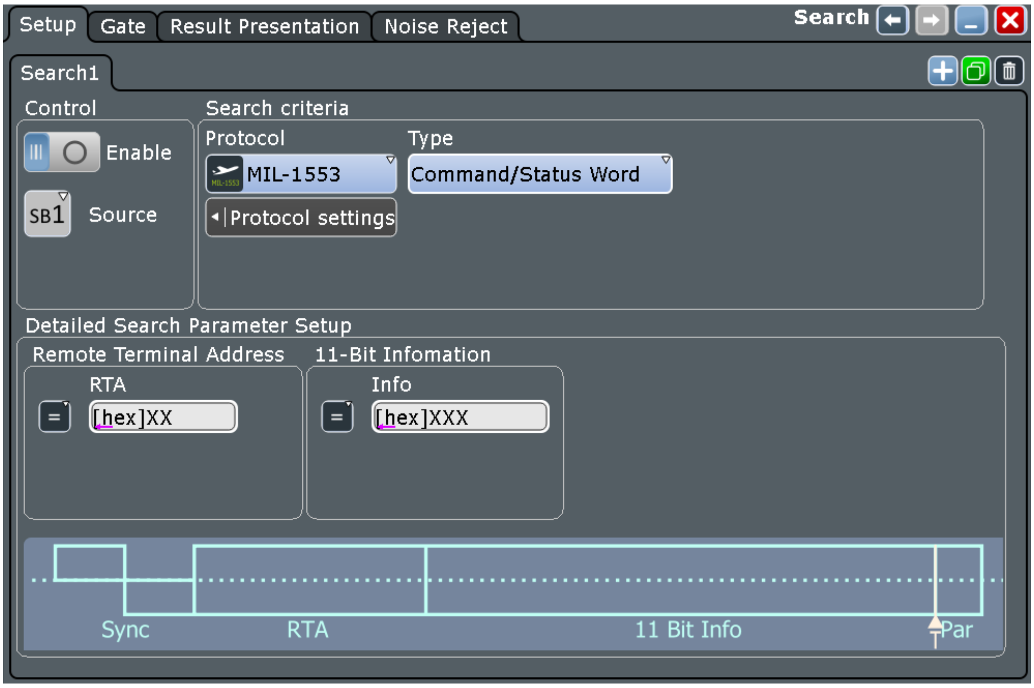
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.9.6.1 MIL Search Setup

Access: [SEARCH] > "Setup" tab



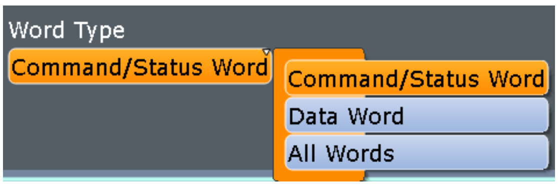
Type
The search criterion is defined by "Type". All trigger types are also available for search.

Additional search parameters are provided under "Detailed Search Parameter Setup".

"Sync"	Searches for a sync impulse.
"Word"	Searches for the selected word type.
"Data Word"	Searches for the specified data word. Additional search parameters: remote terminal address, data pattern and data index.
"Command / Status Word"	Searches for command or status words. Additional search parameters: remote terminal address, and 11-bit information.
"Command Word"	Searches for a command word. Additional search parameters: remote terminal address, subaddress / mode, and data word count / mode code.
"Status Word"	Searches for a status word. Additional search parameters: remote terminal address, and status flags.
"Error condition"	Identifies various errors in the frame, see "Error Condition" on page 630.

Remote command:
`SEARch:TRIGger:MILStd:TYPE` on page 1586

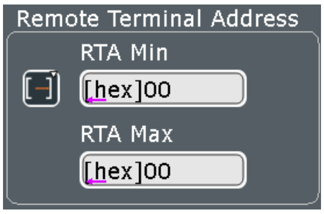
Sync Type / Word Type
Searches for a sync impulse/ word type. You can search for "Command/Status", "All" or "Data" sync pulses/ word types.



Remote command:
`SEARch:TRIGger:MILStd:TPSPecifier` on page 1591

Remote terminal address setup: Condition, RTA
The remote terminal address setup consists of the condition and one or two RTA patterns.

The RTA setup settings are the same as in the MIL trigger setup, see "Remote Terminal Address" on page 627.

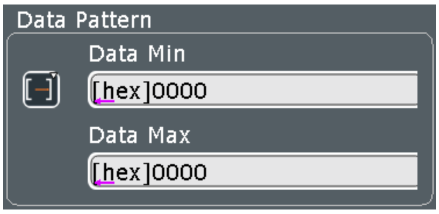


Remote command:
SEARch:TRIGger:MILStd:CDST:RCONdition on page 1586
SEARch:TRIGger:MILStd:CMD:RCONdition on page 1586
SEARch:TRIGger:MILStd:DATA:RCONdition on page 1586
SEARch:TRIGger:MILStd:CDST:RMIN on page 1587
SEARch:TRIGger:MILStd:CMD:RMIN on page 1587
SEARch:TRIGger:MILStd:DATA:RMIN on page 1587
SEARch:TRIGger:MILStd:CDST:RMAX on page 1587
SEARch:TRIGger:MILStd:CMD:RMAX on page 1587
SEARch:TRIGger:MILStd:DATA:RMAX on page 1587

Data pattern setup: Condition, Data min, Data max

The data pattern setup consists of the condition and one or two data patterns.

The data pattern setup settings are the same as in the MIL trigger setup, see "Data Pattern" on page 627.

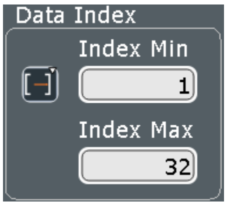


Remote command:
SEARch:TRIGger:MILStd:DATA:DCONdition on page 1588
SEARch:TRIGger:MILStd:DATA:DMIN on page 1588
SEARch:TRIGger:MILStd:DATA:DMAX on page 1589

Data index setup: Condition, Index min, Index max

The data index setup consists of the condition and one or two index patterns.

The data index setup settings are the same as in the MIL trigger setup, see "Data Index" on page 628.

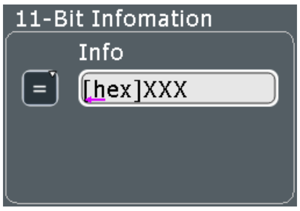


Remote command:
SEARch:TRIGger:MILStd:DATA:ICONdition on page 1589
SEARch:TRIGger:MILStd:DATA:IMIN on page 1589
SEARch:TRIGger:MILStd:DATA:IMAX on page 1589

11-Bit information setup: Condition, Info min, Info max

The 11-bit information setup consists of the condition and one or two 11-bit information patterns.

The 11-bit information setup settings are the same as in the MIL trigger setup, see "11-Bit Information" on page 628.

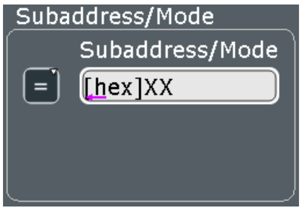


Remote command:
SEARch:TRIGger:MILStd:CDST:ICONdition on page 1587
SEARch:TRIGger:MILStd:CDST:IMIN on page 1588
SEARch:TRIGger:MILStd:CDST:IMAX on page 1588

Subaddress / Mode setup: Condition, Subaddress min, Subaddress max

The subaddress/mode setup consists of the condition and one or two subaddress/ mode patterns.

The subaddress/mode setup settings are the same as in the MIL trigger setup, see "Subaddress/ Mode" on page 629.

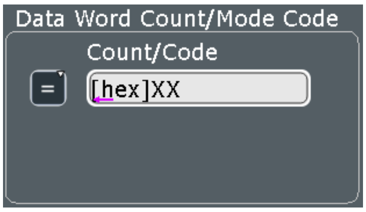


Remote command:
SEARch:TRIGger:MILStd:CMD:SCONdition on page 1588
SEARch:TRIGger:MILStd:CMD:SMIN on page 1588
SEARch:TRIGger:MILStd:CMD:SMAX on page 1589

Data word count / Mode code setup: Condition, Word count min, Word count max

The data word count/mode code setup consists of the condition and one or two patterns.

The subaddress/mode setup settings are the same as in the MIL trigger setup, see "Data Word Count/Mode Code" on page 629.



Remote command:
SEARch:TRIGger:MILStd:CMD:CCONdition on page 1587
SEARch:TRIGger:MILStd:CMD:CMIN on page 1588
SEARch:TRIGger:MILStd:CMD:CMAX on page 1588

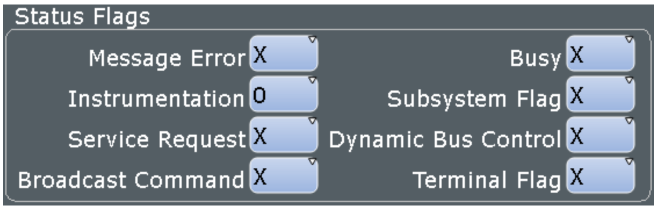
T/R (Transmit/receive)
Specifies the data direction of the selected command.

For details, see "T/R (Transmit/receive)" on page 630.

Remote command:
SEARch:TRIGger:MILStd:CMD:TR on page 1590

Status flags setup
Specifies the values (X, 0, 1) of the status flags.

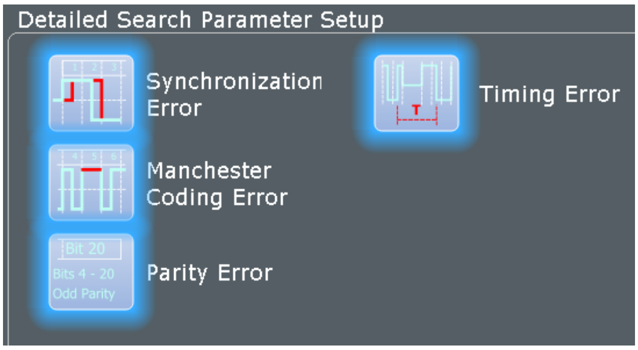
The status flags setup settings are the same as in the MIL trigger setup, see "Status Flags" on page 630.



Remote command:
SEARch:TRIGger:MILStd:STATus:BCReceived on page 1590
SEARch:TRIGger:MILStd:STATus:BUSY on page 1590
SEARch:TRIGger:MILStd:STATus:DBCaccept on page 1591
SEARch:TRIGger:MILStd:STATus:INSTrument on page 1591
SEARch:TRIGger:MILStd:STATus:MERRor on page 1591
SEARch:TRIGger:MILStd:STATus:SREQuest on page 1591
SEARch:TRIGger:MILStd:STATus:SUBSystem on page 1591
SEARch:TRIGger:MILStd:STATus:TERMinal on page 1591

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the MIL trigger setup, see "Error Condition" on page 630



Remote command:
SEARch:TRIGger:MILStd:ERRor:MANChester on page 1590
SEARch:TRIGger:MILStd:ERRor:PARity on page 1590
SEARch:TRIGger:MILStd:ERRor:SYNC on page 1590
SEARch:TRIGger:MILStd:ERRor:TIMing on page 1590

12.9.6.2 MIL Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

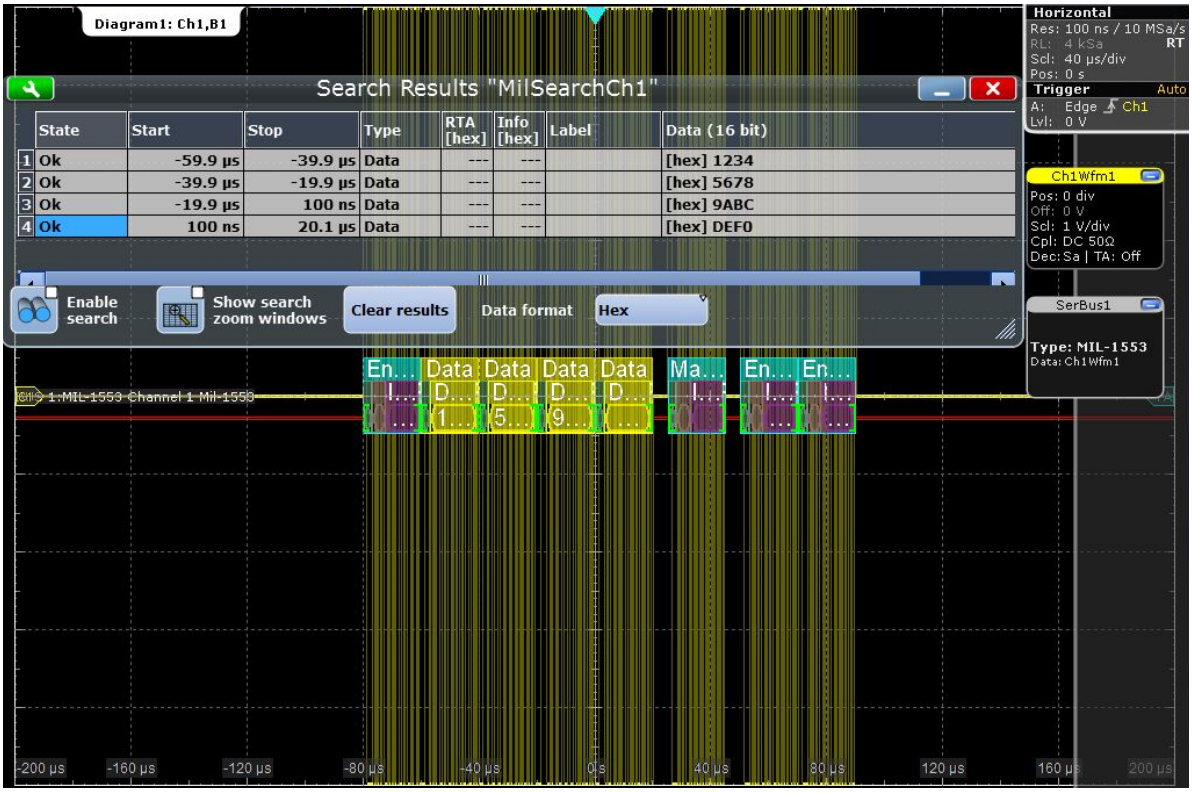


Figure 12-50: Results of a search for all data words (any RTA and any data pattern)

Remote commands:

- `SEARch:RESult:MILStd:WCOunt?` on page 1592
- `SEARch:RESult:MILStd:WORD<m>:INFO?` on page 1594
- `SEARch:RESult:MILStd:WORD<m>:RTADdress?` on page 1593
- `SEARch:RESult:MILStd:WORD<m>:STARt?` on page 1592
- `SEARch:RESult:MILStd:WORD<m>:STATus?` on page 1592
- `SEARch:RESult:MILStd:WORD<m>:STOP?` on page 1593
- `SEARch:RESult:MILStd:WORD<m>:SYMBol?` on page 1593
- `SEARch:RESult:MILStd:WORD<m>:TYPE?` on page 1592

12.10 ARINC 429 (Option R&S RTE-K7)

12.10.1 ARINC 429 Basics

The ARINC 429 is a specification that defines the characteristics of an avionic data bus used on commercial and transport aircraft.

In an ARINC 429 system, a single transmitter/source is connected to 1 to 20 receivers/ sinks on one twisted wire pair. The bus uses differential signals. The ARINC 429 standard uses a simplex communication - data may be transmitted in only one direction. The information is transmitted over the bus in defined series of words.

Word Format

MSB ... LSB

32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
P	SSM		Data																			SDI		Label							

Figure 12-51: Structure of an ARINC 429 word

An ARINC 429 word is 32-bits and consists of the following parts (see Figure 12-51):

- Parity: the most significant bit (MSB). Checks if there are bit errors during the transmission. The total number of logic 1 bits for the word shall be odd.
- Sign/Status Matrix (SSM): the value of these bits depends on the data type. It may be used to report the status of hardware equipment.
- Data:
  - Binary (BNR): stores the data as a binary number.
  - Binary Coded Decimal (BCD): uses 4 data field bits to represent a decimal digit.
  - Discrete data: a combination of BNR and/ or BCD or individual bits that express specific equipment conditions.
  - Maintenance data and acknowledgment
  - Williamsburg / Buckhorn protocol: a bit-oriented protocol that is used for file transfer.
- Source/Destination Identifier (SDI): indicates the intended receiver or the transmitting subsystem.
- Label: gives information about the word's data type.

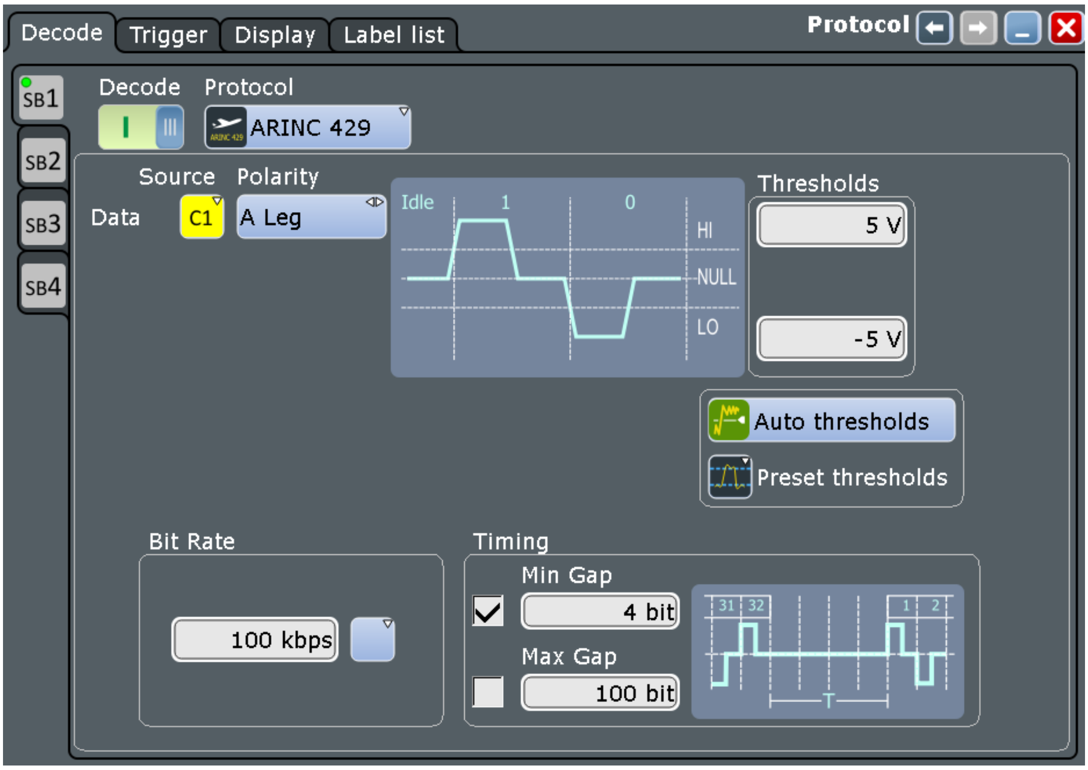
For comfortable analysis, you can load an editable label list, to interpret transferred numeric values as meaningful text labels.

12.10.2 ARINC 429 Configuration

12.10.2.1 ARINC 429 Configuration Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol" = *ARINC 429*

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Data

Sets the source of the selected data line. Usually, the source is one of the analog channels. Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

For triggering on a serial bus, a channel signal is required.

Remote command:
`BUS<m>:ARINc:SOURce` on page 1595

Polarity

Selects the wire on which the bus signal is measured : "A Leg" or "B Leg". The setting affects the digitization of the signal.

Remote command:
`BUS<m>:ARINc:POLarity` on page 1596

Thresholds

Sets the threshold value for digitization of the data signal. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the threshold.

There are three ways to set the thresholds:

- "High" and "Low"
  Upper and lower threshold levels. You can enter the values directly in the fields.
- "Preset thresholds"

Selects the default threshold voltage from a list. The value is set to "Manual" if the threshold was set with "Auto thresholds", or was entered directly.

- "Auto thresholds"
  Sets the thresholds to the middle reference level of the measured amplitudes.

Remote command:
`BUS<m>:ARINc:THReshold:HIGH` on page 1597
`BUS<m>:ARINc:THReshold:LOW` on page 1597
`BUS<m>:ARINc:PRESet` on page 1597
`BUS<m>:SETReflevels` on page 1383

Bit Rate
Selects the number of transmitted bits per second. The value can be set to high speed (100 kbps) or low speed (12.0- 14.5 kbps).

Remote command:
`BUS<m>:ARINc:BRValue` on page 1595
`BUS<m>:ARINc:BRMode` on page 1595

Timing: Min gap, Max gap
Defines the idle time between two words, which is needed for word synchronization. The beginning of the first bit after the gap marks the start of a new word.

You can define a minimum idle time "Min gap", and/or a maximum time "Max gap". The standard defines a minimum of 4-bit times to separate two subsequent words.

Timing settings are relevant for protocol configuration and error trigger.

If "Min gap" and/or "Max gap" are enabled, the instrument detects the specified gaps during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shorter or longer than the specified gaps, respectively.

Remote command:
`BUS<m>:ARINc:MAXGap:BITS` on page 1596
`BUS<m>:ARINc:MAXGap:SELect` on page 1595
`BUS<m>:ARINc:MINGap:BITS` on page 1596
`BUS<m>:ARINc:MINGap:SELect` on page 1596
`TRIGger<m>:ARINc:MINGap:BITS` on page 1600
`TRIGger<m>:ARINc:MINGap:SELect` on page 1600
`TRIGger<m>:ARINc:MAXGap:BITS` on page 1600
`TRIGger<m>:ARINc:MAXGap:SELect` on page 1600

12.10.2.2 Configuring ARINC 429 Signals

For configuration assign the line to the input channel, set the threshold and the timing conditions.

For details on configuration settings, see Chapter 12.10.2.1, "ARINC 429 Configuration Settings", on page 642.

To display the decoded signal, option R&S RTE-K7 is required.

1. Press the [PROTOCOL] key on the front panel.

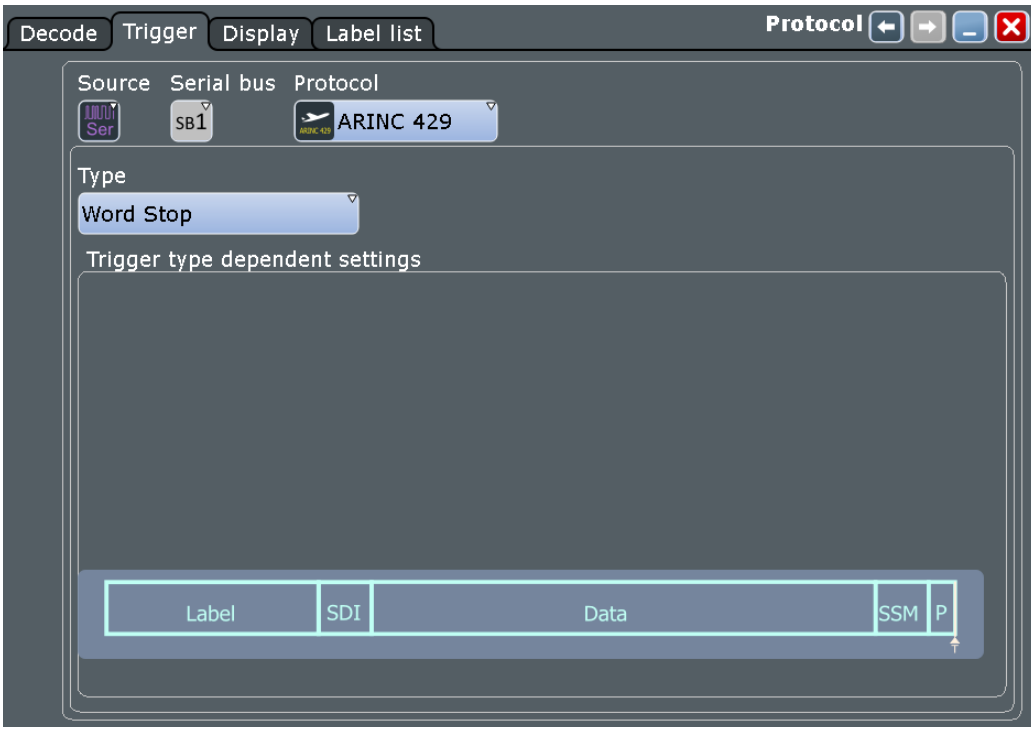
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "ARINC 429".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Tap the "Polarity" button, and select the waveform of the data line.
7. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to the middle reference levels with "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.
8. Tap the "Bit Rate" button and set it for high or low speed.
9. If necessary, tap the "Min Gap" button to select it and set the minimum gap time.
10. If necessary, tap the "Max Response" button to select it and set the maximum response time.

12.10.3 ARINC 429 Trigger

12.10.3.1 ARINC 429 Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = ARINC 429"

Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.



Trigger Type

Selects the trigger type for ARINC 429 analysis.

"Word Start" Sets the trigger to the start of the word.

"Word Stop" Sets the trigger to the stop of the word.

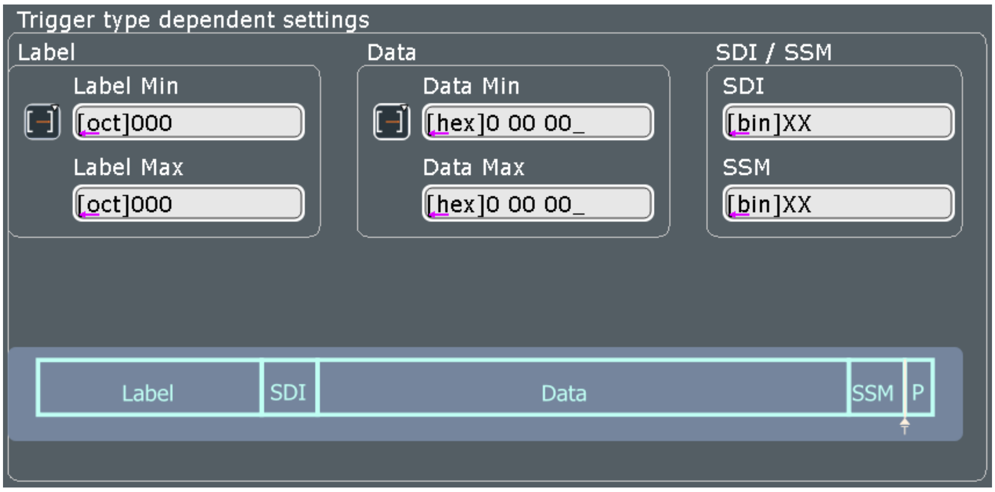
"Label + Data" Sets the trigger on a defined word format. You can define the label, the data and the SDI / SSM bits separately, see "Label + Data" on page 646.

"Error Condition"
Identifies various errors in the word, see "Error Conditions" on page 648.

Remote command:
`TRIGger<m>:ARINc:TYPE` on page 1598

Label + Data

Sets the trigger on a defined word format. You can define the label, the data and the SDI / SSM bits separately.



Label setup: Condition, Label Min, Label Max ← Label + Data

The label setup consists of the condition and one or two label patterns.

"Condition" Defines the operator to set a specific label ("Equal" or "Not equal") or a label range.

"Label Min" Defines the bit pattern of the label.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Label Max" The second label pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ARINc:LABel:CONDition` on page 1599
`TRIGger<m>:ARINc:LABel:MIN` on page 1599
`TRIGger<m>:ARINc:LABel:MAX` on page 1600

Data setup: Condition, Data Min, Data Max ← Label + Data

The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data ("Equal" or "Not equal") or a data range.

"Data Min" Defines the bit pattern of the data.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

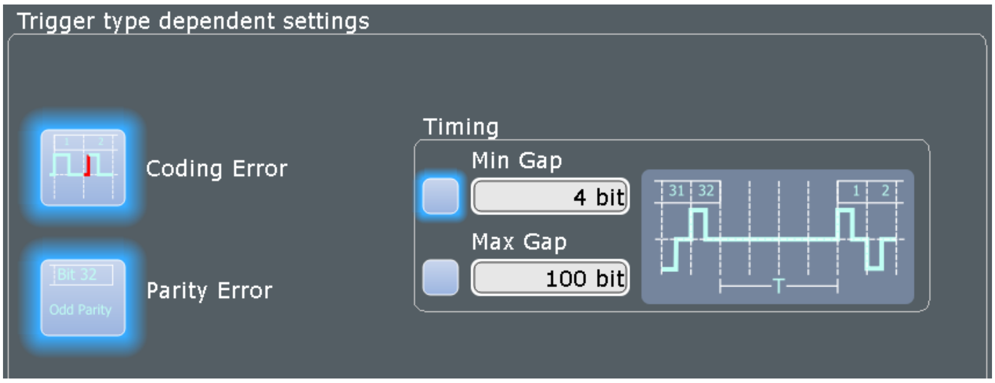
"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ARINc:DATA:CONDition` on page 1598
`TRIGger<m>:ARINc:DATA:MIN` on page 1598
`TRIGger<m>:ARINc:DATA:MAX` on page 1599

SDI / SSM ← Label + Data
Sets the values for the source/destination identifier (SDI) and the sign/status matrix (SSM) bits.

Remote command:
`TRIGger<m>:ARINc:SDI` on page 1601
`TRIGger<m>:ARINc:SSM` on page 1601

Error Conditions
Specifies the error conditions to be triggered on.



Coding error ← Error Conditions
Triggers on a coding error.

Remote command:
`TRIGger<m>:ARINc:ERRor:CODing` on page 1599

Parity Error ← Error Conditions
Checks the parity and triggers if the parity is even.

Remote command:
`TRIGger<m>:ARINc:ERRor:PARity` on page 1599

Timing: Min gap, Max gap ← Error Conditions
Defines the idle time between two words, which is needed for word synchronization. The beginning of the first bit after the gap marks the start of a new word.

You can define a minimum idle time "Min gap", and/or a maximum time "Max gap". The standard defines a minimum of 4-bit times to separate two subsequent words.

Timing settings are relevant for protocol configuration and error trigger.

If "Min gap" and/or "Max gap" are enabled, the instrument detects the specified gaps during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shorter or longer than the specified gaps, respectively.

Remote command:
`BUS<m>:ARINc:MAXGap:BITS` on page 1596
`BUS<m>:ARINc:MAXGap:SELect` on page 1595
`BUS<m>:ARINc:MINGap:BITS` on page 1596
`BUS<m>:ARINc:MINGap:SELect` on page 1596
`TRIGger<m>:ARINc:MINGap:BITS` on page 1600

TRIGger<m>:ARINc:MINGap:SELect on page 1600
TRIGger<m>:ARINc:MAXGap:BITS on page 1600
TRIGger<m>:ARINc:MAXGap:SELect on page 1600

12.10.3.2 Triggering on ARINC 429

Prerequisites: A bus is configured for the ARINC 429 signal to be analyzed.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to ARINC 429.
5. Select the "Trigger type".
6. For more complex trigger types, enter the data pattern conditions.
   For details, see Chapter 12.10.3.1, "ARINC 429 Trigger Settings", on page 645.

12.10.4 ARINC 429 Label List

Label lists are protocol-specific. An ARINC 429 label file contains two values for each identifier:

- "Arinc Label": the ARINC 429 label value, that identifies the data type and the parameters associated with it. The usual data format is octal.
- "Symbolic label": symbolic name of the label, specifying its function.

Example: ARINC 429 PTT file

```
# ----------------------------------------------------------------------------
@FILE_VERSION = 1.0
@PROTOCOL_NAME = arinc429
# ----------------------------------------------------------------------------
# Labels for ARINC 429 protocol
#   Column order: Arinc Label, Symbolic Label
# ----------------------------------------------------------------------------
# ----Definition----
001o, Distance to Go
002o, Time to Go
010o, Present Position - Latitude
011o, Present Position - Longitude
014o, Magnetic Heading
015o, Wind Speed
075o, Gross Weight
125o, Universal Time Coordinated
# ----------------------------------------------------------------------------
```

ARINC 429 Label List

Arinc Label [oct]	Symbolic Label
[oct] 001	Distance to Go
[oct] 002	Time to Go
[oct] 010	Present Position - Latitude
[oct] 011	Present Position - Longitude
[oct] 014	Magnetic Heading
[oct] 015	Wind Speed
[oct] 075	Gross Weight
[oct] 125	Universal Time Coordinated

For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- BUS<m>:ARINc:WORD<n>:SYMBol? on page 1604

12.10.5 ARINC 429 Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

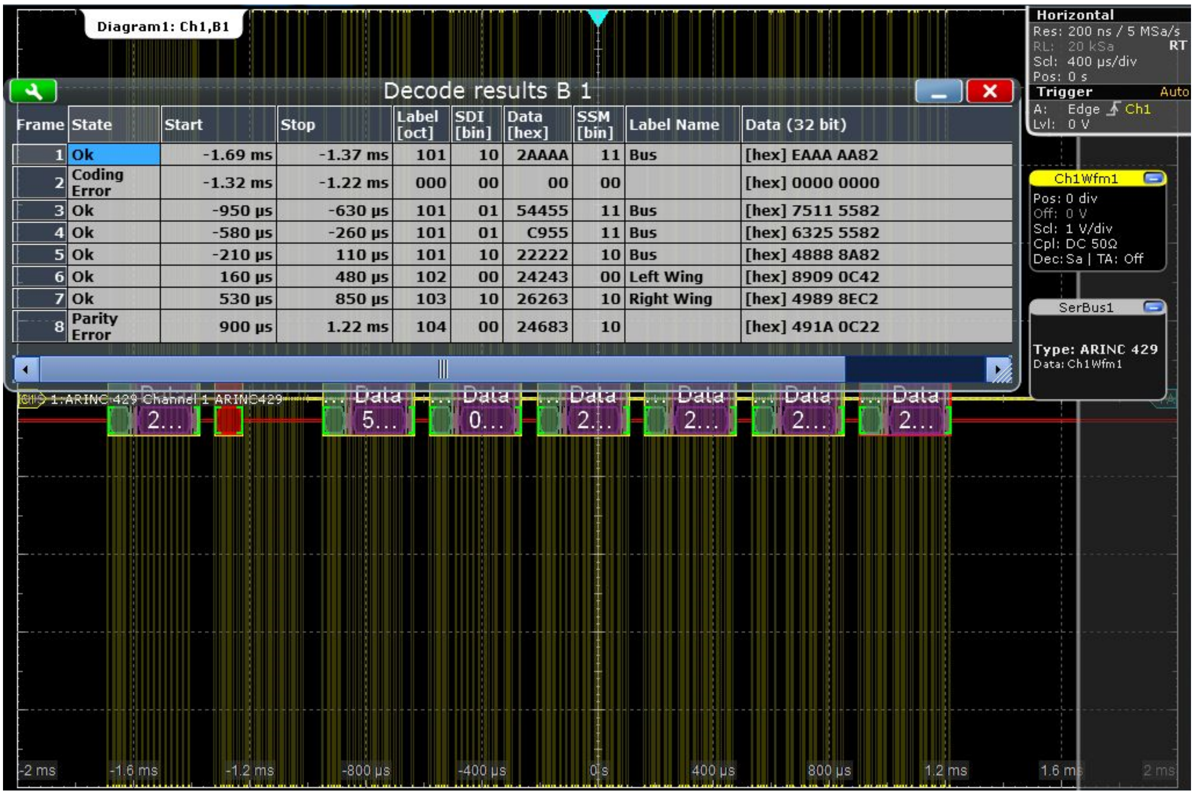


Figure 12-52: Decoded ARINC 429 signal with applied label list and results table. The second and eight frames contain errors.

Table 12-12: Content of the "Decode results" table

Column	Description
State	Overall state of the frame.
Start	Time of word start in relation to the trigger point
Stop	Time of word stop in relation to the trigger point
Label	The value of the label bytes
SDI	The state of the SDI bits
DATA	All 32 bits of the word.
SSM	The state of the SSM bits
Label Name	The label name
Data	The value of the data bytes

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.11.3, "Decode Results", on page 1601.

12.10.6 Search on Decoded ARINC 429 Data

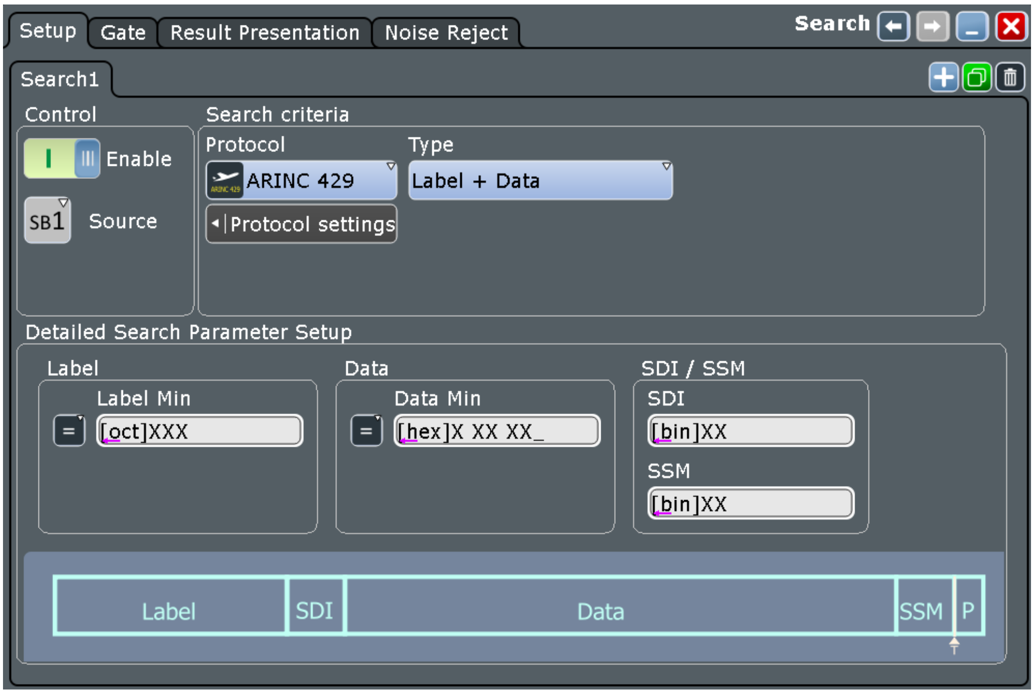
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.10.6.1 ARINC 429 Search Setup

Access: [SEARCH] > "Setup" tab



Type

The search criterion is defined by "Type". All trigger types are also available for search.

Additional search parameters are provided under "Detailed Search Parameter Setup".

"Word Start" Searches for the start word.

"Word Stop" Searches for the stop word.

"Label + Data" Searches for a defined word format. You can search for the label, the data, the SDI, and SSM bits separately.
For details, see "Label + Data" on page 646.

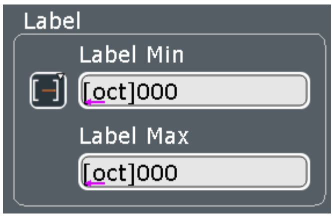
"Error condition"
Identifies various errors in the frame, see "Error Conditions" on page 648.

Remote command:
`SEARch:TRIGger:ARINc:TYPE` on page 1605

Label setup: Condition, Label min, Label max

The label setup consists of the condition and one or two label patterns.

The label setup settings are the same as in the ARINC trigger setup, see "Label setup: Condition, Label Min, Label Max" on page 647.

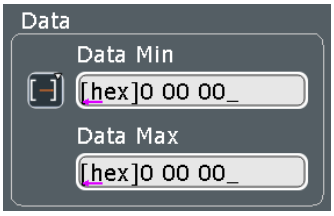


Remote command:
`SEARch:TRIGger:ARINc:LABel:CONDition` on page 1605
`SEARch:TRIGger:ARINc:LABel:MIN` on page 1605
`SEARch:TRIGger:ARINc:LABel:MAX` on page 1606

Data setup: Condition, Data min, Data max

The data setup consists of the condition and one or two data patterns.

The data setup settings are the same as in the ARINC trigger setup, see "Data setup: Condition, Data Min, Data Max" on page 647.

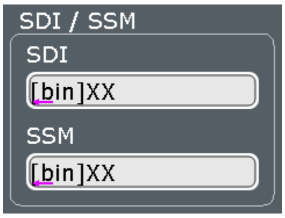


Remote command:
SEARch:TRIGger:ARINc:DATA:CONDition on page 1605
SEARch:TRIGger:ARINc:DATA:MIN on page 1605
SEARch:TRIGger:ARINc:DATA:MAX on page 1606

SDI / SSM setup: SDI, SSM

The SDI / SSM setup consists of the SDI and SSM.

The SDI / SSM setup settings are the same as in the ARINC trigger setup, see "SDI / SSM" on page 648.

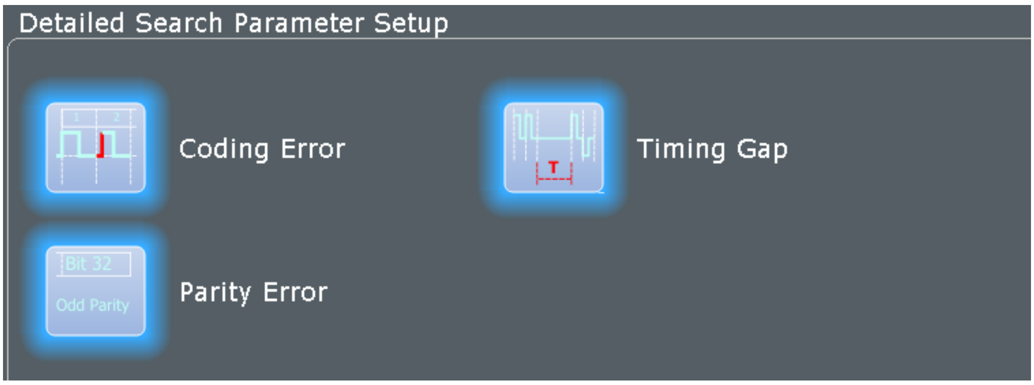


Remote command:
SEARch:TRIGger:ARINc:SDI on page 1606
SEARch:TRIGger:ARINc:SSM on page 1606

Error Condition

Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the ARINC trigger setup, see "Error Conditions" on page 648



Remote command:
SEARch:TRIGger:ARINc:ERRor:CODing on page 1606
SEARch:TRIGger:ARINc:ERRor:PARity on page 1607
SEARch:TRIGger:ARINc:ERRor:TIMing on page 1607

12.10.6.2 ARINC Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

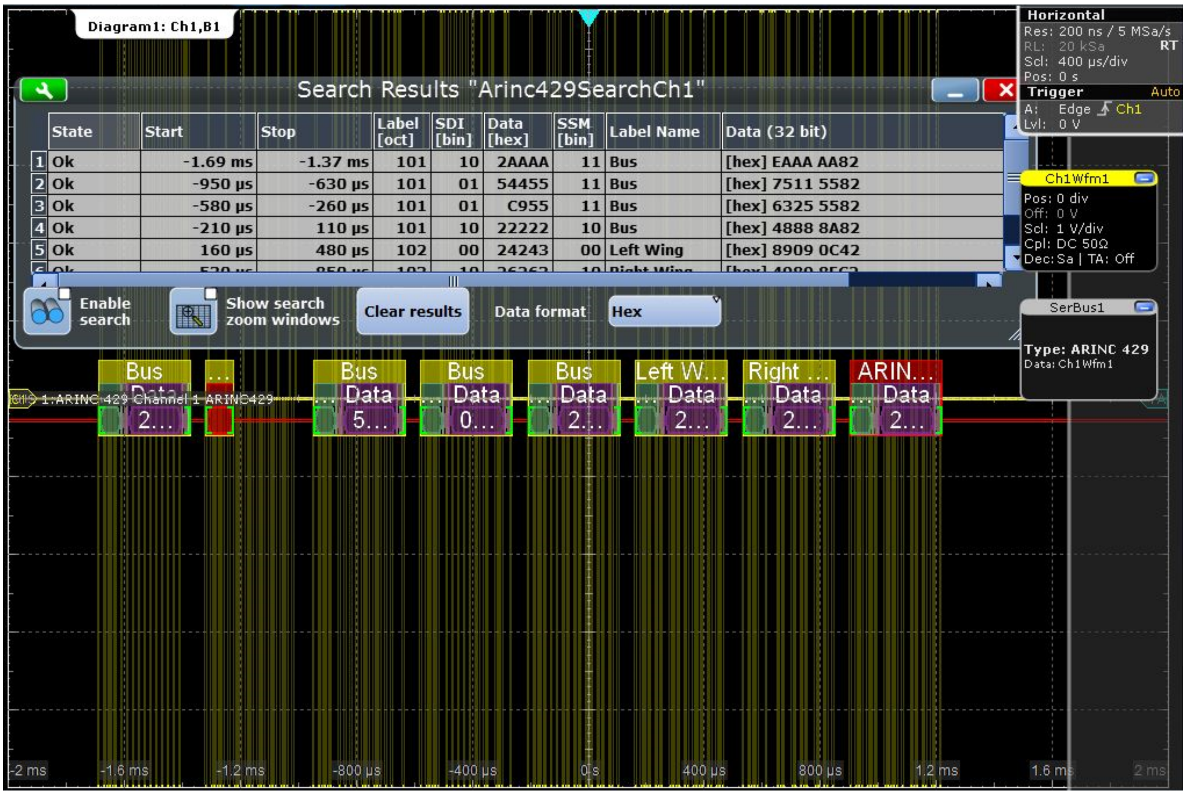


Figure 12-53: Results of a general "Label +data" search with applied label list. All frames are found that contain any label, any data, and any SDI/SSM bits.

Remote commands:

- `SEARch:RESult:ARINc:WCOunt?` on page 1609
- `SEARch:RESult:ARINc:WORD<m>:DATA?` on page 1608
- `SEARch:RESult:ARINc:WORD<m>:LABel?` on page 1608
- `SEARch:RESult:ARINc:WORD<m>:PATTern?` on page 1608
- `SEARch:RESult:ARINc:WORD<m>:SDI?` on page 1609
- `SEARch:RESult:ARINc:WORD<m>:SSM?` on page 1608
- `SEARch:RESult:ARINc:WORD<m>:STARt?` on page 1610
- `SEARch:RESult:ARINc:WORD<m>:STATe?` on page 1610

- `SEARch:RESult:ARINc:WORD<m>:STOP?` on page 1609
- `SEARch:RESult:ARINc:WORD<m>:SYMBol?` on page 1609

12.11 Ethernet 10BASE-T and 100BASE-TX (Option R&S RTE-K8)

Twisted-pair Ethernet technologies are based on the family of standards IEEE 802.3, issued by the Institute of Electrical and Electronics Engineers (IEEE).

R&S RTE-K8 is a firmware option that enables the R&S RTE to analyze Ethernet protocol variants 10BASE-T and 100BASE-TX, by decoding the signal and searching within the decoded events. It is possible to trigger on 10BASE-T signals and 100BASE-TX signals. The option is compatible with the standards IEEE 802.3i of 1990 (10BASE-T) and IEEE 802.3u of 1995 (100BASE-TX).

- The Ethernet Protocol....656
- Ethernet Configuration....657
- Ethernet Trigger....661
- Ethernet Label List....667
- Ethernet Decode Results....668
- Search on Decoded Ethernet Data....671

12.11.1 The Ethernet Protocol

The two Ethernet protocol variants that R&S RTE-K8 can process have the following features:

- 10BASE-T uses Manchester coding (or phase encoding, PE). In terms of a logical Boolean operation, the Manchester value of each bit is the exclusive disjunction (XOR) of the original data value and the clock value. A "0" is expressed by a high-to-low transition, a "1" by a low-to-high transition. These transitions, which occur at the middle of each bit period, make the signal self-clocked.
- 100BASE-TX uses a 4B5B Multi-Level Transmit (MLT-3) encoding. The protocol sequentially cycles through a sequence of the voltage levels -1 V, 0 V, +1 V, and 0 V. To transmit a "1" bit, MLT-3 moves to the next state; to transmit a "0" bit, it stays in the same state. 4B5B block coding is used to map groups of four bits onto groups of five bits. Also, the signal is scrambled.

All Ethernet-over-twisted-pair technologies use wires with four twisted pairs of cables (and 8P8C connectors), but 10BASE-T and 100BASE-TX only require two pairs of wires.

12.11.2 Ethernet Configuration

12.11.2.1 Ethernet Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *Ethernet*

Make sure that the tab of the correct serial bus is selected on the left side.

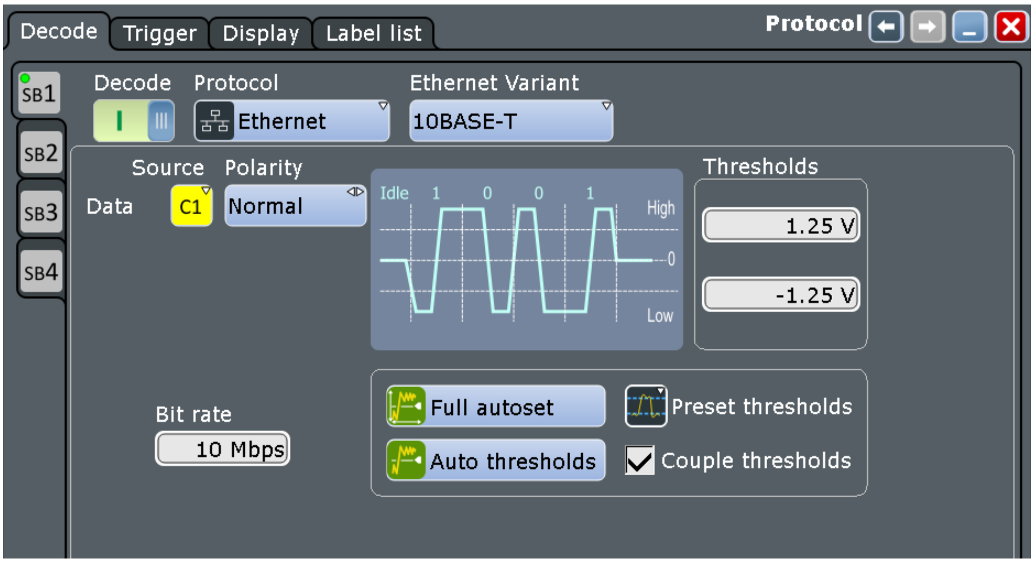


Figure 12-54: Serial bus protocol configuration dialog

For general information on how to configure protocol parameters, see also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1382

Decode
Enables the decoding of the selected bus. The signal icon of the bus appears on the signal bar.

Remote command:
`BUS<m>[:STATe]` on page 1382

Ethernet Variant
Defines the Ethernet protocol variant and transmission speed.

"10BASE-T"

Selects the Ethernet protocol variant 10BASE-T (standard data rate 10 Mbit/s).

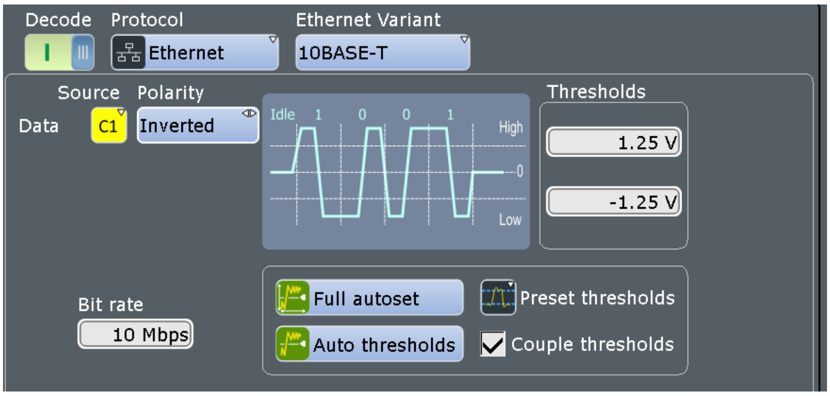


Figure 12-55: Ethernet 10BASE-T protocol configuration (here with inverted polarity)

"100BASE-TX"

Selects the Ethernet protocol variant 100BASE-TX, which provides 100 Mbit/s use data rate. Due to 4b/5b encoding, the raw data rate on the line is 125 Mbit/s. This value is used by R&S RTE-K8 as the bit rate default for 100BASE-TX.

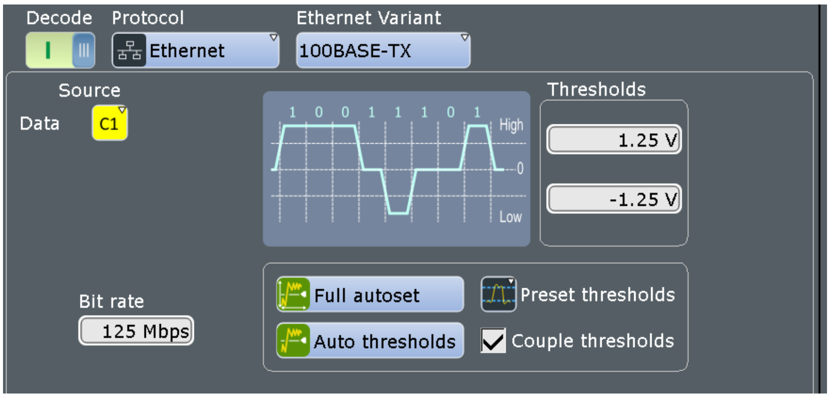


Figure 12-56: Ethernet 100BASE-TX protocol configuration

Remote command:
BUS<m>:ETHernet:VARiant on page 1611

Source

Defines the source settings for the data signal.

Permitted source selections are the analog, mathematical, and reference channels.

As soon as the serial bus trigger has been selected, the only permitted source selections are the analog channels "C1" – "C4", which are required for triggering.

Remote command:
BUS<m>:ETHernet:SOURce on page 1611

Polarity

Defines the polarity ("Normal" or "Inverted") of the data signal. This setting is only available in 10BASE-T.

Remote command:
`BUS<m>:ETHernet:POLarity` on page 1611

Thresholds

Sets the threshold value for the digitization of each signal line. If the signal value on the line is higher than the upper threshold, the signal state is high. Otherwise, if the signal value is below the lower threshold, the signal state is considered low.

There are four ways to set the threshold:

- "Thresholds"
  Enter the values directly: upper threshold in the upper field, lower threshold in the lower field.
- "Full autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
  See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Preset thresholds"
  Either sets individual voltages by selecting "Manual", or sets the voltages to one out of four pre-defined levels:
  – 10BASE-T (0 meters): *±1.25 V*
  – 10BASE-T (100 meters): *±750 mV*
  – 100BASE-TX (0 meters): *±500 mV*
  – 100BASE-TX (100 meters): *±350 mV*

  The "Preset" levels depend on:
  – The Ethernet variant
  – The distance from the transmitter. "0 meters" represents "voltage at transmitter" and "100 meters" represents "voltage at the maximum cable length", according to the standard.

  When any non-predefined threshold is set, the "Preset" value automatically changes to manual (without affecting anything else).
- "Auto thresholds"
  Executes a measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Couple thresholds"
  Couples threshold settings between upper and lower threshold.

Remote command:
`BUS<m>:ETHernet:THReshold:HIGH` on page 1612
`BUS<m>:ETHernet:THReshold:LOW` on page 1612
`BUS<m>:ETHernet:PRESet` on page 1612
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Bit rate

Defines the transmission speed setting for the data signal:

- 10BASE-T: default bit rate 10 Mbps
- 100BASE-TX: default bit rate 125 Mbps

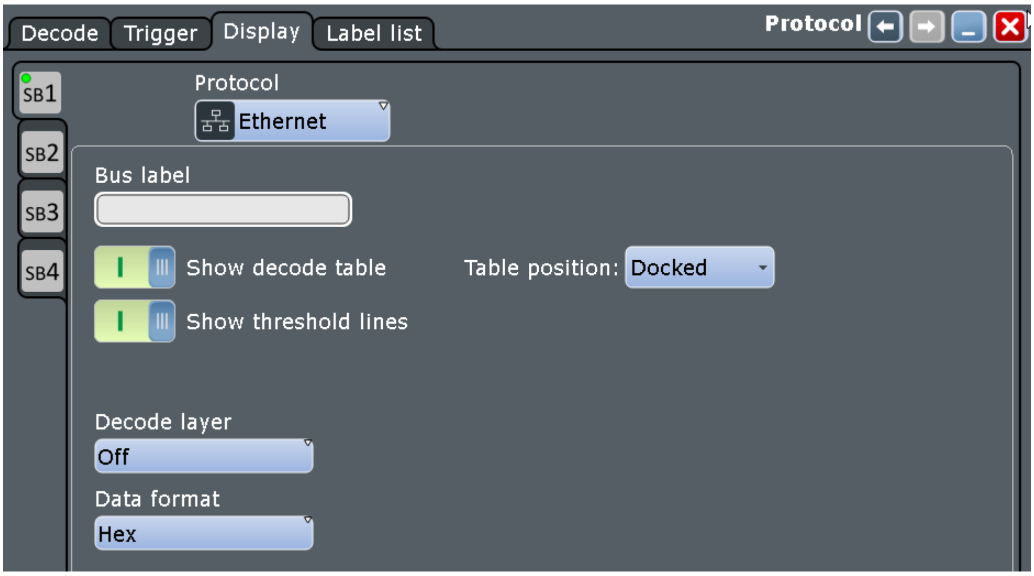
In both variants, the permitted bit rates range from 10 kbps to 150 Mbps. Switching the variant adjusts the bit rate, independent of the previous setting.

Remote command:
`BUS<m>:ETHernet:BITRate` on page 1613

12.11.2.2 Ethernet Display Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol = Ethernet" > "Display" tab

To enhance the decode possibilities of the Ethernet protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Common display settings are explained in Chapter 12.1.3, "Display", on page 475.

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

12.11.2.3 Configuring Ethernet Signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

1. Press the [PROTOCOL] key on the front panel.
2. At the left hand-side, select the vertical tab of the serial bus (SB1–SB4) you want to set up.
3. Select the "Configuration" tab.
4. Tap "Protocol" and select the protocol: "Ethernet".

5. Optionally, you can enter a "Bus label" in the "Display" tab.
6. Tap "Ethernet Variant" and select the variant ("10BASE-T" or "100BASE-TX") you want to set up.

   Note: Note that no triggering on the serial bus is available.
   To trigger the signal, use the edge trigger on the source channel.
7. For the variant "10BASE-T", select the polarity ("Normal" or "Inverted") of the data signal.
8. Set the logical thresholds, see "Thresholds" on page 659.
9. In the protocol "Configuration" tab, select "Decode" to activate the decode functionality.

For details on configuration settings, see Chapter 12.11.2.1, "Ethernet Configuration Settings", on page 657.

12.11.3 Ethernet Trigger

If you need information on how to get started with triggering on Ethernet 10BASE-T signals, see Chapter 12.11.3.2, "Triggering on Ethernet", on page 666. Otherwise proceed with the Ethernet trigger settings.

12.11.3.1 Ethernet Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = Ethernet"

Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

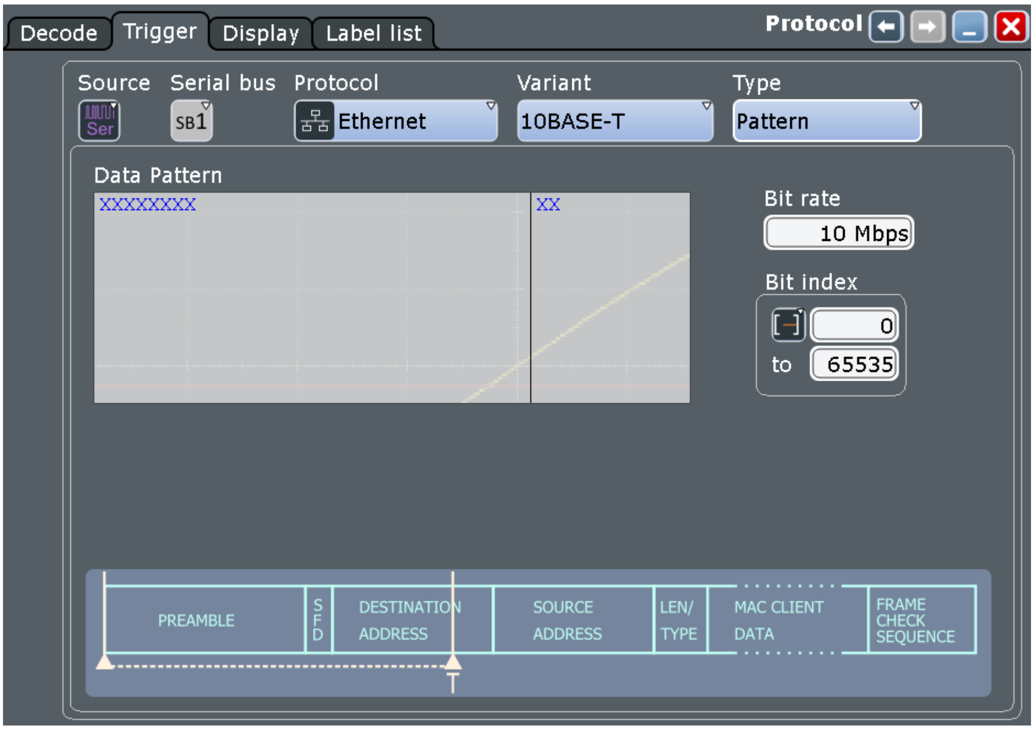


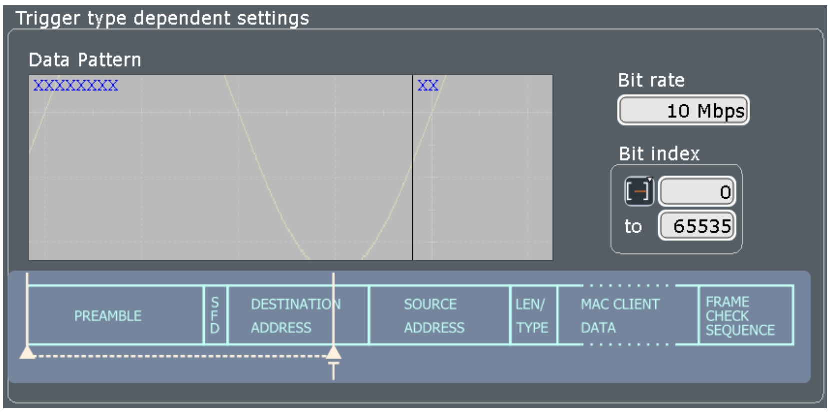
Figure 12-57: Ethernet trigger event settings dialog

Type

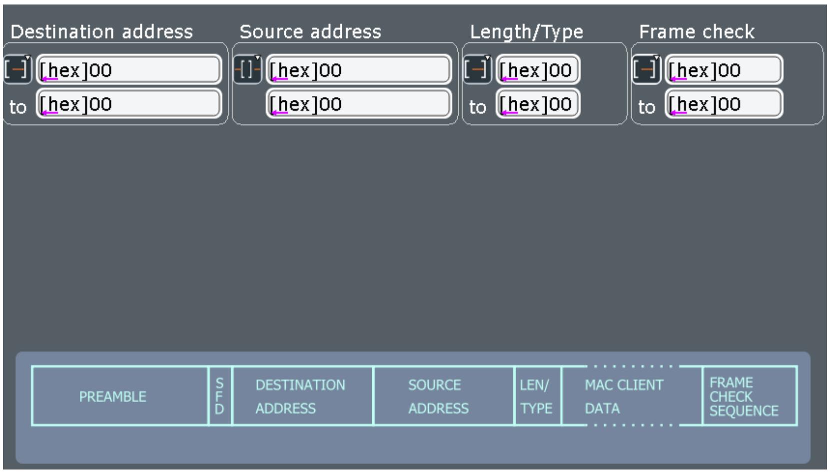
Selects the trigger type for the Ethernet analysis.

"Frame Start" Available only for "Variant > 10BASE-T".
Sets the trigger to the start frame delimiter (SFD). The start of frame (SOF) condition is the occurrence of the preamble; the trigger instant is after the SFD.

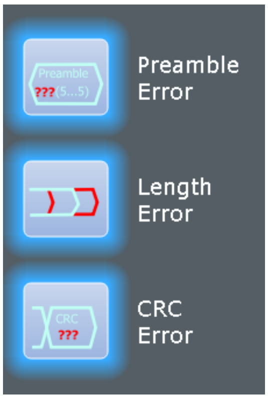
"Pattern" Available only for "Variant > 10BASE-T".
Sets the trigger to any bit pattern (data) that can be freely specified, starting from the beginning of the frame. The trigger instant is after the last bit of the specified data pattern.



"Adv frame " Triggers on an advanced frame.



"Adv error" Triggers on any selected errors.



Remote command:
TRIGger<m>:ETHernet:TYPE on page 1614

Ethernet Variant
Defines the Ethernet protocol variant and transmission speed.

"10BASE-T"

Selects the Ethernet protocol variant 10BASE-T (standard data rate 10 Mbit/s).

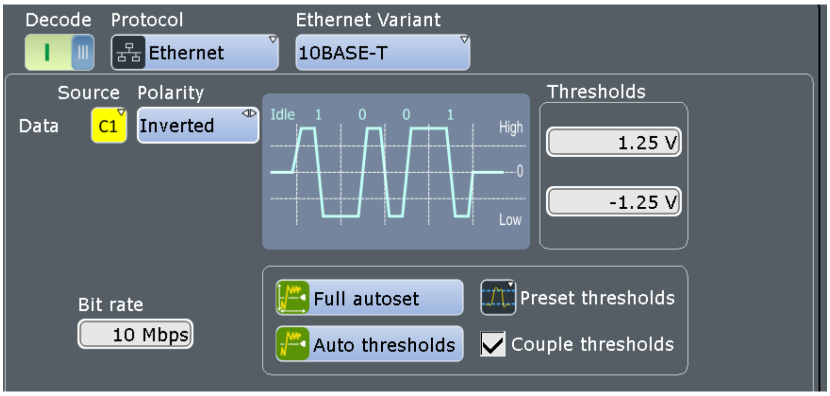


Figure 12-58: Ethernet 10BASE-T protocol configuration (here with inverted polarity)

"100BASE-TX"

Selects the Ethernet protocol variant 100BASE-TX, which provides 100 Mbit/s use data rate. Due to 4b/5b encoding, the raw data rate on the line is 125 Mbit/s. This value is used by R&S RTE-K8 as the bit rate default for 100BASE-TX.

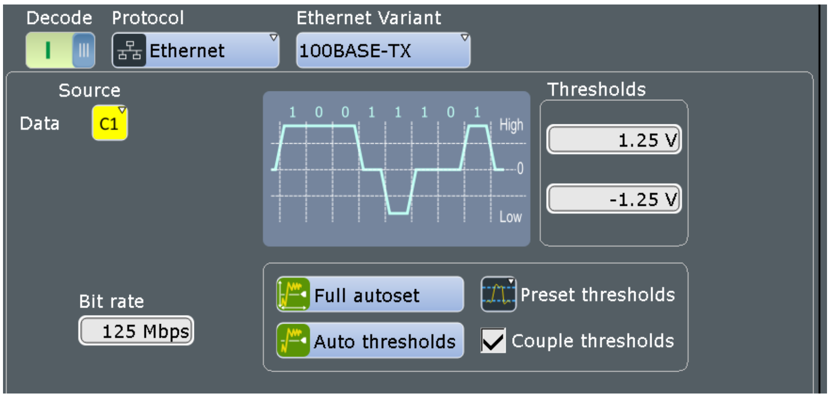


Figure 12-59: Ethernet 100BASE-TX protocol configuration

Remote command:
`BUS<m>:ETHernet:VARiant` on page 1611

Data Pattern

Specifies the data pattern that is to be triggered on.

Remote command:
`TRIGger<m>:ETHernet:PATTern` on page 1614

Bit rate

Defines the transmission speed setting for the data signal:

- 10BASE-T: default bit rate 10 Mbps
- 100BASE-TX: default bit rate 125 Mbps

In both variants, the permitted bit rates range from 10 kbps to 150 Mbps. Switching the variant adjusts the bit rate, independent of the previous setting.

Remote command:
`BUS<m>:ETHernet:BITRate` on page 1613

Bit index
Defines the position of the first bit of the data pattern.

"Bit index operator" Sets the operator ("Equal", "Greater or equal", or "In range").

"Bit index" Sets the bit index (data position), or the start value of a bit index range.

"Bit index to" Sets end value of a bit index range (data position range). Available only, if the "Bit index operator" is set to "In range".

Destination Address
Defines the destination address.

"Condition" Sets the operator to trigger on a specific destination address pattern ("Equal" or "Not equal") or an address range.

"Destinataion Address (Min)" Defines the destination address pattern for all operators that require one pattern.

"Destinataion Adress (Max)" Defines the second destination address pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:DCONdition` on page 1616
`TRIGger<m>:ETHernet:FRAMe:DMAX` on page 1616
`TRIGger<m>:ETHernet:FRAMe:DMIN` on page 1617

Source Address
Defines the source address, the physical address of the device that sends the frame.

"Condition" Sets the operator to trigger on a specific source address pattern ("Equal" or "Not equal") or an address range.

"Source Address (Min)" Defines the source address pattern for all operators that require one pattern.

"Source Adress (Max)" Defines the second source address pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:SCONdition` on page 1617
`TRIGger<m>:ETHernet:FRAMe:SMAX` on page 1617
`TRIGger<m>:ETHernet:FRAMe:SMIN` on page 1617

Length/ Type
Defines the length/type field value.

"Condition" Sets the operator to trigger on a specific address pattern ("Equal" or "Not equal") or an address range.

"Length/Type (Min)" Defines the length/type pattern for all operators that require one pattern.

"Length/Typ (Max)" Defines the length/type pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:TCONdition` on page 1618
`TRIGger<m>:ETHernet:FRAMe:TMAX` on page 1618
`TRIGger<m>:ETHernet:FRAMe:TMIN` on page 1618

Frame check

Defines the frame check sequence value.

"Condition" Sets the operator to trigger on a specific sequence ("Equal" or "Not equal") or an address range.

"Frame check (Min)" Defines the frame check sequence pattern for all operators that require one pattern.

"Frame check (Max)" Defines the second frame check sequence pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:CCONdition` on page 1615
`TRIGger<m>:ETHernet:FRAMe:CMAX` on page 1616
`TRIGger<m>:ETHernet:FRAMe:CMIN` on page 1616

Error

Triggers on enabled errors.

"Preamble error" Triggers on a frame with invalid preamble.

"Length error" Triggers on an incorrect length of the sequence - when additional or missing bits are detected and the sequence of bits is not as expected.

"CRC error" Triggers on a frame that has a mismatch of the cyclic redundancy check (CRC) value between the transmitting and receiving device.

Remote command:
`TRIGger<m>:ETHernet:ERRor:CRC` on page 1614
`TRIGger<m>:ETHernet:ERRor:LENGth` on page 1615
`TRIGger<m>:ETHernet:ERRor:PREamble` on page 1615

12.11.3.2 Triggering on Ethernet

Prerequisite: A bus is configured for the Ethernet signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.

4. Select the serial bus that is set to Ethernet.
5. Tap "Type" and select the trigger type to be used for Ethernet protocol analysis. Available trigger types are "Frame Start", "Pattern", "Adv frame" and "Adv error".
6. Depending on the selected Ethernet variant, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 12.11.3.1, "Ethernet Trigger Settings", on page 661.

12.11.4 Ethernet Label List

Label lists are protocol-specific. An Ethernet label file contains two values for each identifier:

- "Ethernet Header Display": the Ethernet header display value
- "Symbolic label": symbolic name of the header, specifying its function

```
# ----------------------------------------------------------------------------
@FILE_VERSION  = 1.00
@PROTOCOL_NAME = ethernet
# ---------------------------------------------------------------------------
# Labels for Ethernet protocol
#   Column order: Ethernet Header Display, Label
#-----------------------------------------------------------------------------
# Supported MAC Address Format
# xx:xx:xx  standard 24 bit manufactory header
# xx:xx:xx:xx:xx:xx/yy  support other length headers
# yy should be the header length in decimal
# yy should be between 24 - 48
# ---------------------------------------------------------------------------
00:00:0C, Cisco
00:01:13, Olympus
00:01:14, KandaTsu
00:04:07, TopconPo
00:0B:64, KiebackP
00:1B:C5:06:C0:00/36, LuxconSy
00:1B:C5:06:D0:00/36, TesElect
00:1B:C5:06:E0:00/36, TwoDimen
00:1B:C5:06:F0:00/36, LlcEmzio
00:1F:BE, Shenzhen
00:21:8F, Avantgar
00:21:90, GoliathS
00:21:91, D-Link
00:21:92, BaodingG
00:50:C2:5F:60:00/36, Cambridg
00:50:C2:5F:70:00/36, Metrolog
```

```
00:50:C2:5F:80:00/36, GrupoEpe
18:42:2F, AlcatelL
18:44:62, RiavaNet
18:46:17, SamsungE
18:8E:D5, TpVision
18:E7:F4, Apple
40:D8:55:1C:80:00/36, SensataT
40:D8:55:1C:90:00/36, Andy-L
40:D8:55:1C:A0:00/36, RigelEng
40:D8:55:1C:B0:00/36, MgSRL
40:D8:55:1C:D0:00/36, YxlonInt
40:D8:55:1C:E0:00/36, PeterHub
40:D8:55:1C:F0:00/36, OmnikNew
40:D8:55:1D:00:00/36, WebeasyB
FC:F8:B7, TronteqE
FC:FA:F7, Shanghai
FC:FB:FB, Cisco
FC:FE:77, HitachiR
FF:FF:FF:FF:FF:FF/48, BroadCast
```

For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

12.11.5 Ethernet Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Examples

The example in Figure 12-60 shows decoded and binary signals in Ethernet 10BASE-T.

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

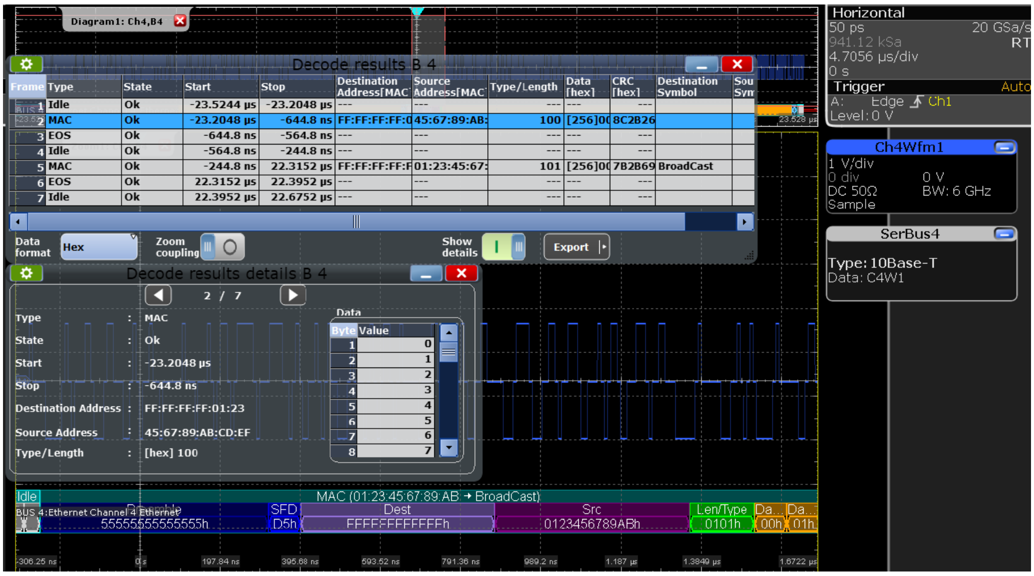


Figure 12-60: Ethernet 10BASE-T: decoded and binary signal, with decode results table and details

green brackets [...]	= start / end of frame
blue frame	= frame ok
red frame	= error frame
grey	= preamble / SFD / FrameCheck
green	= destination address
purple	= source address
brown	= address
yellow	= data

The example in Figure 12-61 shows 100Base-T1 with a failed CRC check.

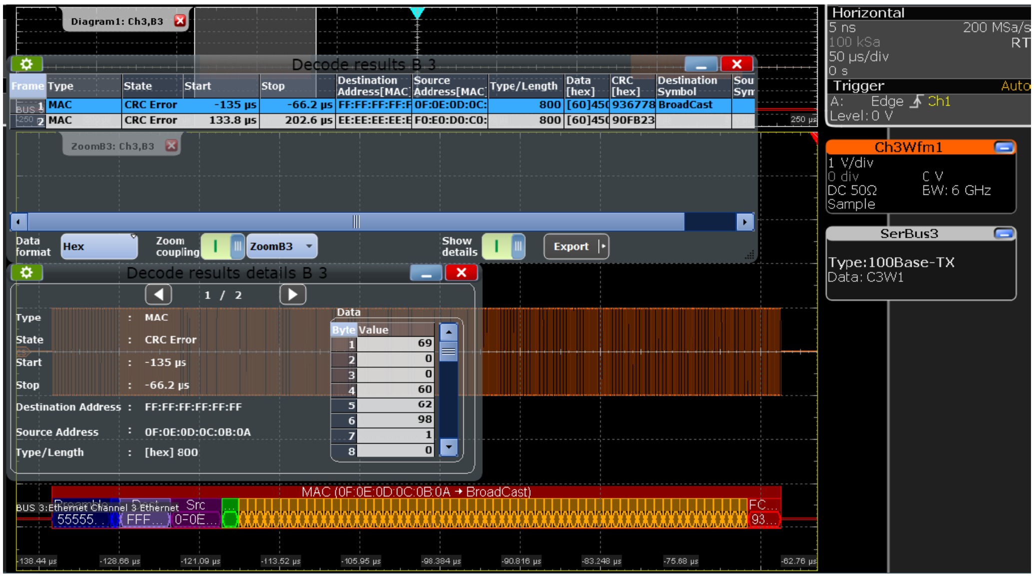


Figure 12-61: Ethernet 100BASE-TX: decoded and binary signal (zoomed view)

green brackets [...]	= start / end of frame
blue frame	= frame ok
red frame	= error frame

grey = preamble / SFD / FrameCheck
green = destination address
purple = source address
brown = address
yellow = data

The content of the "Decode results" table in Figure 12-60 is described in Table 12-13:

Table 12-13: Content of the "Decode results" table

Column	Description
Type	Frame type
State	Overall state of the frame: either OK or the relevant error condition (preamble, length)
Start	Start time of the frame
Stop	Stop time of the frame
Destination Address	Destination address of the frame
Source Address	Source address of the frame
Type/Length	The subprotocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of this field is ambivalent. It could either be the word type (specific for the subprotocol) or the word length.
Data	Values of the data bytes in a frame. The table shows a truncated version; to see all the bytes in a separate data table, activate "Show details". The data format is always hexadecimal.
CRC	FrameCheck (Cyclic Redundancy Code, CRC)
Destination Symbol	Translation (or symbolic label) of the destination address, if the label list is enabled.
Source Symbol	Translation (or symbolic label) of the source address, if the label list is enabled.
Number of Words	Number of words in the frame
Bit rate	Value of the bit rate

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:
- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:
- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.12.3, "Decode Results", on page 1618.

12.11.6 Search on Decoded Ethernet Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

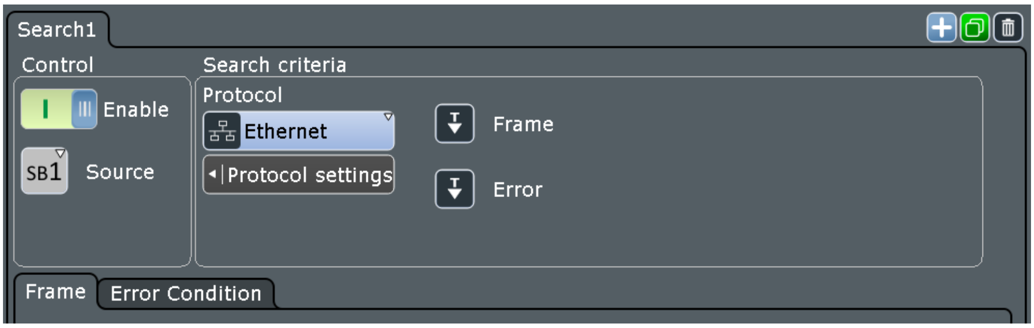
For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

If you need information on how to get started with searching Ethernet data, see Chapter 12.11.6.3, "Searching Ethernet Data", on page 674. Otherwise proceed with the Ethernet search setup.

12.11.6.1 Ethernet Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for Ethernet

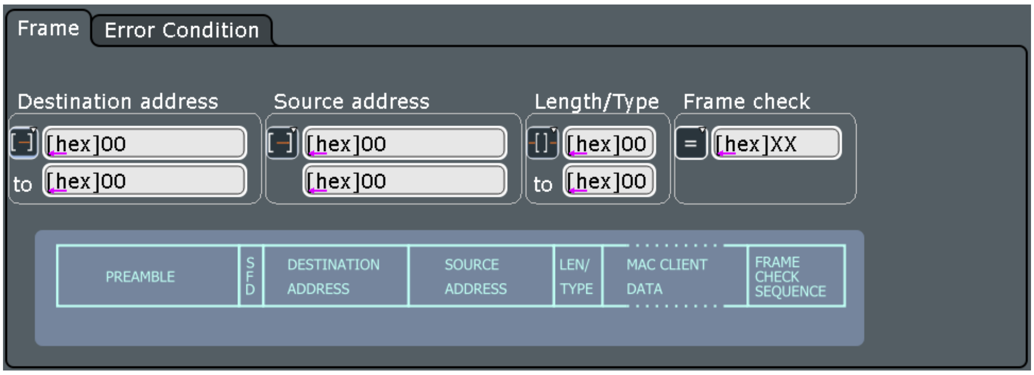
Search criteria
Define the event types to be searched. Available event types are "Frame" and "Error".



Search parameters for each event type are specified in the tabs below the "Search criteria".

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:SELect` on page 1624
`SEARch:TRIGger:ETHernet:ERRor:SELect` on page 1629

Frame
Searches for the following frame conditions: "Destination address", "Source address", "Length/Type", or "Frame check".



Destination address ← Frame
To search for a destination address, an address pattern or optionally an address range have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:DCONdition` on page 1625
`SEARch:TRIGger:ETHernet:FRAMe:DMIN` on page 1625
`SEARch:TRIGger:ETHernet:FRAMe:DMAX` on page 1625

Source address ← Frame
To search for a source address, an address pattern or optionally an address range have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:SCONdition` on page 1626
`SEARch:TRIGger:ETHernet:FRAMe:SMIN` on page 1626
`SEARch:TRIGger:ETHernet:FRAMe:SMAX` on page 1626

Length/Type ← Frame
To search for a frame length or frame type, a type/length pattern or optionally a range of type/length patterns have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:TCONdition` on page 1627
`SEARch:TRIGger:ETHernet:FRAMe:TMIN` on page 1627
`SEARch:TRIGger:ETHernet:FRAMe:TMAX` on page 1627

Frame check ← Frame
To search for a specific pattern, this pattern or optionally a range of patterns have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:CCONdition` on page 1628
`SEARch:TRIGger:ETHernet:FRAMe:CMIN` on page 1628
`SEARch:TRIGger:ETHernet:FRAMe:CMAX` on page 1628

Error Condition
Searches for the following error conditions: "Preamble Error" or "Length Error".

Preamble Error ← Error Condition
Searches for any preamble errors.

Remote command:
`SEARch:TRIGger:ETHernet:ERRor:PREamble` on page 1629

Length Error ← Error Condition
Searches for any length errors.

Remote command:
`SEARch:TRIGger:ETHernet:ERRor:LENGth` on page 1629

12.11.6.2 Ethernet Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- SEARch:RESult:ETHernet:WCOunt? on page 1630
- SEARch:RESult:ETHernet:WORD<m>:TYPE? on page 1632
- SEARch:RESult:ETHernet:WORD<m>:FTYPe? on page 1632
- SEARch:RESult:ETHernet:WORD<m>:STATe? on page 1630
- SEARch:RESult:ETHernet:WORD<m>:STARt? on page 1630
- SEARch:RESult:ETHernet:WORD<m>:STOP? on page 1631
- SEARch:RESult:ETHernet:WORD<m>:DESTaddress? on page 1631
- SEARch:RESult:ETHernet:WORD<m>:SRCaddress? on page 1631
- SEARch:RESult:ETHernet:WORD<m>:DATA? on page 1632
- SEARch:RESult:ETHernet:WORD<m>:CRC? on page 1633
- SEARch:RESult:ETHernet:WORD<m>:DSYMbol? on page 1633
- SEARch:RESult:ETHernet:WORD<m>:SSYMbol? on page 1633
- SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue? on page 1634

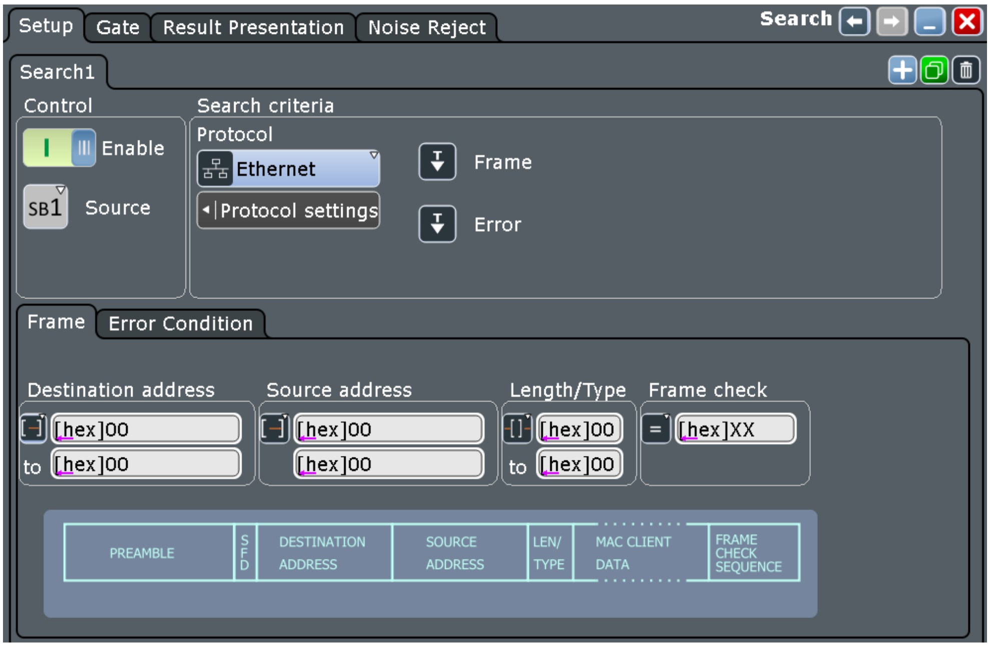
12.11.6.3 Searching Ethernet Data

Prerequisite: A serial bus is configured for the Ethernet signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to Ethernet (e.g. "SerBus1", unless already selected).

   The search dialog for Ethernet protocol analysis is opened.



4. Specify search criteria according to Chapter 12.11.6.1, "Ethernet Search Setup", on page 671.
5. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs an Ethernet decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.12 Ethernet 100BASE-T1 (Option R&S RTE-K57)

100BASE-T1 is an Ethernet based protocol specialized for the needs of the automotive industry, such as in the automotive networking applications, enabling advanced safety, comfort and infotainment features within the automobile.

Its specifications, with reference to IEEE Standard 802.3-2012, were developed by the OPEN (One-Pair Ether-Net) alliance under BroadR-Reach®.

BroadR-Reach is a Broadcom® point-to-point Ethernet physical layer (PHY) technology. Therefore, 100BASE-T1 is also known as BroadR-Reach Physical Layer (BR-PHY) or Open Alliance BroadR-Reach PHY (OABR PHY). For more details on the specifications, refer to http://www.opensig.org/about/specifications/.

This option is compatible with the MII (IEEE 802.3 Clause 22) and IEEE 802.3 MAC operating at 100Mbps.

- 100BASE-T1 Basics....676
- 100BASE-T1 Configuration....677
- 100BASE-T1 Trigger....680
- 100BASE-T1 Label List....686
- 100BASE-T1 Decode Results....687
- Search on Decoded 100BASE-T1 Data....689

12.12.1 100BASE-T1 Basics

The BR-PHY has the following objectives:

- Provides a PHY that supports full duplex operating at 100 Mbps over a pair of unshielded twisted pair (UTP) cable or better cable
- Provides compatibility with the Media Independent Interface (MII) IEEE 802.3 Clause 22 and IEEE 802.3 Media Access Controller (MAC) operating at 100 Mbps
- Achieves bit error rate (BER) of less than $1e^{-10}$

To achieve the above objectives, 100BASE-T1 uses 1000BASE-T PHYs with parts of IEEE 802.3 100BASE-TX in operation at 100 MBps and develops a new solution for the PHY sublayers, i.e. Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer as follows:

- Adopts full duplex communication of 1000BASE-T and therefore echo cancellation on a single twisted-pair channel. This feature reduces cabling while preserving the Ethernet MAC compatibility
- Adopts Pulse Amplitude Modulation (PAM-3) encoding scheme with the following encoding techniques:
  - Data encoding is carried out via a 4b3b encoder that converts the MII data (4B - four bits) with 25 MHz clock to three bits (3B) wide of data that is transmitted during one 33.3 MHz clock period
  - Symbol encoding is carried out via a one-dimensional (1D) PAM-3 encoder that converts the 3-bit groups into pairs of ternary symbols. These symbols are transmitted using three voltage levels (-1 V, 0 V and +1 V). One symbol is transmitted in each symbol period
  - Data scrambling is carried out via a sidestream scramble to randomize the sequence of transmitted symbols and avoid the presence of spectral lines in the signal spectrum

  PAM-3 has a higher spectral efficiency that limits the signaling bandwidth to 33.3 MHz instead of 65 MHz to 80 MHz in 1000BASE-T and 100BASE-TX such that communication occurs in the best part of a twisted-pair channel. This in turn improves return loss, reduces crosstalk and EMI, allows for a more aggressive Electromagnetic Compatibility (EMC) filtering and lower cost (often lower quality) cabling

In terms of the trigger and decoder development, multiple aspects of this protocol are new, specifically:

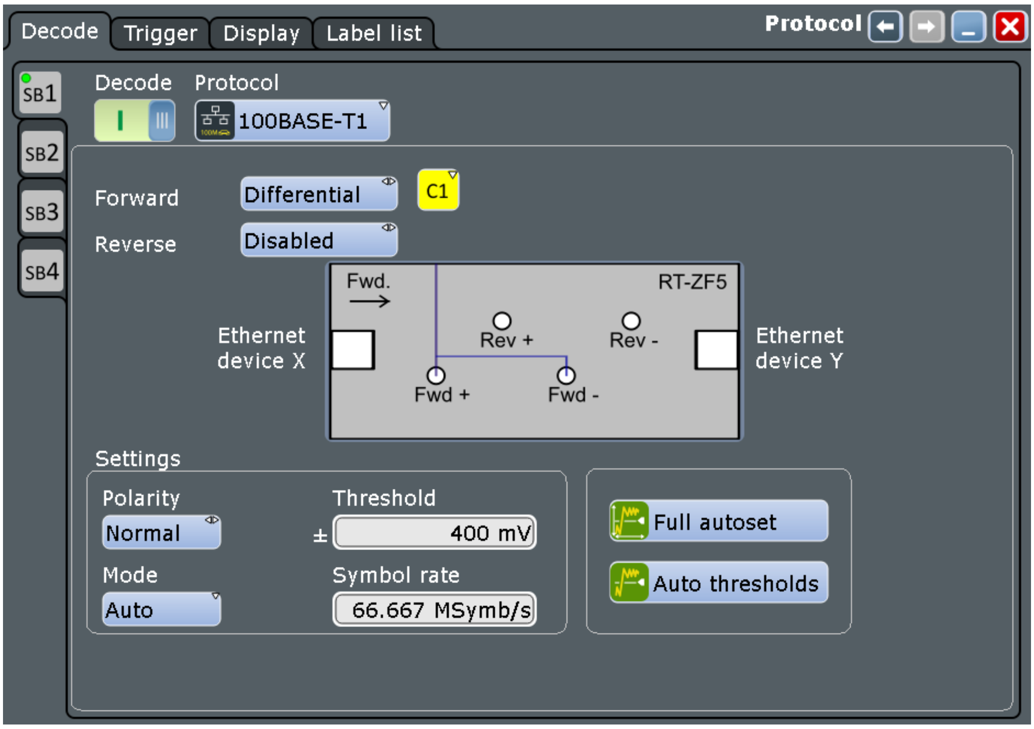
- Sidestream descrambler: Descrambles the randomized sequence of received ternary symbols
- 1D PAM-3 decoder: Decodes the received ternary symbols into groups of three bits according to a conversion chart
- 4b/3b line decoder: Decodes the groups of three bits (result of ternary pair) back to groups of four bits of the data stream

12.12.2 100BASE-T1 Configuration

12.12.2.1 100BASE-T1 Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *100BASE-T1*

Make sure that the tab of the correct serial bus is selected on the left side.



For general information on how to configure protocol parameters, see also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Reverse

Defines the reversed signal settings. The reverse direction is optional and can be disabled.

You can choose between a differential and single-ended signal and set the source channel for the waveform. Only analog channels are available for selection.

Remote command:
BUS<m>:HBTO:RTYP on page 1636
BUS<m>:HBTO:RDIF on page 1636
BUS<m>:HBTO:RDAP on page 1636
BUS<m>:HBTO:RDAM on page 1636

Forward
Defines the forward signal settings.

You can choose between a differential and single-ended signal and set the source channel for the waveform. Only analog channels are available for selection.

Remote command:
BUS<m>:HBTO:FTYP on page 1636
BUS<m>:HBTO:FDIF on page 1635
BUS<m>:HBTO:FDAP on page 1635
BUS<m>:HBTO:FDAM on page 1635

Attenuation
Sets the attenuation factor. It is used to de-amplify the reverse signal before subtracting it from the forward signal.

The effective signal amplitude passed to the decoder is:

$$Amp = (Fwd_+ - Fwd_-) - 10^{\frac{Attn}{20}}(Rev_+ - Rev_-)$$

Remote command:
BUS<m>:HBTO:ATTN on page 1634

Polarity
Selects the polarity of the data signal. You can select between "Normal" and "Inverted".

For "Normal" polarity, active high is defined as +1 V and active low is defined as -1 V. For "Inverted" polarity, active high is defined as -1 V and active low is defined as +1 V.

Remote command:
BUS<m>:HBTO:POLarity on page 1637

Thresholds
Sets the positive and negative thresholds for the digitalization of the signal line.

The signal can have three states:

- High: the signal value is higher than the positive threshold.
- Low: the signal value is below the negative threshold.
- Zero: signal is between the positive and the negative threshold.

There are several ways to set the values of the threshold:
- "Thresholds"
  Enter the value directly.
- "Full autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
  See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Auto threshold"

Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:HBTO:THReshold` on page 1637
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Mode
Selects the desired direction of the full-duplex signal for analysis. If set to "Auto", the decoder automatically detects the "Master" or "Slave" mode.

Remote command:
`BUS<m>:HBTO:MODE` on page 1637

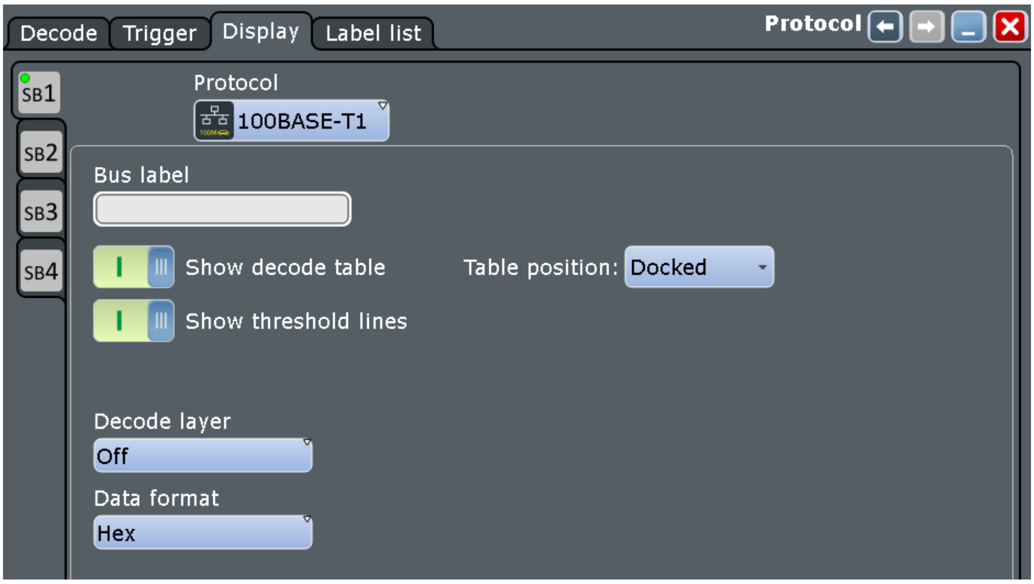
Symbol rate
Defines the transmission rate of ternary symbols which is by default 66.67 MSymb/s. This parameter should be rarely changed.

Remote command:
`BUS<m>:HBTO:SYMRate` on page 1637

12.12.2.2 100BASE-T1 Display Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol = 100BASE-T1" > "Display" tab

To enhance the decode possibilities of the 100BASE-T1 protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Common display settings are explained in Chapter 12.1.3, "Display", on page 475.

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Off"	No decode layer is displayed.
"Ternary symbols"	Decoded ternary symbols.
"Scrambled bits"	Scrambled ternary bits.
"Descrambled bits"	Descrambled ternary bits.
"Reversed Bits"	Reserved bits

12.12.2.3 Configuring 100BASE-T1 Signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

Serial bus setup

1. Press the [PROTOCOL] key on the front panel.
2. At the left hand-side, select the vertical tab of the serial bus (SB1–SB4) you want to set up.
3. Select the "Decode" tab.
4. Tap "Protocol" and select the protocol: "100BASE-T1".
5. Optionally, you can enter a "Bus label" in the "Display" tab.
6. Select the type and source for the reversed and forward signal.
7. Select the polarity ("Normal" or "Inverted") of the signal.
8. Set the logical thresholds.
9. Select the "Mode" and the "Symbol rate".
10. Enable "Decode", if available.

12.12.3 100BASE-T1 Trigger

12.12.3.1 100BASE-T1 Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = 100BASE-T1"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to data sheet.

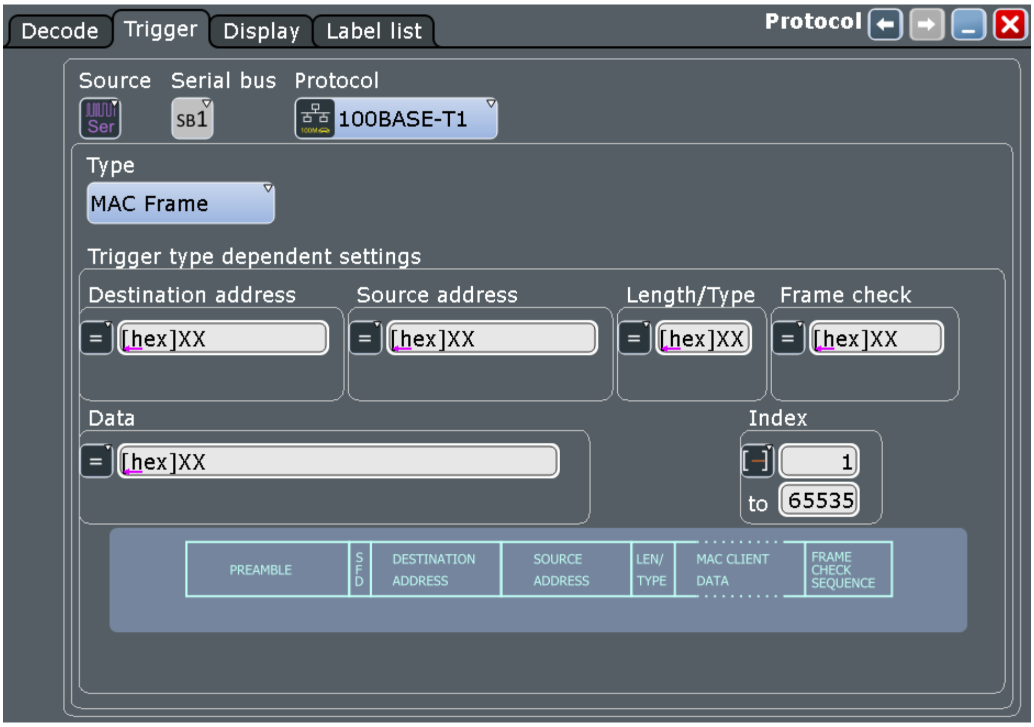


Figure 12-62: 100BASE-T1 trigger event settings dialog

Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Type

Selects the trigger type for 100BASE-T1 analysis.

Remote command:

`TRIGger<m>:HBTO:TYPE` on page 1638

Frame Start ← Type

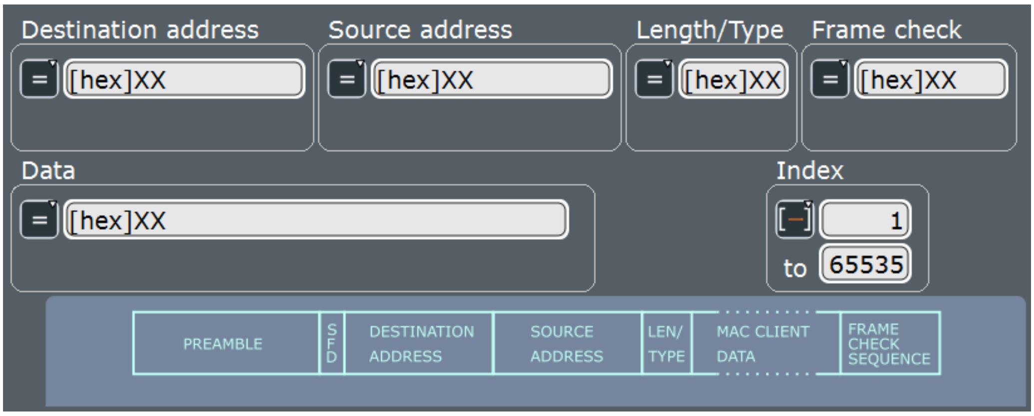
Triggers on the start of frame.

MAC Frame ← Type

Triggers on a Media Access Control (MAC) frame. You can specify:

- Destination address
- Source address
- Length/Type
- Frame check
- Data
- Index

The frame contains addresses of the devices and MAC control information that define how to go about transmitting and receiving frames.



IDLE ← Type

Triggers on an idle frame. The frame is used for clock synchronization.

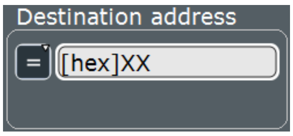
Error ← Type

Triggers on the specified error frame.

Destination address

Sets the specified destination address to be triggered on. The destination address setup consists of the condition and one or two data patterns.

The destination address corresponds to the address of the interface in the device that receives the frame. If the destination address does not match the interface's own Ethernet address, then the interface is free to ignore the rest of the frame.



"Condition" Defines the operator to set a specific destination address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"

Defines the bit pattern of the destination address pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

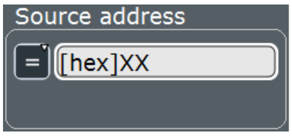
"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:HBTO:DADDress:CONDition on page 1639
TRIGger<m>:HBTO:DADDress:MIN on page 1639
TRIGger<m>:HBTO:DADDress:MAX on page 1640

Source address

Sets the specified source address to be triggered on. The source address setup consists of the condition and one or two data patterns.

The source address is the physical address of the device that sends the frame.



"Condition" Defines the operator to set a specific source address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the source address pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

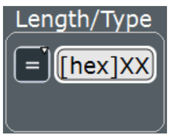
"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:HBTO:SADDress:CONDition on page 1640
TRIGger<m>:HBTO:SADDress:MIN on page 1640
TRIGger<m>:HBTO:SADDress:MAX on page 1641

Length/Type

Sets the specified Length/Type to be triggered on. The Length/Type setup consists of the condition and one or two data patterns.

The value in this field indicates the manner in which the field is being used, either as a length or type field. As a length field, the value in the field indicates the number of logical link control (LLC) data octets that follow in the data field of the frame. As a type field, the value in the field is used to indicate the type of protocol data being carried in the data field of the frame.



"Condition" Defines the operator to set a specific Length/Type, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
: Defines the bit pattern of the Length/Type pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

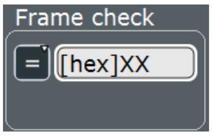
"Data Max"
: The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:LENGth:CONDition` on page 1641
`TRIGger<m>:HBTO:LENGth:MIN` on page 1641
`TRIGger<m>:HBTO:LENGth:MAX` on page 1641

Frame check

Sets the specified frame check to be triggered on. The frame check setup consists of the condition and one or two data patterns.

This field contains a value that is used to check the integrity of the various bits in the frame fields (excluding the preamble / SFD).



"Condition"
: Defines the operator to set a specific frame check, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
: Defines the bit pattern of the frame check pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max"
: The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:CRC:CONDition` on page 1642
`TRIGger<m>:HBTO:CRC:MIN` on page 1642
`TRIGger<m>:HBTO:CRC:MAX` on page 1642

Data

Sets the specified data to be triggered on. The data setup consists of the condition and one or two data patterns.

"Condition"
: Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

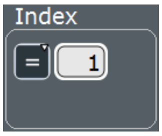
"Data Min/Data"
Defines the bit pattern of the data pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:HBTO:DATA:DCONdition on page 1643
TRIGger<m>:HBTO:DATA:DMIN on page 1643
TRIGger<m>:HBTO:DATA:DMAX on page 1643

Index

Sets the specified value or range within this series of data that is considered for the analysis. The index setup consists of the condition and one or two index values.



"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

"Index Min/Index"
Defines the bit pattern of the index pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

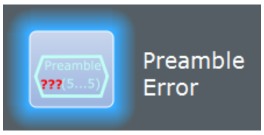
Remote command:
TRIGger<m>:HBTO:DATA:ICONdition on page 1644
TRIGger<m>:HBTO:DATA:IMIN on page 1644
TRIGger<m>:HBTO:DATA:IMAX on page 1644

Errors

Sets the type of error events to be triggered on.

Preamble Error ← Errors

Triggers on a frame with invalid preamble.



Remote command:
TRIGger<m>:HBTO:ERRor:PREamble on page 1644

CRC Error ← Errors
Triggers on a frame that has a mismatch of the Cyclic Redundancy Check (CRC) value between the transmitting and receiving device.

Remote command:
`TRIGger<m>:HBTO:ERRor:CRC` on page 1645

SFD Error ← Errors
Triggers on a frame with invalid Start Frame Delimiter (SFD).

Remote command:
`TRIGger<m>:HBTO:ERRor:SFD` on page 1645

12.12.3.2 Triggering on 100BASE-T1

Prerequisite: A bus is configured for the 100BASE-T1 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to 100BASE-T1.
5. Tap "Type" and select the trigger type to be used for 100BASE-T1 protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.

For details, see Chapter 12.12.3, "100BASE-T1 Trigger", on page 680.

12.12.4 100BASE-T1 Label List

Label lists are protocol-specific. Label lists for 100BASE-T1 are available in CSV and PTT format.

A 100BASE-T1 label file contains two values for each ethernet header:

- Ethernet Header Display

- Symbolic label: name of the address, specifiing its function in the bus network.

12.12.5 100BASE-T1 Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Examples

The example in Figure 12-64 shows decoded signals in 100BASE-T1.

Figure 12-63: 100BASE-T1 decoded signal

green brackets [...] = start / end of frame
dark cyan frame = frame ok
red frame = error frame
light purple = destination address

purple	= source address
orange or yellow	= data
light gray	= idle
gray	= filler
dark gray	= reserved
light blue	= check
blue	= marker
dark blue	= sync / SFD
lime	= count

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

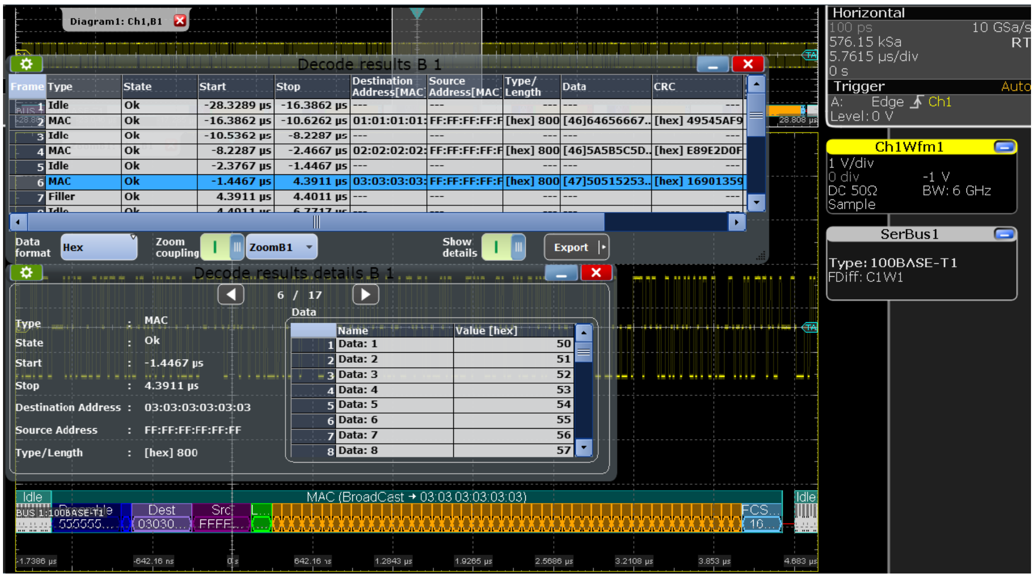


Figure 12-64: 100BASE-T1 decode results table and details

The content of the "Decode results" table in Figure 12-64 is described in Table 12-14:

Table 12-14: Content of the "Decode results" table

Column	Description
Frame	Frame count
Type	Type of frame (e.g. Idle, MAC or data)
State	Overall state of the frame: either OK or the relevant error condition (preamble, CRC or SFD)
Start	Start time of the frame
Stop	Stop time of the frame
Destination Address	Destination address of the frame
Source Address	Source address of the frame
Type/Length	The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of this field is ambivalent. It could either be the word type (specific for the sub-protocol) or the word length.

Column	Description
Data	Values of the data bytes in a frame. The table shows a truncated version; to see all the bytes in a separate data table, activate "Show details". The data format is always hexadecimal.
CRC	Frame Check (Cyclic Redundancy Code, CRC)
Bit rate	Value of the bit rate

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.13.3, "Decode Results", on page 1645.

12.12.6 Search on Decoded 100BASE-T1 Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

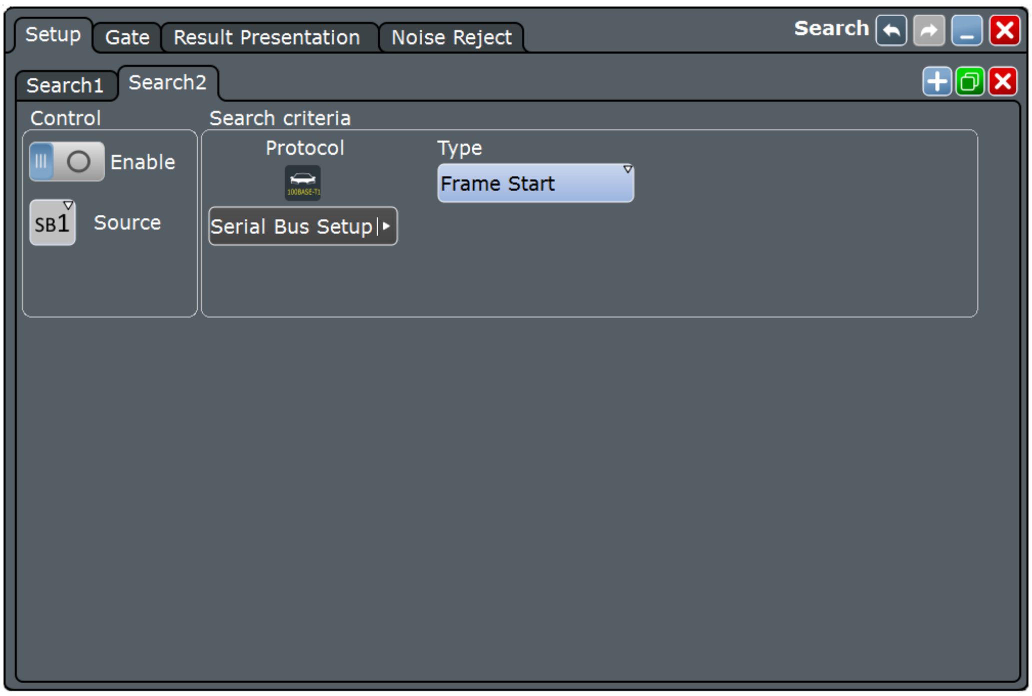
To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

12.12.6.1 100BASE-T1 Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for 100BASE-T1

Search criteria
Use the "Search criteria" dialog to define the event types to be searched.



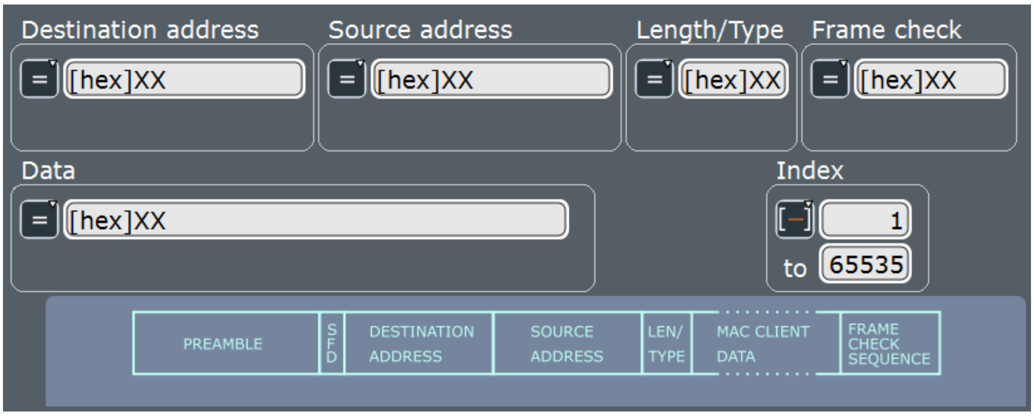
Type
Selects the search type for 100BASE-T1 analysis.

Remote command:
`SEARch:TRIGger:HBTO:TYPE` on page 1651

Frame Start ← Type
Searches for the start of frame.

MAC Frame ← Type
Searches for a Media Access Control (MAC) frame that matches the specified Destination address, Source address, Length/Type, Frame check, Data or Index condition.

The frame contains addresses of the devices and MAC control information that define how to go about transmitting and receiving frames.



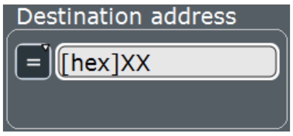
FILLER ← Type
Searches for a Filler frame. The frame is used to maintain transmission activity.

IDLE ← Type
Searches for an idle frame. The frame is used for clock synchronization.

Error ← Type
Searches for the specified error frame.

Destination address
Sets the specified destination address to be searched for. The destination address setup consists of the condition and one or two data patterns.

The destination address corresponds to the address of the interface in the device that receives the frame. If the destination address does not match the interface's own Ethernet address, then the interface is free to ignore the rest of the frame.



"Condition" Defines the operator to set a specific destination address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the destination address pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

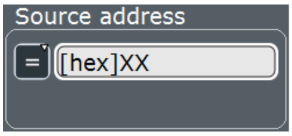
"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:HBTO:DADDress:CONDition` on page 1652
`SEARch:TRIGger:HBTO:DADDress:MIN` on page 1652
`SEARch:TRIGger:HBTO:DADDress:MAX` on page 1653

Source address

Sets the specified source address to be searched for. The source address setup consists of the condition and one or two data patterns.

The source address is the physical address of the device that sends the frame.



"Condition" Defines the operator to set a specific source address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the source address pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

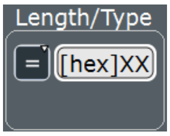
"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:HBTO:SADDress:CONDition` on page 1653
`SEARch:TRIGger:HBTO:SADDress:MIN` on page 1653
`SEARch:TRIGger:HBTO:SADDress:MAX` on page 1654

Length/Type

Sets the specified Length/Type to be searched for. The Length/Type setup consists of the condition and one or two data patterns.

The value in this field indicates the manner in which the field is being used, either as a length or type field. As a length field, the value in the field indicates the number of logical link control (LLC) data octets that follow in the data field of the frame. As a type field, the value in the field is used to indicate the type of protocol data being carried in the data field of the frame.



"Condition" Defines the operator to set a specific Length/Type, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the Length/Type pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

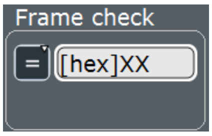
"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:HBTO:LENGth:CONDition` on page 1654
`SEARch:TRIGger:HBTO:LENGth:MIN` on page 1654
`SEARch:TRIGger:HBTO:LENGth:MAX` on page 1655

Frame check

Sets the specified frame check to be searched for. The frame check setup consists of the condition and one or two data patterns.

This field contains a value that is used to check the integrity of the various bits in the frame fields (excluding the preamble / SFD).



"Condition" Defines the operator to set a specific frame check, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the frame check pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:HBTO:CRC:CONDition` on page 1655
`SEARch:TRIGger:HBTO:CRC:MIN` on page 1655
`SEARch:TRIGger:HBTO:CRC:MAX` on page 1656

Data

Sets the specified data to be searched for. The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

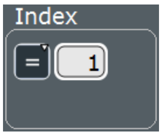
"Data Min/Data"
Defines the bit pattern of the data pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:HBTO:DATA:DCONdition on page 1656
SEARch:TRIGger:HBTO:DATA:DMIN on page 1656
SEARch:TRIGger:HBTO:DATA:DMAX on page 1657

Index

Sets the specified value or range within this series of data that is considered for the search. The index setup consists of the condition and one or two index values.



"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

"Index Min/Index"
Defines the bit pattern of the index pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

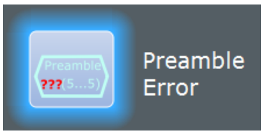
Remote command:
SEARch:TRIGger:HBTO:DATA:ICONdition on page 1657
SEARch:TRIGger:HBTO:DATA:IMIN on page 1657
SEARch:TRIGger:HBTO:DATA:IMAX on page 1658

Errors

Sets the type of error events to be searched for.

Preamble Error ← Errors

Searches for a frame with invalid preamble.



Remote command:
SEARch:TRIGger:HBTO:ERRor:PREamble on page 1658

CRC Error ← Errors

Searches for a frame that has a mismatch of the Cyclic Redundancy Check (CRC) value between the transmitting and receiving device.

Remote command:
SEARch:TRIGger:HBTO:ERRor:CRC on page 1658

SFD Error ← Errors

Searches for a frame with invalid Start Frame Delimiter (SFD).

Remote command:
SEARch:TRIGger:HBTO:ERRor:SFD on page 1658

12.12.6.2 100BASE-T1 Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- SEARch:RESult:HBTO:FCOunt? on page 1659
- SEARch:RESult:HBTO:FRAMe<m>:TYPE? on page 1659
- SEARch:RESult:HBTO:FRAMe<m>:STATe? on page 1660
- SEARch:RESult:HBTO:FRAMe<m>:STARt? on page 1660
- SEARch:RESult:HBTO:FRAMe<m>:STOP? on page 1660
- SEARch:RESult:HBTO:FRAMe<m>:DESTaddress? on page 1661
- SEARch:RESult:HBTO:FRAMe<m>:SRCaddress? on page 1661
- SEARch:RESult:HBTO:FRAMe<m>:DATA? on page 1661
- SEARch:RESult:HBTO:FRAMe<m>:CRC? on page 1662
- SEARch:RESult:HBTO:FRAMe<m>:NUMWords? on page 1662
- SEARch:RESult:HBTO:FRAMe<m>:FTYPe? on page 1662
- SEARch:RESult:HBTO:FRAMe<m>:DSYMbol? on page 1663
- SEARch:RESult:HBTO:FRAMe<m>:SSYMbol? on page 1663
- SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE? on page 1663
- SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue? on page 1664

12.12.6.3 Searching 100BASE-T1 Data

Prerequisite: A serial bus is configured for the 100BASE-T1 signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to 100BASE-T1 (e.g. "SB1", unless already selected).

   The search dialog for 100BASE-T1 protocol analysis opens.
4. Specify search criteria according to Chapter 12.12.6, "Search on Decoded 100BASE-T1 Data", on page 689.
5. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs an 100BASE-T1 decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.13 SENT (Option R&S RTE-K10)

Single Ended Nibble Transmission (SENT) is a serial transmission interface protocol originally specified for the communication of sensors and control units in automotive electronics.

SENT is a protocol standard governed by Society of Automotive Engineers (SAE J2716). For detailed information, refer to the SENT standard specification on http://www.sae.org.

The SENT protocol is used exclusively in automotive applications, as for example electrical power steering, advanced driver assistance like parking assist or sensing of pressure, throttle position, pedal position, airflow mass, liquid level, etc.

The R&S RTE option R&S RTE-K10 provides serial triggering, decoding and a highly customizable search on decoded SENT signals.

12.13.1 The SENT Protocol

This chapter provides an overview of the protocol characteristics, encoding scheme, identifiers and trigger possibilities.

The SENT protocol transmits signal values point-to-point from a sensor to a controller (electronic control unit ECU), unidirectional. In contrast to conventional measurements, you can receive multiple data parameters via the SENT interface in a single transmission. Nevertheless, SENT is characterized by its simplicity and yet very high customizability to meet the individual requirements of the applications.

SENT operates via a three wire connection, a signal line, a supply voltage line for the sensor and a ground line. It transmits data digitally in variable timing units and evaluates the time between two falling edges (single edges). The signal is amplitude modulated with a constant amplitude voltage. Thus influences of interfering signals are not critical.

SENT key features

Main characteristics of SENT are:

- serial communication protocol
- 3 wires: SENT (signal line), 5V (voltage line), GND (ground line)
- output only, from sensor to receiver
- point-to-point transmission, no bus
- digital transmission
- high baud rate
- data transmission in variable timing units of 4 bits (1 nibble) between two falling edges
- transmitter-specific clock period (tick)
- time measured between single falling edges

12.13.1.1 SENT Transmission Concept

A sensor converts the analog measured data to a digital signal, and thus transmits a series of pulses to the receiver. The receiver, e.g. an ECU processes the received signal also digitally.

The format of a SENT message frame has a fixed pulse order and a transmitter-specific clock period. The total transmission time varies depending on the clock variation of the transmitter and the transmitted data values. The data pulses embedded in the transmission sequence represent one or multiple data parameters to be communicated. The last pulses in a message frame are the CRC check pulse, allowing the receiver to perform a number of diagnostic tests, and an optional pause pulse.

A SENT transmission starts without a request from the receiver. Consecutive sequences are transmitted continuously after the falling edge of the last pulse.

The SENT protocol distinguishes between two channel types:

- **Fast channel:** transmits primary data, i.e. sensor readings like temperature, pressure, mass air flow, throttle position.

- **Slow channel:** transmits secondary data consisting of transfer characteristics, sensor ID, type, manufacturer diagnostic, etc.
  The slow channel transmission provides two serial message formats *Short* and *Enhanced* for customizing the secondary data.

The data of both, the fast and the slow channels is transmitted simultaneously, by including two bits of a slow channel message in the message frame of the fast channel. Even though it requires many fast channel messages to complete a slow channel message, you can use this function to transmit several slow channel messages with minimal impact on the primary sensor data and the data rate.

12.13.1.2 SENT Message Definitions

SENT terms

See the specific terms and definition used in SENT protocol:

- **Tick (clock tick):** basic unit of time
  – transmitter-specific nominal clock period
  – 3 µs < clock tick <90 µs, with max. 20 % clock variation
- **Nibble:** minimum unit of data
  – used to transmit data
  – variable timing units between two falling edges

SENT Fast Channel

The SENT protocol enables you to transmit measurements of multiple sensors in one transmission sequence with data signals of varying length. The diagram in Figure 12-65 shows, for example, the encoding scheme for two 12-bit data signals.

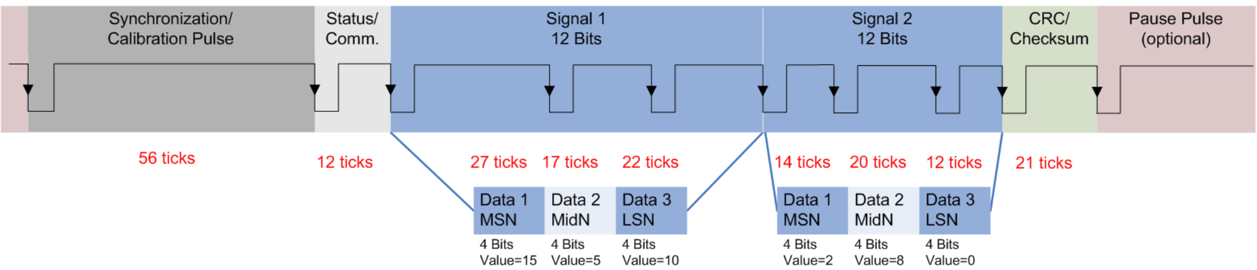


Figure 12-65: Example of a SENT transmission sequence

The format of a SENT transmission sequence consists of the following pulses:

- **Synchronization/Calibration Pulse:**
  – initial sequence of the receiver
  – the start condition is the falling edge of the last pulse (CRC or Pause)
  – nominal pulse period is 56 clock ticks
  – measures the actual clock variation of the transmitter and calculates the tick timing
- **Status/Communication Pulse (Nibble)**

- one 4 bit pulse
- communicates status and enables the sensor to include slow channel message bits
  0: (LSB) specific application
  1: specific application
  2: Serial Data message or specific application (e.g. Infineon TLE4998S)
  3: (MSB) 1= message start; 0=Serial Data message or specific application (e.g. Infineon TLE4998S)
- 12 to 27 clock ticks
- not included in CRC frame calculation

- **Data Pulses (Nibbles)**
  - one up to six 4 bit data nibbles
  - 12 to 27 clock ticks pulse period
  - initial logic 0 time with ≥5 ticks, subsequent logical 1 with variable duration
- **CRC/Checksum**
  - one 4 bit pulse
  - used for error checking of data nibbles (status nibble not included)
  - detects single bit, odd number of nonconsecutive and single burst errors
- **Pause Pulse**
  - one optional pulse
  - variable pulse length: 12 to 768 clock ticks
  - can be used to create a transmission with constant number of clock ticks

SENT Slow Channel

Short Serial Messages

For transmission of a slow channel message, 2 bits are included in a fast channel message, see the status nibble (Bit 2,3) in Figure 12-66.

A short serial message needs 16 fast channel messages until it is completely transmitted. Prerequisite for the complete transmission of the slow channel message are 16 consecutive error-free fast channel transmissions.

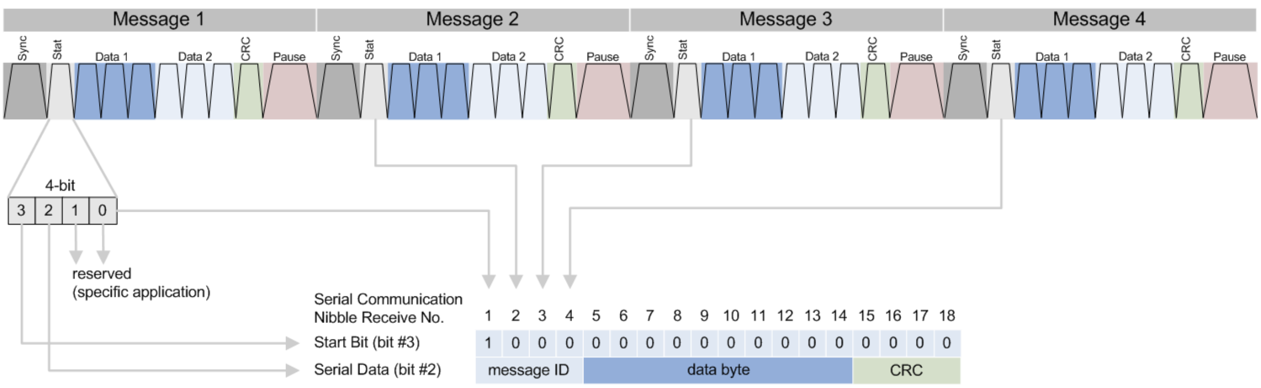


Figure 12-66: One serial message, composed of 16 SENT consecutive fast channel transmissions

Enhanced Serial Messages

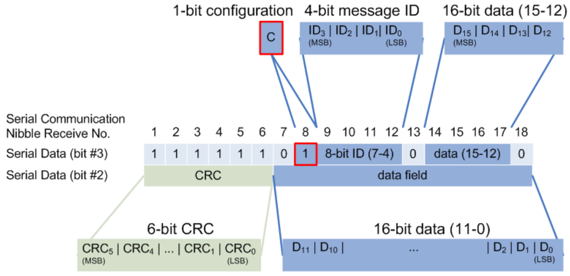
The transmission of an enhanced serial message format requires 18 fast channel transmissions. Each slow channel message is assigned a message ID, which is transmitted with the data.

The enhanced serial message format provides two alternatives for configuring the message:

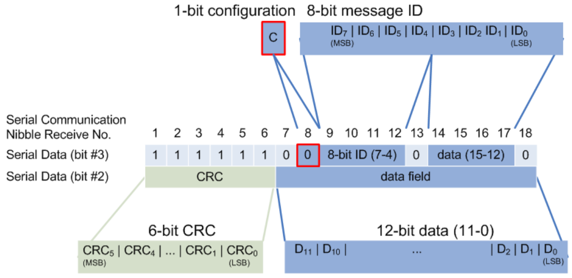
- 4 bit ID and 16 bit data
- 8 bit ID and 12 bit data

The graphs below illustrate the variants.

Table 12-15: Enhanced serial message formats



16 bit data and 4 bit message ID



12 bit data and 8 bit message ID

Trigger

The R&S RTE can trigger on various parts of SENT pulses. The data line must be connected to an input channel, triggering on math and reference waveforms is not possible.

SENT enables you to trigger on:

- Calibration/synchronization pulse
- Transmission sequence
- Serial messages
- Error conditions

12.13.2 SENT Configuration

12.13.2.1 SENT Configuration

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *SENT*

Make sure that the tab of the correct serial bus is selected on the left side.

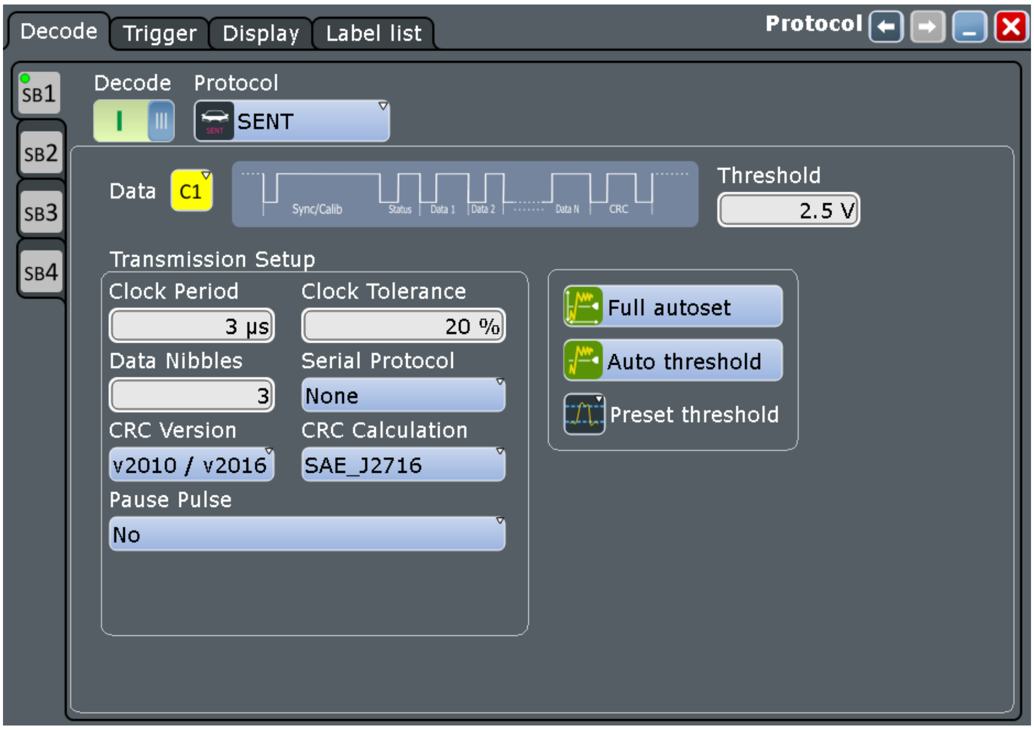


Figure 12-67: SENT protocol configuration dialog

For general information on how to configure protocol parameters, see also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Data

Sets the source of the data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Remote command:
`BUS<m>:SENT:DATA:SOURce` on page 1664

Threshold

Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold.

There are several ways to set the threshold:

- "Threshold"
  Enter the value directly in the field.
- "Full autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
  See also Chapter 12.1.2, "Full Autoset", on page 475.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Preset thresholds"

Selects the default threshold voltage for various signal technologies from a list. The value is set to "Manual" if the threshold was set with "Auto threshold", or was entered directly.

Remote command:
`BUS<m>:SENT:DATA:THReshold` on page 1665
`BUS<m>:SENT:TECHnology` on page 1665
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:FAUToset` on page 1383

Clock Period
Sets the transmitter-specific nominal clock period (clock tick).

The clock period and signal length determine the speed of transmission.

Remote command:
`BUS<m>:SENT:CLKPeriod` on page 1665

Clock Tolerance
Specifies a tolerated deviation of the clock.

Remote command:
`BUS<m>:SENT:CLKTolerance` on page 1666

Data Nibbles
Sets the number of data units in a single transmission sequence.

The maximum number of data nibbles is 6.

Remote command:
`BUS<m>:SENT:DNIBbles` on page 1666

Serial Protocol
Selects the protocol format in the transmitted signal.

"Short"	Short serial messages.
"Enhanced"	Enhanced serial messages.
"None"	No serial messages. Transmission sequences only.

Remote command:
`BUS<m>:SENT:SFORmat` on page 1666

CRC Version
Selects the version the CRC check is based on.

"Legacy"	Based on the crc calculation version used earlier than 2010.
"v2010"	Based on the recent crc calculation version updated in 2010.

Remote command:
`BUS<m>:SENT:CRCVersion` on page 1666

CRC Calculation
Selects the method for CRC calculation.

"SAE_J2716" Calculates the CRC according to the SAE standard.
For this method the checksum is calculated over all nibbles except the communication and status nibble.

"TLE_4998X" Calculates the CRC according to the standard computing method for Infineon TLE_4998X sensors.
For this method the checksum is calculated over all nibbles except the communication nibble.

Remote command:
BUS<m>:SENT:CRCMethod on page 1667

Pause Pulse
Determines whether a pause pulse is transmitted after the checksum nibble.

You can use this pulse to create a transmission with a constant number of clock ticks. The pause pulse length can be between a minimum of 12 clock ticks up to 768 (3*256) ticks at a maximum.

"No"
No pause pulse between the transmission sequences.

"Yes" Pause pulse with fixed length at the end of each transmission sequence.
The R&S RTE computes the length of the pause pulse automatically.

"For constant frame length"
Pause pulse with dynamic length to maintain a fixed transmission sequence length.
To define the constant frame length, set the number of clock ticks under Frame Length in clock ticks.

Remote command:
BUS<m>:SENT:PPULse on page 1667

Frame Length in clock ticks
Determines the frame length in terms of ticks. The dialog displays this settings parameter, if the signal has a constant frame length.

Remote command:
BUS<m>:SENT:PPFLength on page 1667

12.13.2.2 Configuring SENT Protocol

The configuration of the SENT is simple - assign the two lines to input channels, and set the thresholds.

For details on configuration settings, see Chapter 12.13.2.1, "SENT Configuration", on page 700.

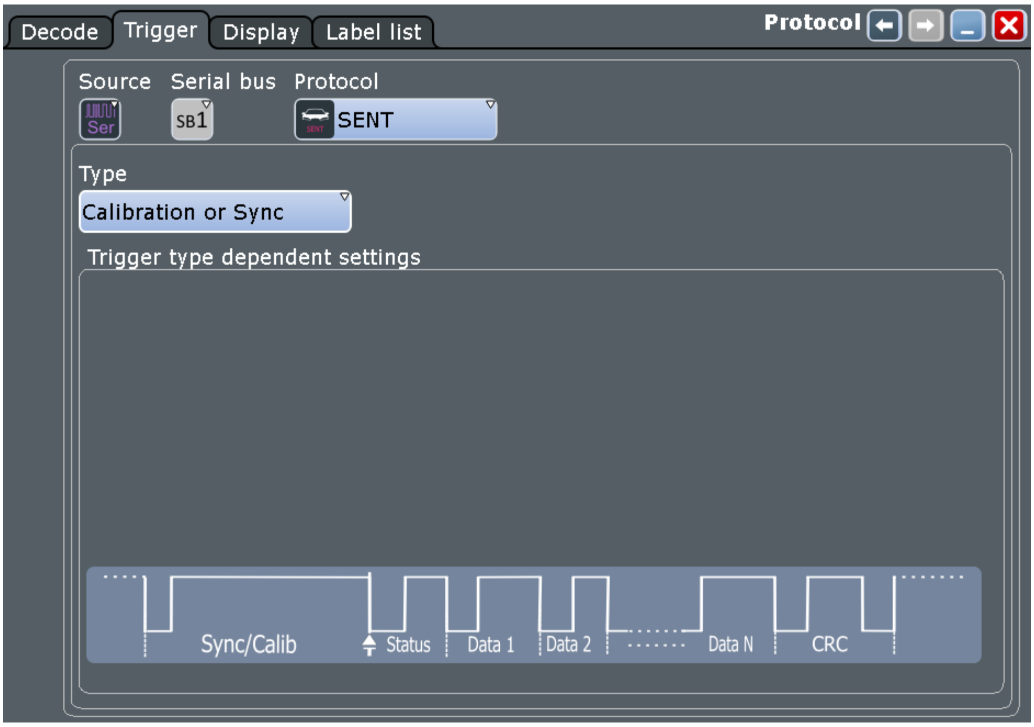
1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.

4. Tap the "Protocol" button and select the protocol: "SENT".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Tap the "Data" button, and select the waveform of the data line.
7. If necessary, set the "Transmission Setup" parameters.
8. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to an automatic value with "Full autoset"/ "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.
9. Enable "Decode", if available.

12.13.3 SENT Trigger

12.13.3.1 SENT Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = SENT"



Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Type

Selects the event of the SENT transmission sequence or message to be triggered on.

The RTE triggers always on the falling edge of a pulse, i.e. at the end of the selected type nibble in a transmission sequence.

"Calibration or Sync"

Triggers at the end of the "Synchronization/Calibration" pulse.

This setting does not require any input parameters.

"Transmission Sequence"

Triggering depends on the additional selectable "Sequence" parameter:

- "Sequence > Status": triggers at the end of the "Status" nibble.

- "Sequence > Status+Data": triggers at the end of the last data nibble.

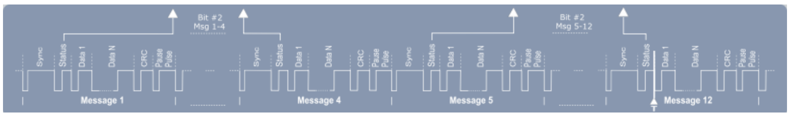
Description of trigger type specific settings: "Transmission Sequence setup" on page 706

"Serial Message"

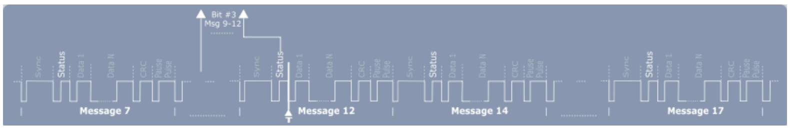
Triggering on a serial message depends on the serial protocol selected with Serial Protocol > Short | Enhanced and the associated setting parameters:

- "Short" serial message
  - "Sequence > Identifier": triggers at the end of the "ID".

  - "Sequence > ID+Data": triggers at the end of the "ID and Data".



  Description of trigger type specific settings: "Serial Message setup" on page 708
- "Enhanced" serial message
  - "Sequence > Identifier": triggers at the end of the "ID".



  - "Sequence > ID+Data": triggers at the end of the "ID and Data".

  Description of trigger type specific settings: "Serial Message setup" on page 708

"Error condition"

Triggers if certain errors occur.

You can select the following error events for triggering:

- "Form Error"
- "Calibration Pulse Error"
- "Pulse Period Error"
- "CRC Error"
- "Irregular Frame Length Error"

Description of error-specific trigger conditions, see "Error conditions setup" on page 710.

Remote command:
`TRIGger<m>:SENT:TYPE` on page 1668

Transmission Sequence setup

Configures the trigger conditions for trigger type transmission sequence.

Note: The displayed parameters depend on the selected "Sequence". The instrument displays the data setting parameters when you select "Status+Data", see "Type" on page 705.

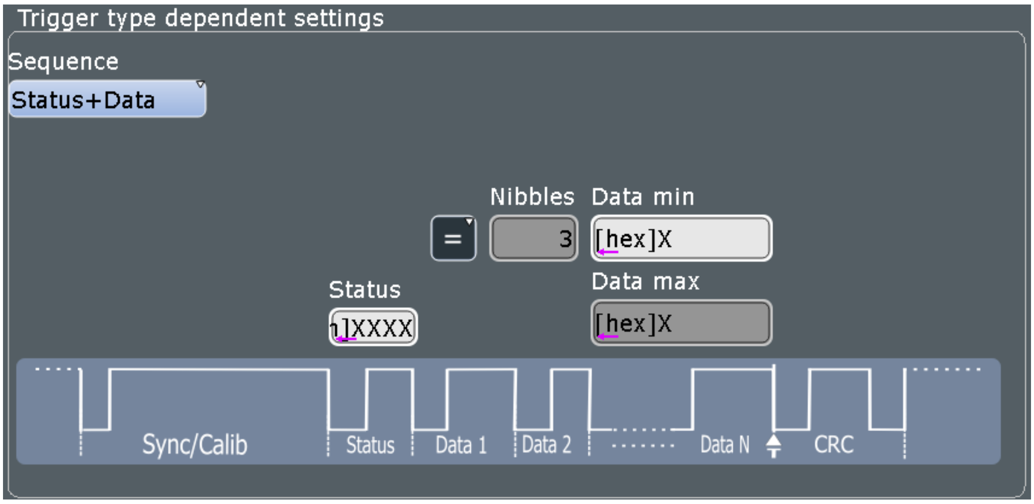


Figure 12-68: Trigger setting parameters of a transmission sequence

Sequence ← Transmission Sequence setup
Selects the condition for triggering in a single transmission sequence.

You can trigger on the end of the status nibble or the combination of the status and data nibbles.

Remote command:
`TRIGger<m>:SENT:TTYPe` on page 1669

Status ← Transmission Sequence setup
Defines the data bits for the status nibble.

Remote command:
`TRIGger<m>:SENT:STATus` on page 1669

Condition ← Transmission Sequence setup
Selects the operator to define a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`TRIGger<m>:SENT:TDCN` on page 1670

Nibbles ← Transmission Sequence setup
Displays the number of data nibbles of the transmission sequence.

Remote command:
`BUS<m>:SENT:DNIBbles` on page 1666

Data min ← Transmission Sequence setup

Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
`TRIGger<m>:SENT:TDMN` on page 1670

Data max ← Transmission Sequence setup

The second data pattern is required to specify a range with conditions "In range" and "Out of range".

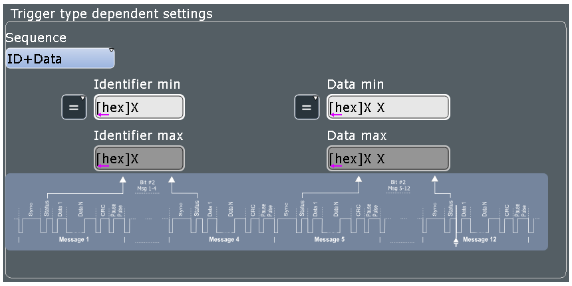
Remote command:
`TRIGger<m>:SENT:TDMX` on page 1670

Serial Message setup

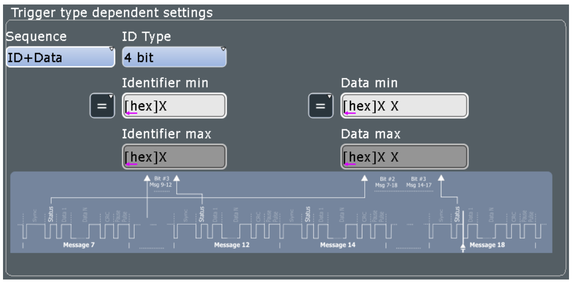
Configures the trigger conditions for a serial message.

Note: The displayed parameters depend on the selected "Sequence". The instrument indicates the data setting parameters when you select "ID+Data", see "Type" on page 705.

Trigger setting parameters of the serial message types



Short Serial Message



Enhanced Serial Message

Sequence ← Serial Message setup

Selects the condition for triggering in a serial message.

You can trigger on the end of an identifier nibble or the combination of the identifier and data nibble(s).

Remote command:
`TRIGger<m>:SENT:STYPe` on page 1670

ID Type ← Serial Message setup

Selects the message ID format for the enhanced serial message type.

You can select either 4-bit or 8-bit message ID.

Remote command:
`TRIGger<m>:SENT:SIDType` on page 1671

Identifier Condition ← Serial Message setup

The available operators:

- Equal, Not equal
- Less than, Greater than

- Less or equal, Greater or equal
- In range, Out of range

Remote command:
TRIGger<m>:SENT:SICN on page 1671

Identifier min ← Serial Message setup
Defines the bit pattern of the message identifier. In binary format, use the following characters: 1; 0; or X (any bit).

The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Remote command:
TRIGger<m>:SENT:SIMN on page 1671

Identifier max ← Serial Message setup
The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:SENT:SIMX on page 1672

Data Condition ← Serial Message setup
Selects the operator to set a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
TRIGger<m>:SENT:SDCN on page 1672

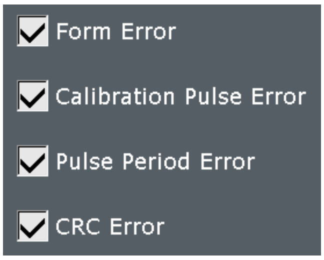
Data min ← Serial Message setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
TRIGger<m>:SENT:SDMN on page 1672

Data max ← Serial Message setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:SENT:SDMX on page 1672

Error conditions setup



Executes a trigger event if one or more of the following errors occur:

- "Form Error"
  Detects a form error in serial messages (short and enhanced).
  An error occurs when at least one of the transmission sequences that form a serial message has an error.
- "Calibration Pulse Error"
  Detects a calibration pulse error in transmission sequences.
  An error occurs when:
  – the duration of the "Calibration/Sync" pulse (in ticks) is less than 56*(1-clock tolerance) or more than 56*(1+clock tolerance)
  – the "Calibration/Sync" pulse duration of frame (–1) varies by more than 1.5625% from the "Calibration/Sync" pulse duration of frame (n)
- "Pulse Period Error"
  Detects an error in the "Calibration/Sync" pulse in transmission sequences.
  An error occurs when a nibble has any of the following:
  – Number of ticks at low is fewer than 4 ticks.
  – Nibble value < 0 (fewer than 12 ticks) or > 15 (more than 27 ticks).
- "CRC Error"
  Detects a checksum error in both, the transmission sequences and serial messages.
  The CRC length is 4 bits for transmission sequences and short serial messages, and 6 bit of enhanced serial messages.
- "Irregular Frame Length Error"
  Detects frame length errors in transmission sequences when pause pulse mode constant frame length is set, see "Pause Pulse" on page 703.
  A frame length error occurs, when the total length of the transmission sequence (including pause pulse) does not match the frame length setting, see "Frame Length in clock ticks" on page 703.

Remote command:
TRIGger<m>:SENT:FORMerror on page 1672
TRIGger<m>:SENT:PULSeerror on page 1673
TRIGger<m>:SENT:PPERioderror on page 1673
TRIGger<m>:SENT:CRCerror on page 1673
TRIGger<m>:SENT:IRFLength on page 1674

12.13.3.2 Triggering on SENT Signals

Prerequisites: A SENT bus is configured, see Chapter 12.13.2.2, "Configuring SENT Protocol", on page 703.

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to SENT.
5. Select the "Trigger type".
6. For more complex trigger types, set the parameters according to the selected "Trigger type".
   For details, see Chapter 12.13.3.1, "SENT Trigger Settings", on page 704.

12.13.4 SENT Label List

SENT label lists provide a very useful way of translating the decoded data into user format. The label lists are highly customizable. The format of supplying the label list description is through a `.xml` file and is explained with an example, see "Label List Structure for SENT Protocol" on page 711.

For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Label List Structure for SENT Protocol

```
<sb:FRAME NAME="Diagnostic Error Codes" STATE="ON">
  <!-- Start of a Frame Definition -->
  <!-- This block defines the information of a Transmission Sequence
   or Serial Message:
   NAME => Symbolic Label of the Frame
   STATE [ON/OFF] => When ON, this frame Translation is taken into consideration.
                     When OFF, this frame Translation is skipped.-->
<sb:DESCRIPTION> used to diagnose the current SENT System</sb:DESCRIPTION>
  <!-- Doesn't affect the Translation -->
<sb:ID-VALUE>01</sb:ID-VALUE>
  <!-- ID Value of the Serial Message (in decimal) -->
  <!-- Absence of the ID-VALUE field implies that the current Frame Translation
   is to be used for Transmission Sequences and not for a Serial Message -->
<sb:ID-LENGTH>8</sb:ID-LENGTH>
  <!-- ID Length of the Serial Message (in bits) -->
<sb:DATA-SIZE>12</sb:DATA-SIZE>
  <!-- Data Length of the Serial Message (in bits) -->
<sb:SIGNALS>
  <!-- This block defines the information of the Signals embedded
   in the Data Field of the Frame (Transmission Sequence or Serial Message) -->
<sb:SIGNAL ID="Diagnostic">
  <!-- Unique ID of the Signal (no effect on Translation) -->
<sb:SHORT-NAME>Diagnostic Code</sb:SHORT-NAME>
  <!-- Name of the Signal -->
```

```
<sb:DESCRIPTION></sb:DESCRIPTION>
  <!-- Info Field (no effect on Translation) -->
<sb:BIT-POSITION>11</sb:BIT-POSITION>
  <!-- Starting Bit position of the Signal
   (The whole Data Field is represented as MSB -> LSB Sequence) -->
<sb:BIT-LENGTH>12</sb:BIT-LENGTH>
  <!-- Number of Bits representing the Signal Value -->
<sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
  <!-- Byte Order of the Signal Value [MSB or LSB], Default: MSB -->
<sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
  <!-- Representation of the Bits [ENUM, UNSIGNED_INT, INT, FLOAT, DOUBLE],
   Default: UNSIGNED_INT
   The Signal Value is calculated according to the following:
   Translated_Value = Encoded_Value * FACTOR + OFFSET -->
<sb:FACTOR>1.0</sb:FACTOR>
  <!-- Signal Factor (decimal value)-->
<sb:OFFSET>0.0</sb:OFFSET>
  <!-- Signal Offset (decimal value)-->
<sb:MIN>0</sb:MIN>
  <!-- Minimum Signal Value (decimal value) -->
<sb:MAX>4096</sb:MAX>
  <!-- Maximum Signal Value (decimal value) -->
<sb:ENUM-VALUES>
  <!-- This block is only valid (and taken into consideration)
   when the VALUE-TYPE is ENUM
   It defines the Enumeration List Translation of the Signal -->
<sb:ENUM INDEX="0" LABEL="No Error"/>
   <!-- INDEX is the Enum Value (corresponds to the Signal Value in decimal),
    LABEL is the matching Translated Signal Value -->
<sb:ENUM INDEX="1" LABEL="Channel 1 out of range high"/>
</sb:ENUM-VALUES>
  <!-- End of Signal Enumeration List Definition -->
</sb:SIGNAL>
  <!-- End of a Signal Definition -->
  <!-- More Signals can be defined here! -->
</sb:SIGNALS>
  <!-- End of list of Signals Definition -->
</sb:FRAME>
  <!-- End of Frame Definition -->
```

For an example to label list translation, see Chapter 12.13.4.1, "SENT Label List Translation Example", on page 714.

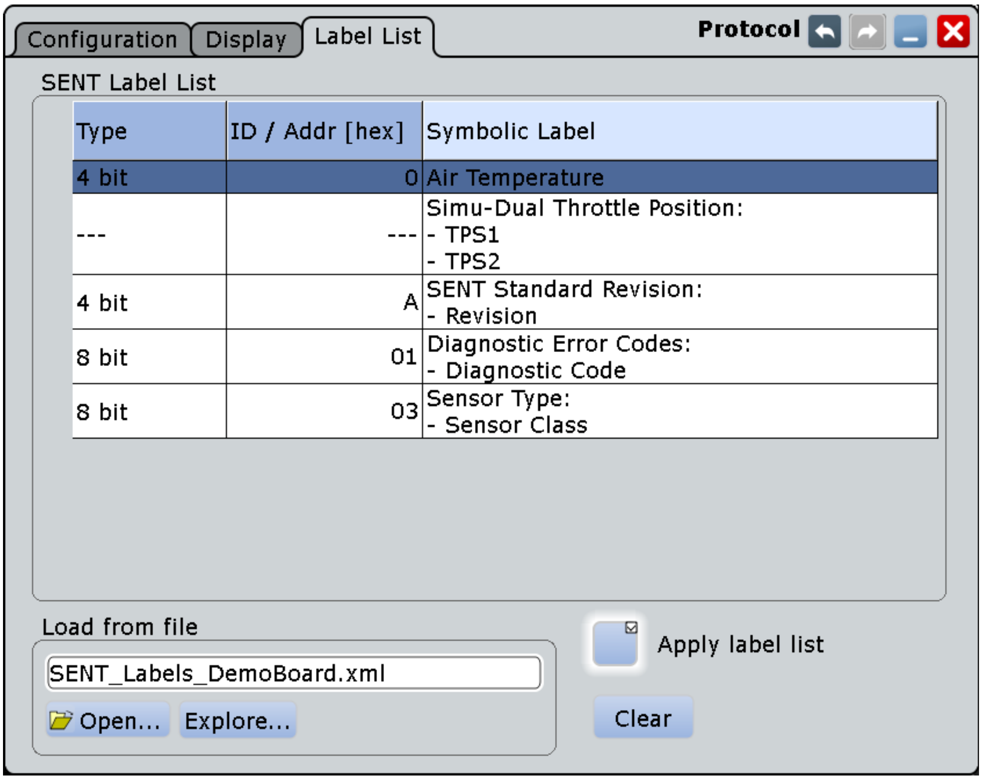


Figure 12-69: SENT label list

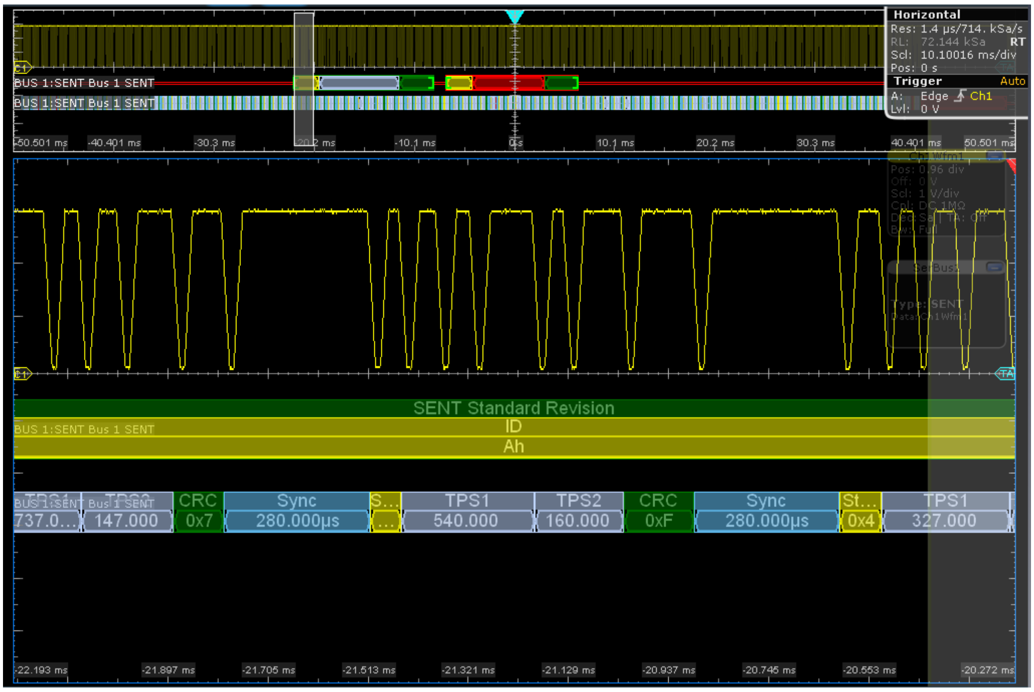


Figure 12-70: SENT decode results with label list translation

Remote command:

- BUS<m>:SENT:FRAMe<n>:SYMBol? on page 1679
- BUS<m>:SENT:FRAMe<n>:SDATa? on page 1679
- BUS<m>:SENT:FRAMe<n>:SDEXport? on page 1679

12.13.4.1 SENT Label List Translation Example

The example shows the xml sequence for a label list translation in the SENT protocol:

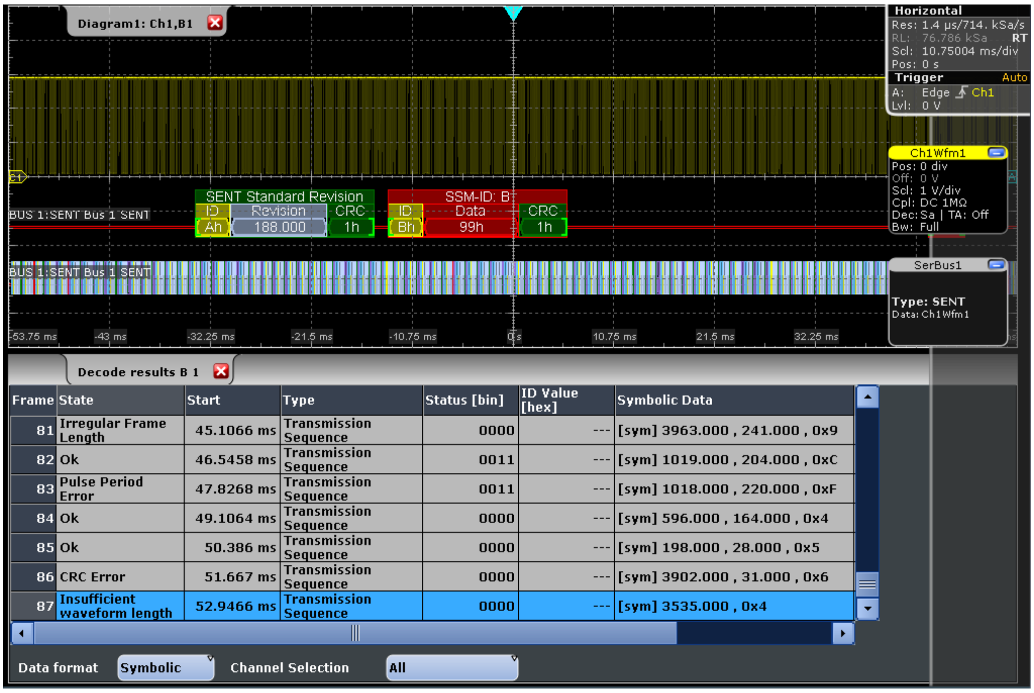


Figure 12-71: SENT label list translation

```
<?xml version="1.0" encoding="UTF-8"?>
<sb:LABEL-LIST-FILE>
 <sb:PROJECT ID="SENT-TRANSLATION SYSTEM">
  <sb:SHORT-NAME>SENT</sb:SHORT-NAME>
  <sb:LONG-NAME>SENT-Translation System Demo</sb:LONG-NAME>
  <sb:DESCRIPTION>This is the database for Translation demo for SENT.</sb:DESCRIPTION>
 </sb:PROJECT>
 <sb:FRAMES>
  <sb:FRAME NAME="Air Temperature" STATE="ON">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>0</sb:ID-VALUE>
   <sb:ID-LENGTH>4</sb:ID-LENGTH>
   <sb:DATA-SIZE>16</sb:DATA-SIZE>
  </sb:FRAME>
  <sb:FRAME NAME="Humidity" STATE="OFF">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>2</sb:ID-VALUE>
   <sb:ID-LENGTH>4</sb:ID-LENGTH>
```

```
 <sb:DATA-SIZE>16</sb:DATA-SIZE>
</sb:FRAME>
<sb:FRAME NAME="Barometric Pressure" STATE="OFF">
 <sb:DESCRIPTION></sb:DESCRIPTION>
 <sb:ID-VALUE>4</sb:ID-VALUE>
 <sb:ID-LENGTH>4</sb:ID-LENGTH>
 <sb:DATA-SIZE>16</sb:DATA-SIZE>
</sb:FRAME>
<sb:FRAME NAME="Configuration Code" STATE="OFF">
 <sb:DESCRIPTION></sb:DESCRIPTION>
 <sb:ID-VALUE>04</sb:ID-VALUE>
 <sb:ID-LENGTH>8</sb:ID-LENGTH>
 <sb:DATA-SIZE>12</sb:DATA-SIZE>
</sb:FRAME>
<sb:FRAME NAME="Manufacturer Code" STATE="OFF">
 <sb:DESCRIPTION></sb:DESCRIPTION>
 <sb:ID-VALUE>05</sb:ID-VALUE>
 <sb:ID-LENGTH>8</sb:ID-LENGTH>
 <sb:DATA-SIZE>12</sb:DATA-SIZE>
</sb:FRAME>
<sb:FRAME NAME="Sensor Type" STATE="ON">
 <sb:DESCRIPTION>specifies the SENT Sensor Type</sb:DESCRIPTION>
 <sb:ID-VALUE>03</sb:ID-VALUE>
 <sb:ID-LENGTH>8</sb:ID-LENGTH>
 <sb:DATA-SIZE>12</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Sensor Class">
   <sb:SHORT-NAME>Sensor Class</sb:SHORT-NAME>
   <sb:BIT-POSITION>11</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>32.0</sb:MAX>
   <sb:ENUM-VALUES>
    <sb:ENUM INDEX="0" LABEL="Not Specified"/>
    <sb:ENUM INDEX="1" LABEL="P"/>
    <sb:ENUM INDEX="2" LABEL="P/-"/>
    <sb:ENUM INDEX="3" LABEL="P/S"/>
    <sb:ENUM INDEX="4" LABEL="P/S/Default T"/>
    <sb:ENUM INDEX="5" LABEL="P/S/Sensor-Specific T"/>
    <sb:ENUM INDEX="6" LABEL="P1/P2"/>
    <sb:ENUM INDEX="7" LABEL="P/Default T"/>
    <sb:ENUM INDEX="8" LABEL="P/Sensor-Specific T"/>
    <sb:ENUM INDEX="9" LABEL="P1/P2/Default T"/>
    <sb:ENUM INDEX="10" LABEL="P1/P2/Sensor-Specific T"/>
    <sb:ENUM INDEX="16" LABEL="Not Defined"/>
    <sb:ENUM INDEX="17" LABEL="MAF (hi-res,lin)"/>
```

```
    <sb:ENUM INDEX="18" LABEL="MAF (hi-res,non-lin)"/>
    <sb:ENUM INDEX="19" LABEL="MAF (hi-res,lin) / Pressure"/>
    <sb:ENUM INDEX="20" LABEL="MAF (hi-res,non-lin) / Pressure"/>
    <sb:ENUM INDEX="21" LABEL="MAF (lin) / Pressure (hi-res)"/>
    <sb:ENUM INDEX="22" LABEL="MAF (non-lin) / Pressure (hi-res)"/>
   </sb:ENUM-VALUES>
  </sb:SIGNAL>
 </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="SENT Standard Revision" STATE="ON">
 <sb:SHORT-NAME>SENT Standard</sb:SHORT-NAME>
 <sb:DESCRIPTION>specifies the SENT Standard Revision Number</sb:DESCRIPTION>
 <sb:ID-VALUE>10</sb:ID-VALUE>
 <sb:ID-LENGTH>4</sb:ID-LENGTH>
 <sb:DATA-SIZE>8</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Revision">
   <sb:SHORT-NAME>Revision</sb:SHORT-NAME>
   <sb:DESCRIPTION>SENT-Standard Revision Number</sb:DESCRIPTION>
   <sb:BIT-POSITION>7</sb:BIT-POSITION>
   <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>4.0</sb:MAX>
   <sb:ENUM-VALUES>
    <sb:ENUM INDEX="0" LABEL="Not defined"/>
    <sb:ENUM INDEX="1" LABEL="J2716 Rev 1"/>
    <sb:ENUM INDEX="2" LABEL="J2716 Rev 2"/>
    <sb:ENUM INDEX="3" LABEL="J2716 Rev 3"/>
   </sb:ENUM-VALUES>
  </sb:SIGNAL>
 </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Diagnostic Error Codes" STATE="ON">
 <sb:DESCRIPTION>used to diagnose the current SENT System</sb:DESCRIPTION>
 <sb:ID-VALUE>01</sb:ID-VALUE>
 <sb:ID-LENGTH>8</sb:ID-LENGTH>
 <sb:DATA-SIZE>12</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Diagnostic">
   <sb:SHORT-NAME>Diagnostic Code</sb:SHORT-NAME>
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:BIT-POSITION>11</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
   <sb:FACTOR>1.0</sb:FACTOR>
```

```
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>4096</sb:MAX>
      <sb:ENUM-VALUES>
       <sb:ENUM INDEX="0" LABEL="No Error"/>
       <sb:ENUM INDEX="1" LABEL="Channel 1 out of range high"/>
       <sb:ENUM INDEX="2" LABEL="Channel 1 out of range low"/>
       <sb:ENUM INDEX="3" LABEL="Initialization Error (Channel 1)"/>
       <sb:ENUM INDEX="4" LABEL="Channel 2 out of range high"/>
       <sb:ENUM INDEX="5" LABEL="Channel 2 out of range low"/>
       <sb:ENUM INDEX="6" LABEL="Initialization Error (Channel 2)"/>
       <sb:ENUM INDEX="7" LABEL="Channel 1 and 2 Rationality Error"/>
       <sb:ENUM INDEX="1025" LABEL="Slow Channel Temperature out of range high"/>
       <sb:ENUM INDEX="1026" LABEL="Slow Channel Temperature out of range low"/>
       <sb:ENUM INDEX="1027" LABEL="Slow Channel Temperature initialization error"/>
       <sb:ENUM INDEX="1028" LABEL="Slow Channel Humidity out of range high"/>
       <sb:ENUM INDEX="1029" LABEL="Slow Channel Humidity out of range low"/>
       <sb:ENUM INDEX="1030" LABEL="Slow Channel Humidity initialization error"/>
       <sb:ENUM INDEX="1031" LABEL="Slow Channel Barometric Pressure out of range high"/>
       <sb:ENUM INDEX="1032" LABEL="Slow Channel Barometric Pressure out of range low"/>
       <sb:ENUM INDEX="1033" LABEL="Slow Channel Barometric Pressure initialization error"/>
      </sb:ENUM-VALUES>
     </sb:SIGNAL>
    </sb:SIGNALS>
   </sb:FRAME>
   <sb:FRAME NAME="Simu-Dual Throttle Position" STATE="ON">
    <sb:SHORT-NAME>DTP</sb:SHORT-NAME>
    <sb:DATA-SIZE>20</sb:DATA-SIZE>
    <sb:SIGNALS>
     <sb:SIGNAL ID="Channel_1">
      <sb:SHORT-NAME>TPS1</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>19</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:VALUE-TYPE>UNSIGNED_INT</sb:VALUE-TYPE>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
     </sb:SIGNAL>
     <sb:SIGNAL ID="Channel_2">
      <sb:SHORT-NAME>TPS2</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>7</sb:BIT-POSITION>
      <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
```

```
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
  </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Dual Throttle Position" STATE="OFF">
  <sb:SHORT-NAME>DTP</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
    <sb:SIGNAL ID="Channel_1">
      <sb:SHORT-NAME>TPS1</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>23</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:VALUE-TYPE>UNSIGNED_INT</sb:VALUE-TYPE>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
    <sb:SIGNAL ID="Channel_2">
      <sb:SHORT-NAME>TPS2</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>11</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
  </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Mass Air Flow (16)" STATE="OFF">
  <sb:SHORT-NAME>MAF/P</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
    <sb:SIGNAL ID="Channel_1">
      <sb:SHORT-NAME>MAF</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>23</sb:BIT-POSITION>
      <sb:BIT-LENGTH>16</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
```

```
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Mass Air Flow (16/8)" STATE="OFF">
  <sb:SHORT-NAME>MAF/P</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>MAF</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>16</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
   <sb:SIGNAL ID="Channel_2">
    <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>7</sb:BIT-POSITION>
    <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Mass Air Flow (14/10)" STATE="OFF">
  <sb:SHORT-NAME>MAF/P</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>MAF</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>14</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
```

```
    </sb:SIGNAL>
    <sb:SIGNAL ID="Channel_2">
     <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
     <sb:DESCRIPTION>""</sb:DESCRIPTION>
     <sb:BIT-POSITION>9</sb:BIT-POSITION>
     <sb:BIT-LENGTH>10</sb:BIT-LENGTH>
     <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
     <sb:FACTOR>1.0</sb:FACTOR>
     <sb:OFFSET>0.0</sb:OFFSET>
     <sb:MIN>0</sb:MIN>
     <sb:MAX>10000.0</sb:MAX>
     <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
   </sb:SIGNALS>
  </sb:FRAME>
  <sb:FRAME NAME="Single Secure Sensor" STATE="OFF">
   <sb:SHORT-NAME>SSS</sb:SHORT-NAME>
   <sb:DATA-SIZE>24</sb:DATA-SIZE>
   <sb:SIGNALS>
    <sb:SIGNAL ID="Channel_1">
     <sb:SHORT-NAME>Ch1</sb:SHORT-NAME>
     <sb:DESCRIPTION>""</sb:DESCRIPTION>
     <sb:BIT-POSITION>23</sb:BIT-POSITION>
     <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
     <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
     <sb:FACTOR>1.0</sb:FACTOR>
     <sb:OFFSET>0.0</sb:OFFSET>
     <sb:MIN>0</sb:MIN>
     <sb:MAX>10000.0</sb:MAX>
     <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
    <sb:SIGNAL ID="Channel_2">
     <sb:SHORT-NAME>Counter</sb:SHORT-NAME>
     <sb:DESCRIPTION>""</sb:DESCRIPTION>
     <sb:BIT-POSITION>11</sb:BIT-POSITION>
     <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
     <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
     <sb:FACTOR>1.0</sb:FACTOR>
     <sb:OFFSET>0.0</sb:OFFSET>
     <sb:MIN>0</sb:MIN>
     <sb:MAX>256.0</sb:MAX>
     <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
   </sb:SIGNALS>
  </sb:FRAME>
  <sb:FRAME NAME="Pressure Sensor" STATE="OFF">
   <sb:SHORT-NAME>P</sb:SHORT-NAME>
   <sb:DATA-SIZE>24</sb:DATA-SIZE>
   <sb:SIGNALS>
    <sb:SIGNAL ID="Channel_1">
```

```
      <sb:SHORT-NAME>Pressure1</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>23</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
    <sb:SIGNAL ID="Channel_2">
      <sb:SHORT-NAME>Pressure2</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>11</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
  </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Pressure and Temperature Sensor" STATE="OFF">
  <sb:SHORT-NAME>P/T</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
    <sb:SIGNAL ID="Channel_1">
      <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>23</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
    </sb:SIGNAL>
    <sb:SIGNAL ID="Channel_2">
      <sb:SHORT-NAME>Temperature</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>11</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
```

```
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
     </sb:SIGNAL>
    </sb:SIGNALS>
   </sb:FRAME>
   <sb:FRAME NAME="Pressure and Secure Sensor" STATE="OFF">
    <sb:SHORT-NAME>P/S</sb:SHORT-NAME>
    <sb:DATA-SIZE>24</sb:DATA-SIZE>
    <sb:SIGNALS>
     <sb:SIGNAL ID="Channel_1">
      <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>23</sb:BIT-POSITION>
      <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
     </sb:SIGNAL>
     <sb:SIGNAL ID="Channel_2">
      <sb:SHORT-NAME>Counter</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>11</sb:BIT-POSITION>
      <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
     </sb:SIGNAL>
    </sb:SIGNALS>
   </sb:FRAME>
  </sb:FRAMES>
</sb:LABEL-LIST-FILE>
```


12.13.5 SENT Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Data is decoded and displayed in the order of its reception. The "Decode results" box shows the detailed decoded data for each frame as it is received.

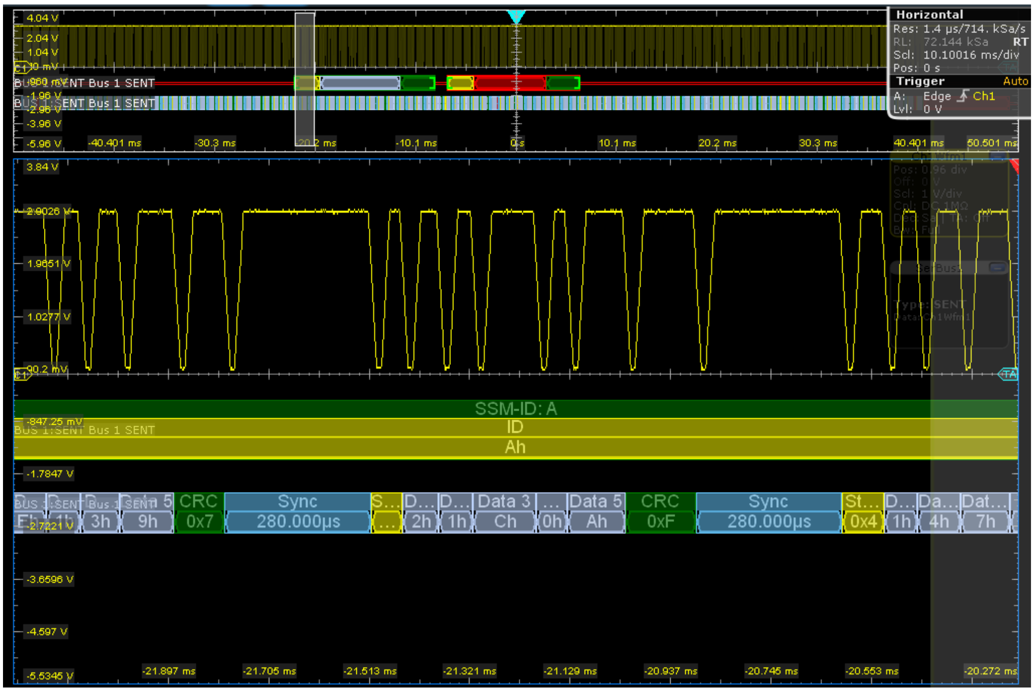


Figure 12-72: SENT decode results display

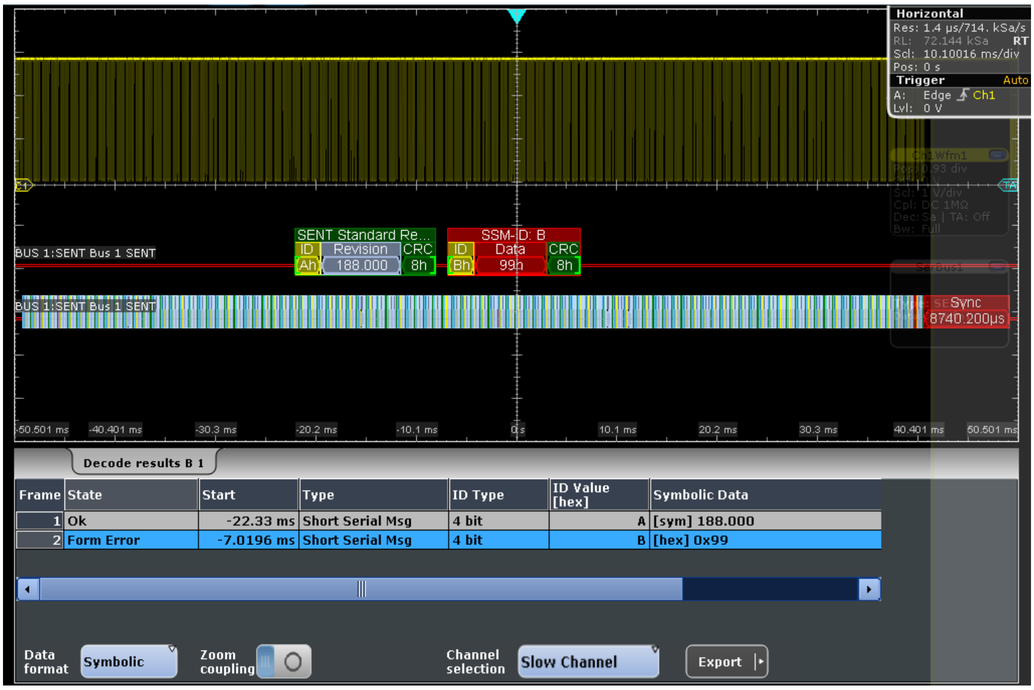


Figure 12-73: SENT decode results of a short serial message

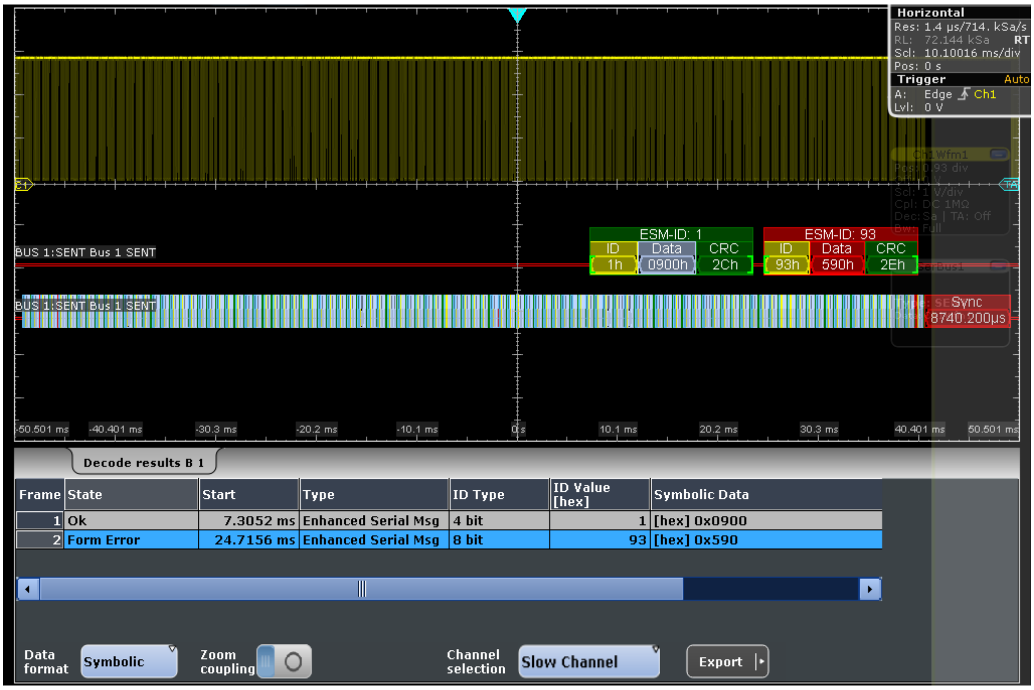


Figure 12-74: SENT decode results of an enhanced serial message

Table 12-16: Content of the decode result table

Column	Description
State	Overall state of the frame
Start	Time of frame start in relation to the trigger point
Type	Frame type
Sync Duration	Time of the synchronization pulse
Status [bin]	Status value
ID Type	Identifier type
ID Value [hex]	Identifier value
Data Nibbles	Value of the data nibble
Symbolic Data	Symbolic data value
Label	Symbolic label name defined in the label list
CRC [hex]	CRC sequence value
Pause Pulse [Ticks]	Number of the pulse pause clock
Bit rate	Value of the bit rate

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.14.3, "Decode Results", on page 1674.

12.13.6 Search on Decoded SENT Data

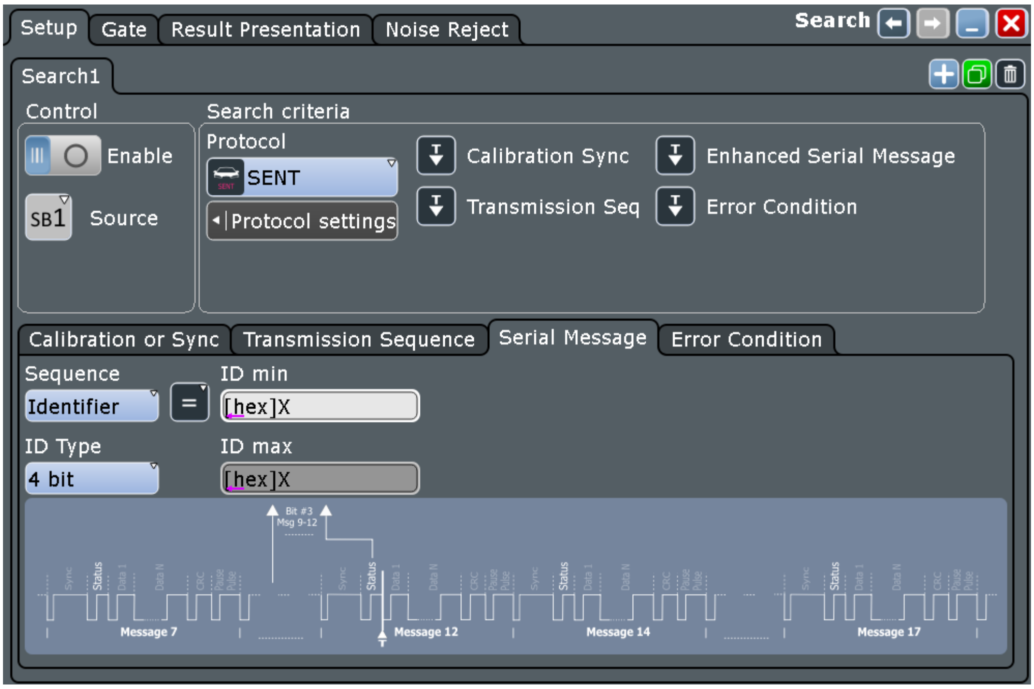
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.13.6.1 SENT Search Setup

Access: [SEARCH] > "Setup" tab



Search criteria
Enable the events to be searched for.

Unlike triggering, where you can trigger only on one defined event, you can search for various different events in one search.

"Calibration Sync"
Searches for the end of the "Calibration/Synchronization" sequence, i.e. the falling edge.
This setting does not require any input parameters.

"Transmission Seq"

Searches for the end of the status nibble in a single transmission sequence, or the end of the combination of the status and data nibble(s).
Description of the specific settings: "Transmission Sequence setup" on page 727

"Serial Message"

Searching on a serial message depends on the serial protocol selected with Serial Protocol > Short | Enhanced and the associated setting parameters:

- "Sequence > Identifier": searches for the end of the identifier nibble.
- "Sequence > ID+Data": searches for the end of the "ID and Data" nibble.

Description of the serial messages specific settings: "Serial Message setup" on page 729

"Error Condition"

Searches for the end of certain error events.
Description of trigger type specific settings: "Error conditions setup" on page 731

Remote command:
`SEARch:TRIGger:SENT:CALibration` on page 1681
`SEARch:TRIGger:SENT:TRANsmission` on page 1682
`SEARch:TRIGger:SENT:SMSG` on page 1682
`SEARch:TRIGger:SENT:ERRor` on page 1682

Transmission Sequence setup

Configures the search conditions for the transmission sequence.

Note: The displayed parameters depend on the selected "Sequence". The instrument displays the data setting parameters when you select "Status+Data", see "Sequence" on page 728.
The search type specific conditions are the same as for the trigger type, see "Transmission Sequence setup" on page 706.

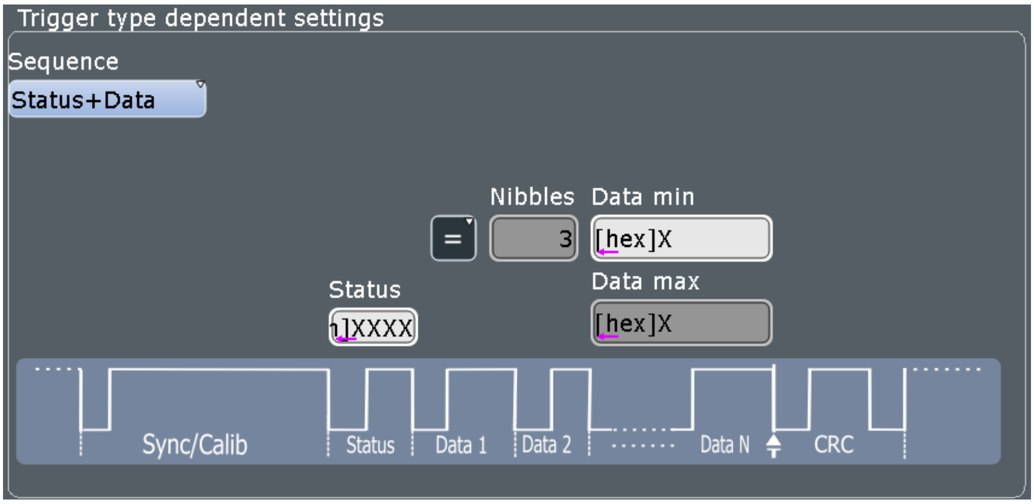


Figure 12-75: Search setting parameters of a transmission sequence

Sequence ← Transmission Sequence setup
Selects the condition for searching in a single transmission sequence.

You can search for the end of the status nibble or the combination of the status and data nibble(s).

Remote command:
SEARch:TRIGger:SENT:TTYPe on page 1683

Status ← Transmission Sequence setup
Defines the data bits for the status nibble.

Remote command:
SEARch:TRIGger:SENT:STATus on page 1683

Condition ← Transmission Sequence setup
Selects the operator to define a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
SEARch:TRIGger:SENT:TDCN on page 1684

Data Nibbles ← Transmission Sequence setup
Sets the number of data units in a single transmission sequence.

The maximum number of data nibbles is 6.

Remote command:
BUS<m>:SENT:DNIBbles on page 1666

Data min ← Transmission Sequence setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
`SEARch:TRIGger:SENT:TDMN` on page 1684

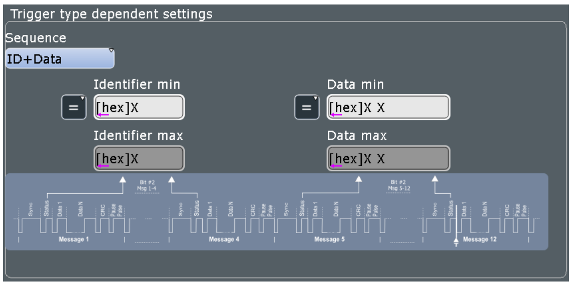
Data max ← Transmission Sequence setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SENT:TDMX` on page 1684

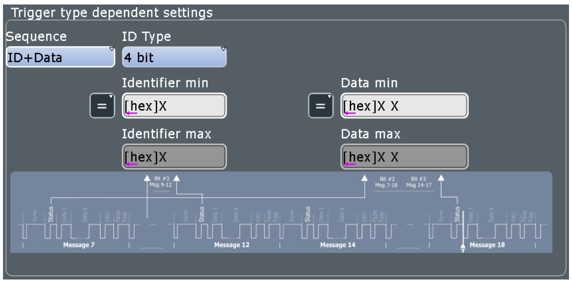
Serial Message setup
Configures the search conditions for a serial message.

Note: The displayed parameters depend on the selected "Sequence". The instrument indicates the data setting parameters when you select "ID+Data", see "Sequence" on page 729.
The description of the search type specific settings are the same as for the trigger type, see "Serial Message setup" on page 708.

Search setting parameters of the serial message types



Short Serial Message



Enhanced Serial Message

Sequence ← Serial Message setup
Selects the condition for searching in a serial message.

You can search for the end of an identifier nibble or the combination of the identifier and data nibble(s).

Remote command:
`SEARch:TRIGger:SENT:STYPe` on page 1685

ID Type ← Serial Message setup
Selects the message ID format for the enhanced serial message type.

You can select either 4 bit or 8 bit message ID.

Remote command:
`SEARch:TRIGger:SENT:SIDType` on page 1685

Identifier Condition ← Serial Message setup
Selects the operator to set a specific identifier or an identifier range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
SEARch:TRIGger:SENT:SICN on page 1685

Identifier min ← Serial Message setup
Defines the bit pattern of the message identifier. In binary format, use the following characters: 1; 0; or X (any bit).

The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

Remote command:
SEARch:TRIGger:SENT:SIMN on page 1686

Identifier max ← Serial Message setup
The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:SENT:SIMX on page 1686

SSM Data Condition ← Serial Message setup
Selects the operator to set a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
SEARch:TRIGger:SENT:SDCN on page 1686

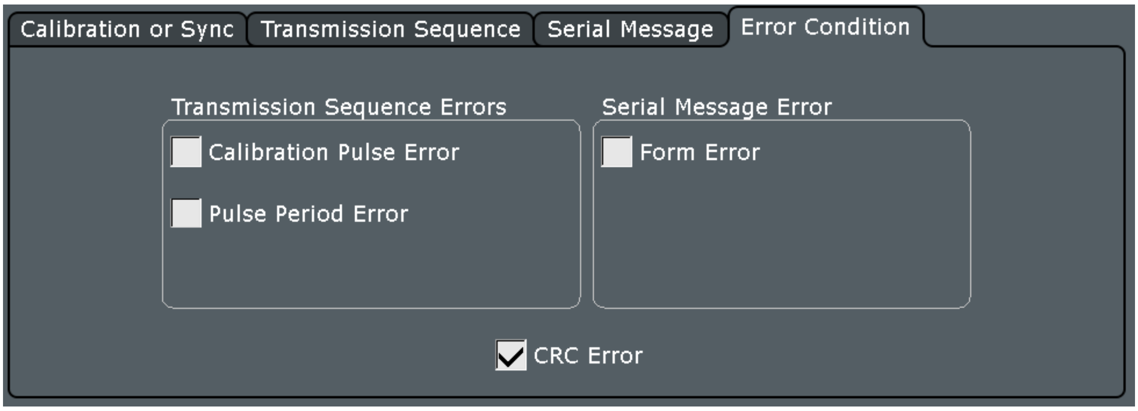
Data min ← Serial Message setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
SEARch:TRIGger:SENT:SDMN on page 1687

Data max ← Serial Message setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:SENT:SDMX on page 1687

Error conditions setup



Performs the search on one or more of the following error events:

- "Transmission sequence errors"
  – "Calibration Pulse Error"
    Searches for calibration pulse errors.
  – "Pulse Period Error"
    Searches for pulse period errors in a transmission sequence.
  – "Irregular Frame Length Error"
    Searches for irregular frame length errors in a transmission sequence if pause pulse mode is set to constant frame length..
- Serial message error
  – "Form Error"
    Searches for format errors in serial messages.
- "CRC Error"
  Searches for errors in the complete data transmission.

Remote command:
`SEARch:TRIGger:SENT:PULSeerror` on page 1687
`SEARch:TRIGger:SENT:PPERioderror` on page 1688
`SEARch:TRIGger:SENT:FORMerror` on page 1688
`SEARch:TRIGger:SENT:CRCerror` on page 1688
`SEARch:TRIGger:SENT:IRFLength` on page 1688

12.13.6.2 SENT Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

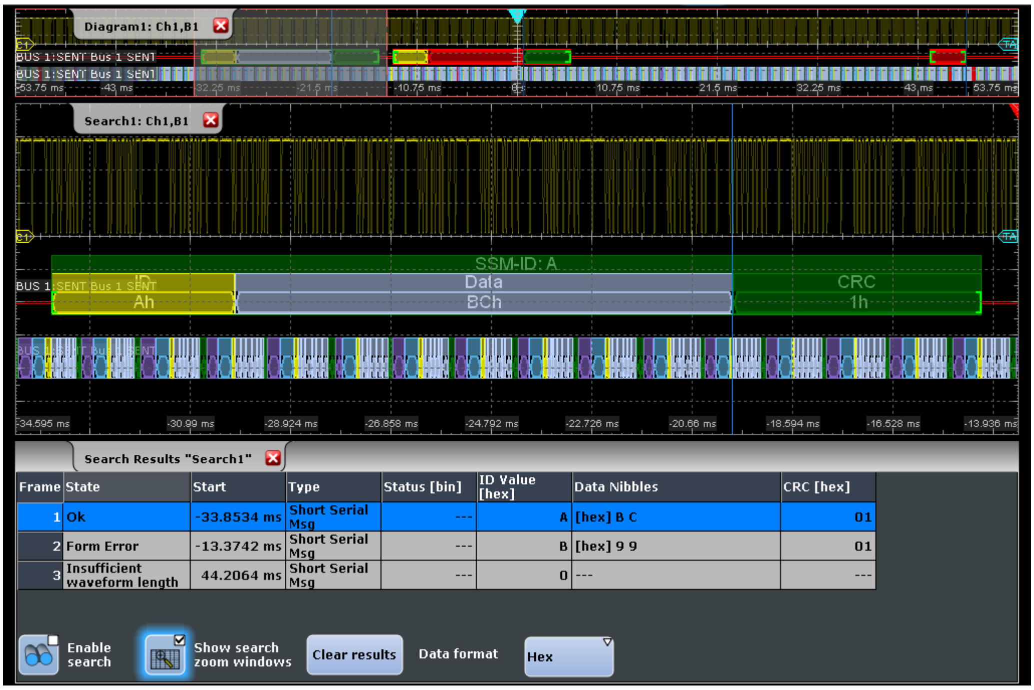


Figure 12-76: Search on the Status nibble in a SENT transmission signal

The columns in the search result table are the same as in the decoding table, see Chapter 12.13.5, "SENT Decode Results", on page 722.

Remote commands:

- SEARch:RESult:SENT:FCOunt? on page 1689
- SEARch:RESult:SENT:FRAMe<m>:STATus? on page 1693
- SEARch:RESult:SENT:FRAMe<m>:STARt? on page 1692
- SEARch:RESult:SENT:FRAMe<m>:STOP? on page 1693
- SEARch:RESult:SENT:FRAMe<m>:DATA? on page 1690
- SEARch:RESult:SENT:FRAMe<m>:CSValue? on page 1690
- SEARch:RESult:SENT:FRAMe<m>:IDTYpe? on page 1690
- SEARch:RESult:SENT:FRAMe<m>:IDValue? on page 1690
- SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe? on page 1691
- SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue? on page 1691
- SEARch:RESult:SENT:FRAMe<m>:PAPTicks? on page 1691
- SEARch:RESult:SENT:FRAMe<m>:SCOM? on page 1692
- SEARch:RESult:SENT:FRAMe<m>:SDATa? on page 1692
- SEARch:RESult:SENT:FRAMe<m>:SYMBol? on page 1693
- SEARch:RESult:SENT:FRAMe<m>:SYNCduration? on page 1693
- SEARch:RESult:SENT:FRAMe<m>:TYPE? on page 1694

12.14 Custom: Manchester / NRZ (Option R&S RTE-K50)

R&S RTE-K50 is a firmware option that enables the R&S RTE to analyze customizable serial bus signals encoded by the following coding standards:

- Manchester
- Manchester II
- NRZ Clocked
- NRZ Unclocked

For analysis, signals encoded in any of these protocols can be triggered and decoded.

Due to the free format description, no search within the decoded events is available.

This chapter describes:

- Custom: Manchester / NRZ Protocols....................733
- Custom: Manchester / NRZ Configuration....................735
- Custom: Manchester / NRZ Trigger....................755
- Custom Filter....................760
- Custom: Manchester / NRZ Decode Results....................762
- Search on Decoded Custom Manchester / NRZ Data....................765

12.14.1 Custom: Manchester / NRZ Protocols

"Manchester" coding is a self-clocked coding scheme also known as phase-shift keying (or phase encoding, PE). It is used in protocols such as ProfiBus (IEC 61158), DALI (Digital Addressable Lighting Interface, IEC 60929 and IEC 62386), MVB (Multifunction Vehicle Bus, part of IEC 61375 for Train Communication Networks, TCN), and Ethernet 10BASE-T (10 Mbit/s, IEEE 802.3i). In terms of a logical Boolean operation, the Manchester value of each bit (as per G. E. Thomas) is the exclusive disjunction (XOR) of the original data value and the clock value. A "0" is expressed by a high-to-low transition, a "1" by a low-to-high transition. These transitions, which occur at the middle of each bit period, make the signal self-clocked.

"Manchester II" coding (as per IEEE 802.3) is represented by inverted Manchester values: a "0" is expressed by a low-to-high transition, a "1" by a high-to-low transition.

NRZ stands for "non-return-to-zero" coding: Typically a "1" is represented by a positive voltage and a "0" is represented by a negative voltage, with no "zero" voltage state. NRZ code requires only half the bandwidth of Manchester code, and it can either be clocked or unclocked. NRZ unclocked signals require a user-defined bit rate and gap time setting for triggering and decoding.

12.14.1.1 Special Features of Manchester Coding

In practical protocols, Manchester coding appears in many variations, often employing deliberate coding violations to encode special waveform features, such as unambiguous synchronization and termination patterns. To adapt to these specific Manchester implementations and handle ambiguous signals, the option R&S RTE-K50 for Custom

Serial Bus uses a combination of automatic algorithms and user configurable parameters.

Quaternary Symbols

The software supports not just traditional binary symbols "0" and "1", but also arbitrary violation waveforms that use two additional symbols, yielding a total of four valid "quaternary bit" values. The two additional violation symbols are "H" (high) and "L" (low). Values of "H" correspond to a waveform lacking a transition in the center of the bit, with a physical high voltage state. Similarly, "L" violations also lack a center transition, but have a physical low voltage state. Most Manchester synchronization and termination conventions, even those containing violations, may be expressed as sequences of these four symbols. R&S RTE-K50 uses the quaternary notation to support Manchester patterns in the honeycomb display and to describe synchronization and termination patterns in the frame description table.

Idle Conditions

The state of the signal line in between messages is the idle condition. Manchester appears in practical standards with varying idle conditions: it can idle at the high, low, or middle voltage state. High and low idle states correspond to "biphase" Manchester, while the middle voltage (often ground) adds a third state to become "ternary" Manchester. Using ternary Manchester, option R&S RTE-K50 can usually establish the gaps between messages automatically. Using binary Manchester, the software has no way to automatically discriminate an idling bus from monotonic sequences of "H" or "L" violations. For these biphase situations, R&S RTE-K50 offers a "Gap Time" detection feature, which allows to distinguish long intervals of non-transitions between bus idling and sequences of violations. Other differences between biphase and ternary Manchester are managed automatically by the software, with no user input required.

Edge Conventions

Most Manchester encodings establish the beginning of the first bit by a first transition, hence an "overhead" edge. The center of the bit is then marked by a second transition, which is a "sampling" edge. Some Manchester implementations, however, sample the first bit on the first edge. The option R&S RTE-K50 attempts to automatically detect this situation. Unfortunately, it is possible to trick the algorithm with waveforms that contain many (legitimate) violations. In these situations, the user can force a "First Edge" or "Second Edge" convention for handling edges. Edge sampling according to the "First Edge" convention is more likely to appear in biphase Manchester, but the software also supports this setting for ternary Manchester situations.

Bit Rate

Typically, a single bit rate is clearly specified in Manchester protocols; however, some implementations use a variable bit rate. By default, R&S RTE-K50 automatically determines the bit rate with no user input required. However, there are fundamental ambiguities possible in Manchester, if the bitrate is unknown. In particular, sequences like "0000", "1111", "0101", "1010", and many situations involving "H" and "L" violations, cannot be decoded without a known bit rate. The situation becomes even less defined with eventual Manchester coding violations. In these situations, a fixed "Bit Rate" setting has to be provided by the user to bypass the software's estimation algorithm.

12.14.2 Custom: Manchester / NRZ Configuration

If you need information on how to get started with configuring the custom serial bus setup, see Chapter 12.14.2.5, "Configuring Custom Manchester / NRZ Signals", on page 754. Otherwise proceed with the configuration settings.

12.14.2.1 Custom: Manchester / NRZ Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *Custom*

Make sure that the tab of the correct serial bus is selected on the left side.

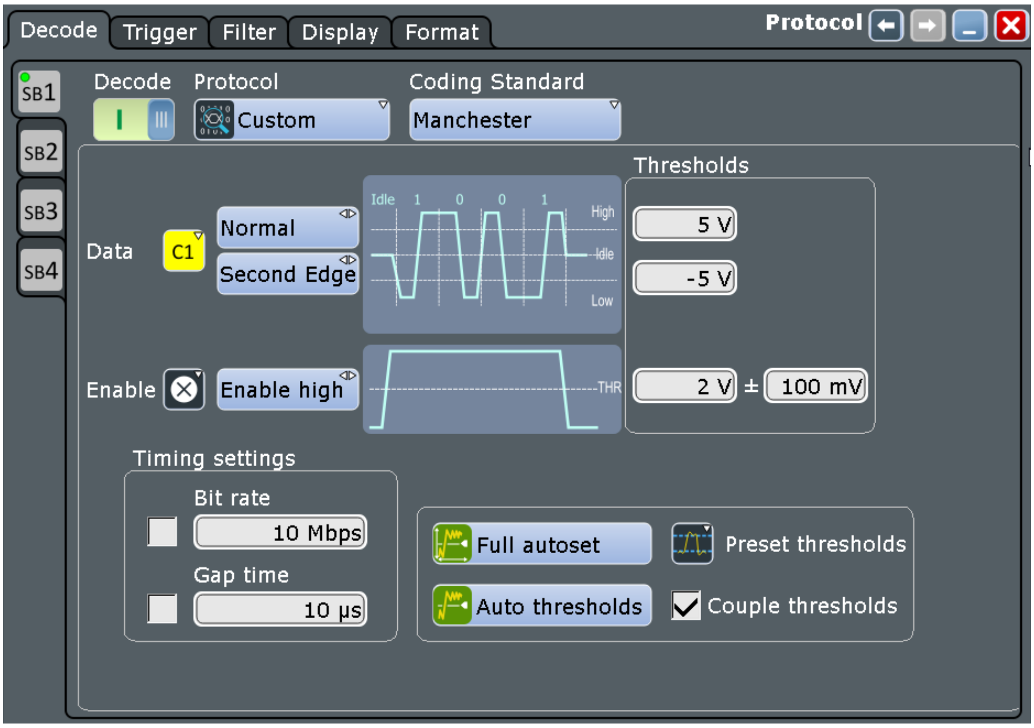


Figure 12-77: Coding standard selection in the serial bus protocol configuration dialog

For general information on how to configure protocol parameters, see also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Coding Standard
To define the coding of the custom serial bus to be analyzed, select one of the following standards:

"Manchester" Selects the coding standard Manchester.
Optional "Timing settings" are "Bit Rate" (default: disabled, 10 Mbit/s) and "Gap time" (default: disabled, 10 µs), as shown in Figure 12-77.

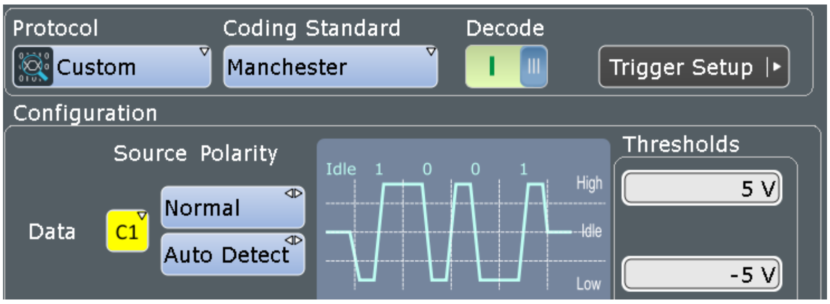


Figure 12-78: Custom serial bus coding configuration Manchester

"Manchester II" Selects the coding standard Manchester II, which is the inverted signal of the coding standard Manchester.
Optional "Timing settings" are "Bit Rate" (default: disabled, 10 Mbit/s) and "Gap time" (default: disabled, 10 µs), as shown in Figure 12-77.

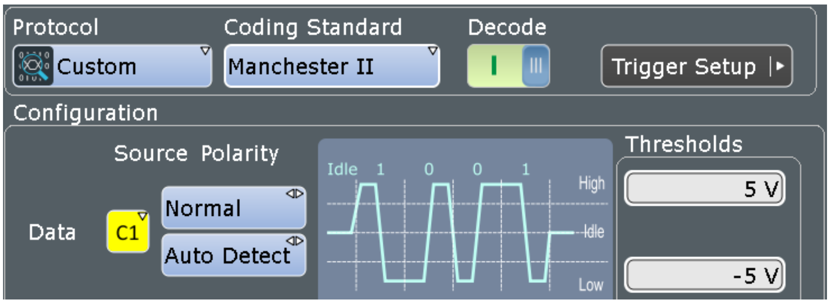


Figure 12-79: Custom serial bus coding configuration Manchester II

w

"NRZ Clocked" Selects the coding standard NRZ Clocked.
Optional "Timing settings" is "Gap time" (default: disabled, 10 µs), as shown in Figure 12-80.

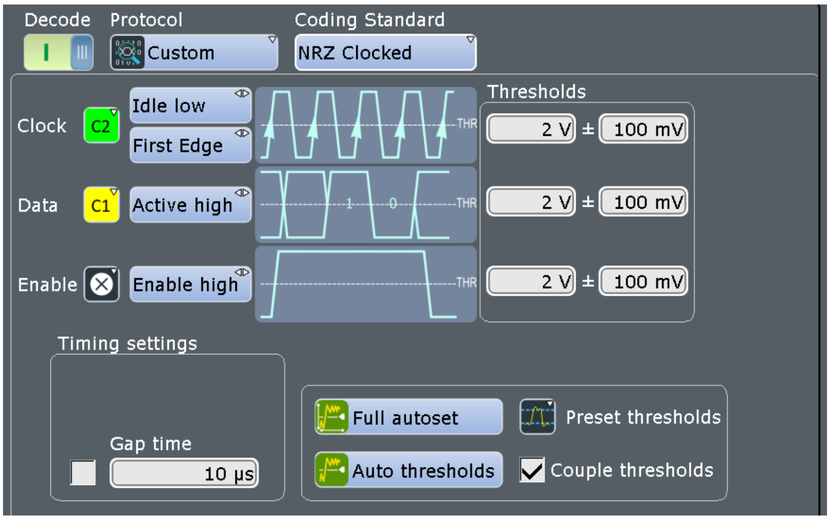


Figure 12-80: Custom serial bus coding configuration NRZ clocked

"NRZ Unclocked" Selects the coding standard NRZ unclocked.
Obligatory "Timing settings" are "Bit Rate" (default 10 Mbit/s) and "Gap time" (default 10 µs), as shown in Figure 12-81.

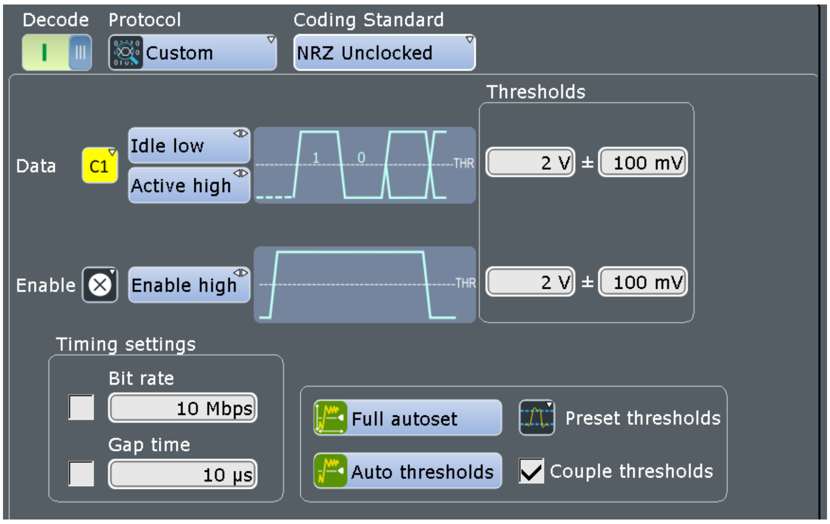


Figure 12-81: Custom serial bus coding configuration NRZ Unclocked

Remote command:
`BUS<m>:CMSB:CODing` on page 1695

Data Source

Defines the input source for the custom serial bus data signal.

The data source for Manchester and NRZ coding standards is selected separately, independent of each other. The data source is set to default upon switching the coding standard.

Permitted source selections are:
- For "Manchester"/ "Manchester II":
  - Decoding: the analog, mathematical, and reference channels
  - Triggering: the analog channels
- For "NRZ Clocked":
  - Decoding: the analog, mathematical, reference and digital channels
    Digital channels can be only used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.
  - Triggering: the analog and digital channels
- For "NRZ Unclocked":
  - Decoding: the analog, mathematical, reference and digital channels
    Digital channels can be only used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.
  - Triggering: the analog channels

Remote command:
BUS<m>:CMSB:MANChester:DATA on page 1696
BUS<m>:CMSB:NRZ:DATA on page 1699

Clock Source
Defines the source for the custom serial bus clock signal (only available for the coding standard "NRZ Clocked").

Permitted source selections are the analog, mathematical, reference and digital channels.

Digital channels can be only used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on the serial bus, when the NRZ clocked coding standard is selected, analog or digital channel sources are required.

Remote command:
BUS<m>:CMSB:NRZ:CLCK on page 1698

Data Polarity (Manchester)
Defines the polarity of the custom serial bus data signal in Manchester coding standards. The available settings are "Normal" or "Inverted".

Remote command:
BUS<m>:CMSB:MANChester:POLarity on page 1696

Data Polarity (NRZ)
Defines the polarity of the custom serial bus data signal in NRZ coding standards. The available settings are:

"Active high" The value "1" is represented by a voltage above the threshold.

"Active low" The value "1" is represented by a voltage below the threshold.

Remote command:
BUS<m>:CMSB:NRZ:POLarity on page 1701

Data Idle Polarity (NRZ Unclocked)
Defines the idle polarity of the custom serial bus data signal (only available for the coding standard "NRZ Unclocked"). The available settings are:

"Idle low" The base value of the clock is "0"; after an idle period, the data signal starts with a low-to-high transition.

"Idle high" The base value of the clock is "1"; after an idle period, the data signal starts with a high-to-low transition.

Remote command:
`BUS<m>:CMSB:NRZ:IDLPolarity` on page 1699

Clock Polarity (NRZ Clocked)

Defines the polarity of the custom serial bus clock signal (only available for the coding standard "NRZ Clocked"). The available settings are:

"Idle low" The base value of the clock is "0".

"Idle high" The base value of the clock is "1".

Remote command:
`BUS<m>:CMSB:NRZ:CPOLarity` on page 1699

Clock Phase (Manchester)

Defines the phase of the custom serial bus clock signal for the Manchester coding standards. The available settings are:

"Auto Detect" Lets the decoder automatically select the method ("First Edge" or "Second Edge") for detecting the clock phase.

"First Edge"
- At "Idle" = "low": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge.
- At "Idle" = "high": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge.

"Second Edge"
- At "Idle" = "low": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge.
- At "Idle" = "high": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge.

Note: The requirement to specify "First Edge" or "Second Edge" (or let the decoder decide) has the following background: In Manchester coding, an edge is always a transition from high to low (0) or from low to high (1). However, if the signal comes from the idle state, this implies that right before the first valid edge, there is always an overhead transition from idle to high or from idle to low. Some standards may regard this as a valid transition. To avoid a potentially ambiguous situation, a decision has to be made if the first edge is indeed only some overhead transition - or a transition that needs to be sampled.
For more details on edge conditions, see Chapter 12.14.1.1, "Special Features of Manchester Coding", on page 733.

Remote command:
`BUS<m>:CMSB:MANChester:CPHase` on page 1698

Clock Phase (NRZ Clocked)

Defines the phase of the custom serial bus clock signal for the coding standard "NRZ Clocked", depending on "Clock Polarity". The available settings are:

"First Edge"	• At "Idle" = "low": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge • At "Idle" = "high": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge
"Second Edge"	• At "Idle" = "low": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge • At "Idle" = "high": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge

Remote command:
`BUS<m>:CMSB:NRZ:CPHase` on page 1700

Enable Source (NRZ)
Defines the input source for the custom serial bus enable signal.

If an input is chosen, signals are only decoded when this channel is in the enabled state. This allows you to mark a time when the signal on the selected source is active and when not.

Permitted source selections are the analog, mathematical, and reference channels.

When the serial bus trigger has been selected, the only permitted source selections are the analog channels "C1" – "C4", which are required for triggering.

Math and Reference channels can only be selected, if no serial bus trigger is selected.

Remote command:
`BUS<m>:CMSB:NRZ:ENBLe` on page 1700

Enable Polarity (NRZ)
Selects whether the transmitted enable signal is active when the voltage is below the Thresholds ("Enable low") or higher than it ("Enable high").

Remote command:
`BUS<m>:CMSB:NRZ:ENAPolarity` on page 1701

Thresholds
Sets the threshold value for the digitization of each signal line and the hysteresis. If the signal voltage on the line is higher than the upper threshold, the signal state is high. Otherwise, if the signal voltage is below the lower threshold, the signal state is considered low.

- Manchester coding standards use 3-state signals with an upper and a lower voltage threshold in the range of -25 V to +25 V. A low-to-high transition requires the signal to exceed the upper threshold; a high-to-low transition requires the signal to fall below the lower threshold.
  You can set a hysteresis only for the enable signal.
- NRZ coding standards use a single voltage threshold for the data line. The value is in the range of -25 V to +25 V. It is entered into the middle of three available threshold input fields, or into the upper available threshold input field for NRZ Unclocked.
  You can set the hysteresis for all signal lines.
- In the NRZ Clocked coding standard, there is an additional clock voltage threshold available. This value in the range of -25 V to +25 V is entered into the upper threshold input field.
  You can set the hysteresis for the clock line.

There are four ways to set the threshold:

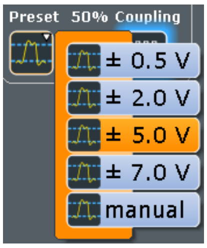
"Threshold" Directly sets the threshold values.

- For Manchester: upper threshold in the upper field, lower threshold in the lower field.
- For NRZ Clocked: clock threshold in the upper field, data threshold in the middle field and enable threshold in the lower field. Hysteresis values are in the right column.
- For NRZ Unclocked: data threshold in the upper field and enable threshold in the lower field. Hysteresis values are in the right column.

Remote command:
`BUS<m>:CMSB:MANChester:THReshold:HIGH` on page 1696
`BUS<m>:CMSB:MANChester:THReshold:LOW` on page 1697
`BUS<m>:CMSB:NRZ:THReshold:CLCK` on page 1701
`BUS<m>:CMSB:NRZ:THReshold:DATA` on page 1702
`BUS<m>:CMSB:NRZ:THReshold:ENBLe` on page 1702
`BUS<m>:CMSB:NRZ:HYSTeresis:CLCK` on page 1701
`BUS<m>:CMSB:NRZ:HYSTeresis:DATA` on page 1702
`BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe` on page 1702

"Preset"

- Either sets individual voltages by selecting "manual",
- or sets the voltages to one out of various pre-defined levels.



When any non-predefined threshold is set, the "Preset" status automatically changes to "manual" (without affecting anything else).

Remote command:
`BUS<m>:CMSB:MANChester:THReshold:PRESet` on page 1697
`BUS<m>:CMSB:NRZ:THReshold:PRESet` on page 1703

"Full autoset" Starts software algorithms for determining the signal threshold levels and bitrate. See also Chapter 12.1.2, "Full Autoset", on page 475.

Remote command:
`BUS<m>:FAUToset` on page 1383

"Auto threshold" Executes a measurement of reference levels and sets the thresholds to the middle reference voltage level of the measured amplitude.

Remote command:
`BUS<m>:SETReflevels` on page 1383

"Couple thresholds"
- For Manchester and Manchester II coding, the upper and lower thresholds are coupled to voltage values with the same magnitude but opposite sign (positive for the upper threshold and negative for the lower threshold). However, if the upper threshold is set to a negative voltage or the lower threshold is set to a positive voltage, coupling is disabled, and the other voltage (the one that was not actively set) is automatically adjusted, to avoid an upper threshold below the lower one, or a lower threshold above the upper one.
- For NRZ Clocked coding, the clock and data threshold values are coupled to the same voltage.

Remote command:
`BUS<m>:CMSB:MANChester:THReshold:COUPling` on page 1697
`BUS<m>:CMSB:NRZ:THReshold:COUPling` on page 1703

Enable Bit Rate
Enables the bit rate settings for the coding standards "Manchester" and "Manchester II". This setting is not available for "NRZ Clocked", but always enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard.

Remote command:
`BUS<m>:CMSB:BITRate:ENABle` on page 1704

Bit Rate
Defines the transmission speed setting for the data signal. A bit rate definition is optional for the coding standards "Manchester" and "Manchester II", not available for "NRZ Clocked", but obligatory for "NRZ Unclocked" (and also for triggering on signals in any coding standard). Default bit rate is 10 Mbps, permitted bit rates range from 300 bps to 2 Gbps.

For more details on the bit rate, see Chapter 12.14.1.1, "Special Features of Manchester Coding", on page 733.

Remote command:
`BUS<m>:CMSB:BITRate:VALue` on page 1704

Enable Gap Time
Enables the gap time settings (always enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard).

Remote command:
`BUS<m>:CMSB:GAPTime:ENABle` on page 1704

Gap time
Specifies a minimum gap time (idle time or timeout) between two frames. A gap time definition is optional for the coding standards "Manchester", "Manchester II" and "NRZ Clocked", but obligatory for "NRZ Unclocked" (and also for triggering on signals in any coding standard). Default gap time is 10 µs, permitted gap times range from 1 ns to 1 s.

For more details on gap time and idle conditions, see Chapter 12.14.1.1, "Special Features of Manchester Coding", on page 733.

Remote command:
BUS<m>:CMSB:GAPTime:VALue on page 1704

12.14.2.2 Frame Format Configuration

This dialog enables you to describe the generic format and logical structure of typical protocols by creating customized frame descriptions of various structures and lengths.

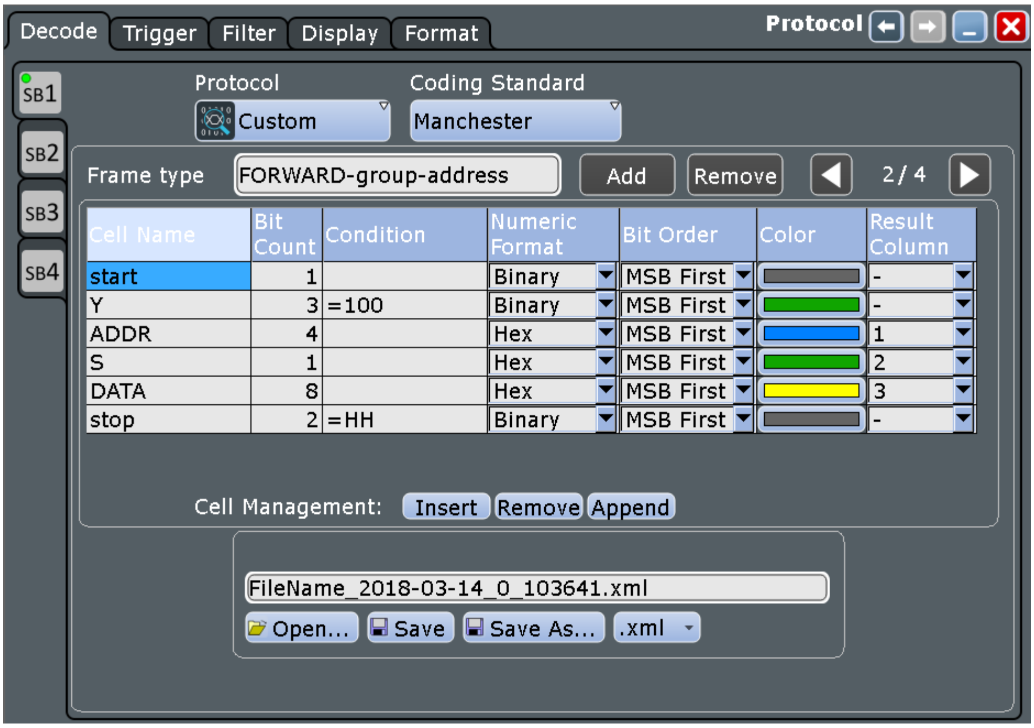


Figure 12-82: Example of a custom "DALI" frame format description (frame 2 of 4)

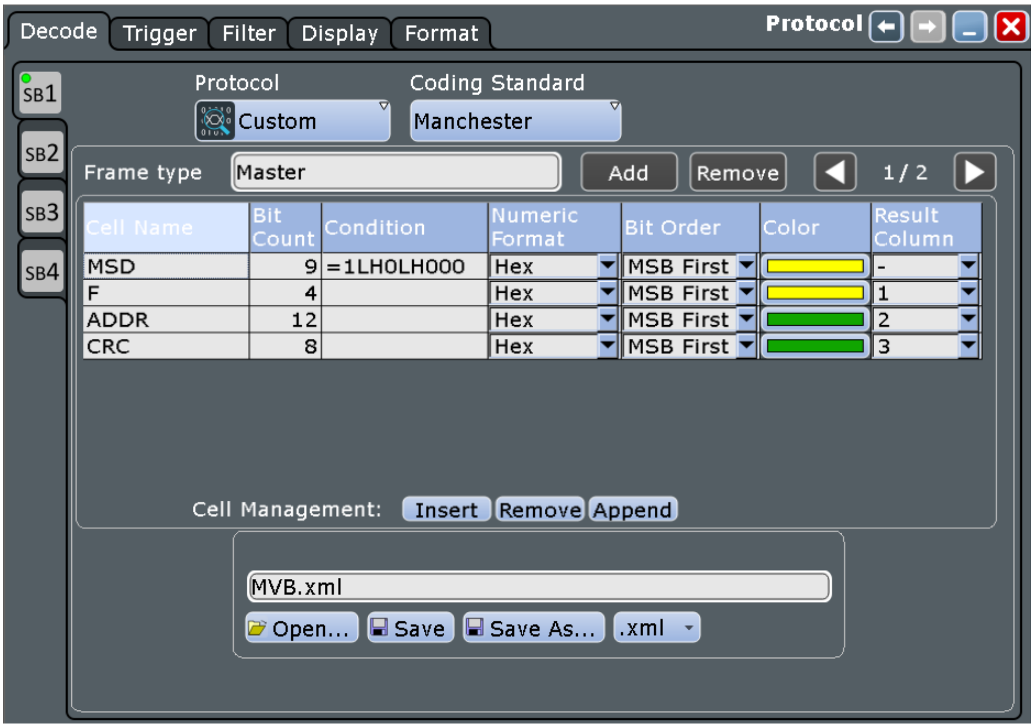


Figure 12-83: Example of a custom "MVB" frame format description (frame 1 of 2)

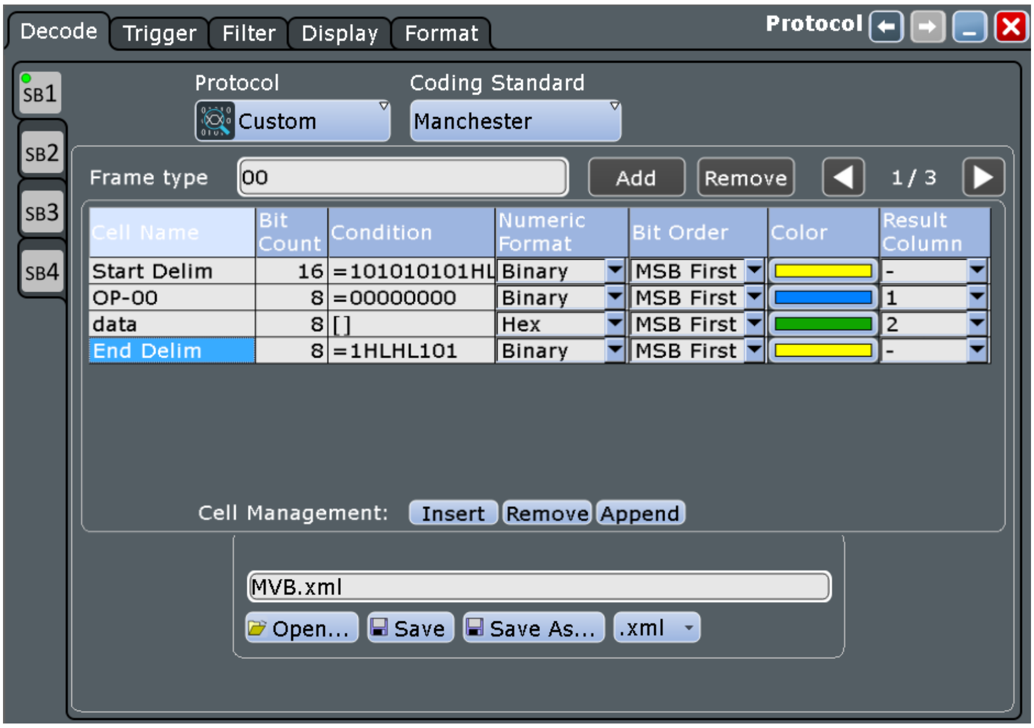


Figure 12-84: Example of a custom "ProfiBus Voltage" frame format description (frame 1 of 3)

Figure 12-85: Example of a crc, parity and block frame conditions

Frames

A frame format description (or *frame description*, for short) is represented by one "page" in this dialog. It can be created by clicking on "Add". If one or several frame descriptions already exist, the new frame description is then created at the end of the frame format description list (or *frame list*, for short). Describing a frame format requires assigning it a name as well as creating Cells and specifying cell descriptions. The number of frame descriptions is limited to 100. The frame description that is on display can be deleted from the frame list by clicking "Remove".

The "Frame type" string is intended for the user to label the frame description (typically according to the specifications of the applicable protocol standard). For example, MDIO (Management Data Input/Output) specifies the frames "READ", "WRITE", "ADDRESS". The example for the DALI protocol in Figure 12-82 has been created with the frame type "*FORWARD-group-address*".

The frame format dialog also provides the features "Open", "Save", "Save As...", and "Explore...", to store created sets of frame descriptions into files (in ".xml" format), or load existing files of this kind.

The frame identification is executed top down, in the order in which the frame formats are described in the frame list. This provides a hierarchy of criteria for identifying frames.

In case it is required to change the order of previously created frame descriptions, it is recommended to save the set of frame descriptions, and then edit the XML file with any suitable editor. This also allows - with due care - additional editing features, if required.

If no user-defined frame description should be suitable to identify an incoming frame, per default such a "missed" frame is reproduced as "Undescribed Bits" in the honeycomb display, as in the example in Figure 12-92. These bits are not shown in the results table.

Note: There may be frame descriptions that positively identifies each kind of frame, e.g. if no **equal** operator (see Condition) is defined for any of the cells. This "catches" every frame, even if there are other frame descriptions to follow in the frame list. Therefore, if a "catch all" frame description is used, it should be placed at the end of the frame list, or it overwrites any subsequent frame description. However, instead of using a "catch all" frame description, the built-in "undescribed bits" display as mentioned above may be the better approach to create frame descriptions.

It is in the responsibility of the user to define unambiguous settings for each frame type. For a description of these conditions in `XML` file format and the required `XML` grammar and syntax, see Chapter 12.14.2.3, "XML Syntax", on page 752.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:TYPE` on page 1705
`BUS<m>:CMSB:ADDFrame` on page 1705
`BUS<m>:CMSB:CLR` on page 1705
`BUS<m>:CMSB:FCOunt?` on page 1705

Cells

A cell description (which is represented by one row in one frame description) can be created at any position of a frame description (see Frames) by clicking on "Insert". This brings up a new cell description in the active frame description, on top of the selected position. The "Append" button adds a cell description at the end of a frame description, below the lowest existing cell description. The number of cell descriptions is not limited. The "Remove" button deletes a selected cell description from the active frame description.

The cell result can be selected to be displayed in a specified result column of the decode table (see "Result Column" on page 751).

Note: The cell descriptions must be sequential and complete. No gaps are allowed, since the Bit Count is used to calculate the start position of the next cell.

A frame type is identified when all user-defined cell conditions are met, which can be regarded as related by the Boolean AND operator.

This can also locate a synchronization pattern, specified by the equal operator in the Condition cell. For example, if you define a "Preamble" cell with the condition *=FFFFFFFF*, the decoder scans the data for this pattern, and then synchronize to it.

The cells in a frame are described by:

- Cell Name
- Bit Count
- Condition
- Numeric Format
- Bit Order
- Color
- Result Column

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CCOunt?` on page 1706
`BUS<m>:CMSB:FRAMe<n>:APPend` on page 1706

Cell Name

The strings in the column title describe cell names. They do not have to be unique; cell names are just for user support.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME` on page 1706

Bit Count

This crucial information defines the length of the cell and - based on the previous cells - also the cell end position and the next cell start position within a frame.

If, for a given bit count, the Condition value is longer, it is truncated. If the condition value is shorter, it is padded with 0. Both truncation and padding occur at the left side of the condition value.

Examples:
- if "Condition" is "=111000" and the Bit Order is "MSB", then
  - if "Bit Count" is 4, the truncated condition is "=1000"
  - if "Bit Count" is 8, the padded condition is "=00111000"
- if "Condition" is "=111000" and the bit order is "LSB" (accordingly, the condition in "MSB" format would be "=000111"), then
  - if "Bit Count" is 4, the truncated condition is "=1000" for LSB and "=0001" for MSB
  - if "Bit Count" is 8, the padded condition is "=00111000" for LSB and "=00011100" for MSB

These examples are true for the Numeric Format specified as "binary".

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount` on page 1706

Condition

This text field is used to apply various conditions and functionalities for a cell. Among others, it can be used to identify mandatory values (such as CRC checksum or ID) that help to identify a frame. The numeric format and bit order of the condition value has to match up with what is defined in the fields Numeric Format and Bit Order.

The following conditions are implemented:

"= (equal)"

The **equal** operator (represented by the "=" sign) defines a pattern for the cell to match. Valid condition entries are characters that match the cell's defined Numeric Format, Bit Order, and Bit Count. In binary format, for example, valid characters are "1", "0", "H" (high), and "L" (low).

Three cases have to be distinguished (cases A, B1, and B2), depending on the presence of a Variable Length Array: [] in the same frame description:

- **Case A**: If there is **no** "Variable Length Array" cell, then each cell marked with the equal operator acts as a key to identify a frame type. Only if all these cells match up with the expected value, the frame type is identified.
- **Cases B1 and B2**: If there **is** a "Variable Length Array" cell, then the equal operator has two different functionalities, depending on the position of the equal-operator cell within the frame description:
  - **B1**: If the cell is located *anywhere before* the "Variable Length Array" cell, the condition acts as a key to identify a frame type (as in case A).
  - **B2**: If the cell is located *immediately after* the "Variable Length Array" cell, the condition acts as an array delimiter.
    (Note: If the cell, which is marked with the equal operator, is located after the "Variable Length Array" cell, but *not* immediately after it, the decode result is unpredictable.)

Typically, Manchester protocols use code violations for synchronization. The states "H" and "L", supported by the equal operator in binary Numeric Format, mark that a transition is expected at this bit, but only a high or low signal is found.
Examples for the MVB protocol:
Master - Delimiter: "*=1LH0LH000*" (also shown in Figure 12-83)
Slave - Delimiter: "*=0000LH0LH*"
For more details on the violation symbols "H" and "L", see Chapter 12.14.1.1, "Special Features of Manchester Coding", on page 733.
Also, the length of the pattern must correspond to the bit field length (or the results are unpredictable).

"[ ] (array)"

The **array** operator (represented by the "[" and "]" bracket signs) defines the number of permissible repetitions of the cell.

Example: Fixed-Length Array: [n]

The length parameter "n" is a decimal number > 0, which determines that the cell is repeated n times within the frame. If, for example, the Bit Count is 8, then the array operator identifies n cells of 8-bit length, and present them in the results table and honeycomb display with the specified name and color.

A fixed-length array is treated the same as other cells, except the real length of such an array is n · bit count.

Example: Variable Length Array: []

This array operator with empty "[" and "]" bracket signs does not determine a fixed size array. The cell could be repeated any number of times, including 0 times. As a result, the cell and the frame are of unspecified length (a situation that covers typical use cases). For an example, see Figure 12-84.

The length of the frame is then determined by the end of frame condition, which can be an operator or a gap.

For processing reasons, only one (1) variable length array is supported in a frame, and a delimiter must follow immediately in the next cell after it. This is required to enable the software to correctly terminate the array. Otherwise the bits could not be assigned correctly, and it would not be possible to determine where a repletion starts and where it ends. With an end of frame condition, the software can calculate the length of one single array within a frame. But if there were more arrays, it would be impossible to know which array was how long.

The variable length array can also be the last cell of a frame. In this case, no delimiter is required. If decoded successfully, the detailed view in the results table shows the elements of the array. If the cell name of the array is "Data", then the detail view of result table displays the elements with an array index as "Data: 1", "Data: 2" ... etc.

If the variable length array cell is selected in the Result Column, it is shown as array [n], where n is the actual size detected in the waveform.

As an exception to the rule, it is permissible to assign the variable length array to consecutive fields. In this case, the fields are treated as a structure which is repeated. For example, if two consecutive fields are defined as A[] and B[], the decoder creates a sequence of ABABAB until the end condition has been detected.

Note: It is possible to combine check functions in a dynamic array. In the example above, if B[] is extended by odd(1), with "1" being the index of A[], then B checks the parity for each index of A.

"crc5usb(n-m)"	The **crc 5-bit** operator performs a check for a 5\-bit CRC function using the polynomial as defined by the USB standard. n and m define the index range for the CRC check. For example, if the CRC shall check fields 1 to 4, the function shall be written "crc5usb(1-5)". If the range of the CRC check includes an array, all elements in the array is included in the CRC check. If the check fails, the CRC field is marked as "CRC error" in the result details and displayed in the color red in the honeycomb display. The frame that contains the field is marked in the same way, except if another higher priority error is found within this frame.
"odd(n-m), even(n-m)"	The "**parity**" operators perform checks on odd or even parity in the given index range n to m. Odd parity is fulfilled if the count of "1" bits in the range including the parity bit is odd. Even parity is fulfilled if the count of "1" bits in the range including the parity bit is even. If the parity check fails, the parity field is marked as "CRC error" in the result details and displayed in red color in the honeycomb display. The frame that contains the field is marked in the same way, except if another higher priority error is found within this frame.
"{Block}"	The **block** operator is represented by the "{" and "}" bracket signs. Consecutive fields marked with "{Block}" and using the same name are displayed in the honeycomb display as a consecutive packet of name "Block" with the first field's color. This feature is a visual effect in the honeycomb only.

Remote command:
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition on page 1707

Numeric Format

Selects from the following numeric data formats for the Condition value:

- Decimal
- Hexadecimal
- Octal
- Binary

The following rules apply:

- If the condition value contains at least one "H", "h", "L", "l", "X", "x", and the remaining characters only contain "1" and/or "0", the numeric format is automatically interpreted as binary, regardless of its definition.
- The wild-card characters "x" and "X" are only supported in binary format.
  Examples: If the numeric format is set to be "HEX", then:
  – "=1HL111000" is valid (read as binary)
  – "=0x10101" is valid (read as binary, the "x" is interpreted here as a wild card)
  – "=1010" is valid (read as HEX, with a total of 16 bits)
  – "=0x5A" is valid (read as HEX "5A", since "0x" is a valid HEX prefix; nevertheless, it is recommended to enter "5A" instead)
  – "=5X12" is invalid
  – "=1H33" is invalid

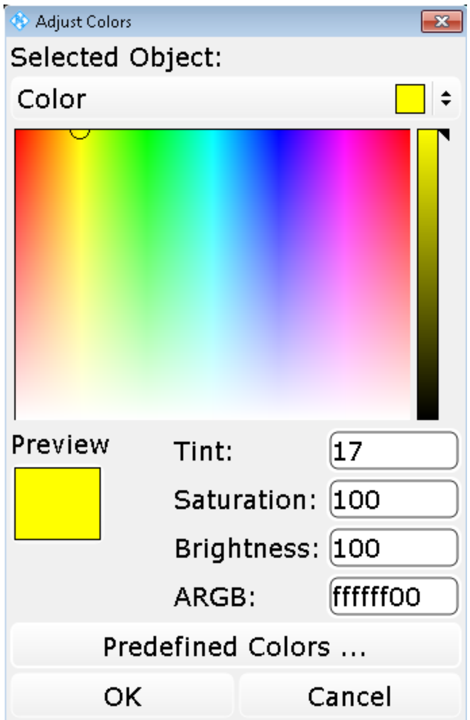
Remote command:
BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat on page 1707

Bit Order
This defines, in which order the bits of a cell's Condition value are evaluated: either the most significant bit (MSB) or the least significant bit (LSB) first. Since the bit order is considered for the interpretation of the condition, you should specify MSB or LSB correctly.

Remote command:
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder on page 1708

Color
Opens a dialog to select the color representation of different cells in the honeycomb display. Assigning user-selected colors helps to interpret the decode results more easily.

In the "Adjust Colors" dialog, you can either select one of the predefined colors or define a new one.



Remote command:
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB on page 1708

Result Column
This determines which cells shall be displayed in which result columns of the decode table. No index means that the result is not displayed. The decode table supports three result columns, which have to be unique for each frame type. For different frame types, though, the user can define different result columns, to display unrelated information.

Note: To see more than the three selected results, bring up a full list of the states and values of all cells by activating "Show details" in the decode table dialog. For an example, see Figure 12-91.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN` on page 1708

Open or Save XML File
For efficient working and for convenient exchange of frame descriptions, they can both be loaded ("Open") or saved ("Save" / "Save As...") in `XML` file format. "Explore..." opens the `SaveXML` folder, which is the "Default Path" for saving frame descriptions.

Remote command:
`BUS<m>:CMSB:LOAD` on page 1709
`BUS<m>:CMSB:SAVE` on page 1709

12.14.2.3 XML Syntax

This chapter explains the required grammar and syntax of `XML` files, which contain frame descriptions and can be loaded or saved. Below is a typical example of such an `XML` file:

```
<?xml version="1.0" encoding="utf-8"?>
<FrameDescription xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Version="1" xsi:noNamespaceSchemaLocation=".\Schema\FrameDescription.xsd">
  <Frame Type="00">
    <Format Name="Start Delim" BitCount="16" Condition="=101010101HL" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
    <Format Name="OP-00" BitCount="8" Condition="=00000000" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="0080ff" Column="1"/>
    <Format Name="data" BitCount="8" Condition="[]" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="10a500" Column="2"/>
    <Format Name="End Delim" BitCount="8" Condition="=1HLHL101" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
  </Frame>
  <Frame Type="01">
    <Format Name="Start Delim" BitCount="16" Condition="=101010101HL" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
    <Format Name="OP-01" BitCount="8" Condition="=00000001" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="0080ff" Column="1"/>
    <Format Name="data" BitCount="8" Condition="[]" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="10a500" Column="2"/>
    <Format Name="CRC" BitCount="16" Condition="" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="ff808080" Column="2"/>
    <Format Name="End Delim" BitCount="8" Condition="=1HLHL101" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00"" Column="-"/>
  </Frame>    <Frame Type="ff">
    <Format Name="Start Delim" BitCount="16" Condition="=101010101HL" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
    <Format Name="OP-ff" BitCount="8" Condition="=11111111" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="0080ff" Column="1"/>
    <Format Name="data" BitCount="8" Condition="[]" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="ffb233af" Column="2"/>
    <Format Name="CRC" BitCount="16" Condition="" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="ff808080" Column="2"/>
    <Format Name="End Delim" BitCount="8" Condition="=1HLHL101" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
  </Frame>
</FrameDescription>
```

Figure 12-86: Example of XML file syntax with three custom frame format descriptions

The first out of three `XML` frames in Figure 12-86 is interpreted by the software in the following way:

Cell Name	Bit Count	Condition	Numeric Format	Bit Order	Color	Result Column
Start Delim	16	=101010101HL	Binary	MSB First		-
OP-00	8	=00000000	Binary	MSB First		1
data	8	[]	Hex	MSB First		2
End Delim	8	=1HLHL101	Binary	MSB First		-

Figure 12-87: Example of one custom frame format description for the MVB protocol

For the context of this figure, see Chapter 12.14.2.2, "Frame Format Configuration", on page 743.

A suitable XML file as shown in Figure 12-86 is composed as follows:

Header:

```
<?xml version="1.0" encoding="utf-8"?>
```

Root Element:

```
<FrameDescription xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 Version="1" xsi:noNamespaceSchemaLocation=".\Schema\FrameDescription.xsd">
```

The root element contains the "Frame Description" attributes, including a link for the fileFrameDescription.xsd. This schema file, which is installed in the system, enables the software to validate an XML file before opening it.

Frame:

A frame description must include between *0* and *n* tags of the following kind:

```
<Frame> </Frame>
```

Frame Type:

Each "<Frame>" tag requires a "Type" attribute in string format:

```
<Frame Type = "string">
```

Tells the software the name of each frame, as described in section Frame Type.

Format:

Each frame must include between *1* and *n* tags of the following kind:

```
<Format> </Format>
```

Together with the attributes, this is written in short form, as in Figure 12-86:

```
<Format attribute... attribute... attribute... />
```

The format describes the fields (or Cells) in each frame. It can have the following attributes:

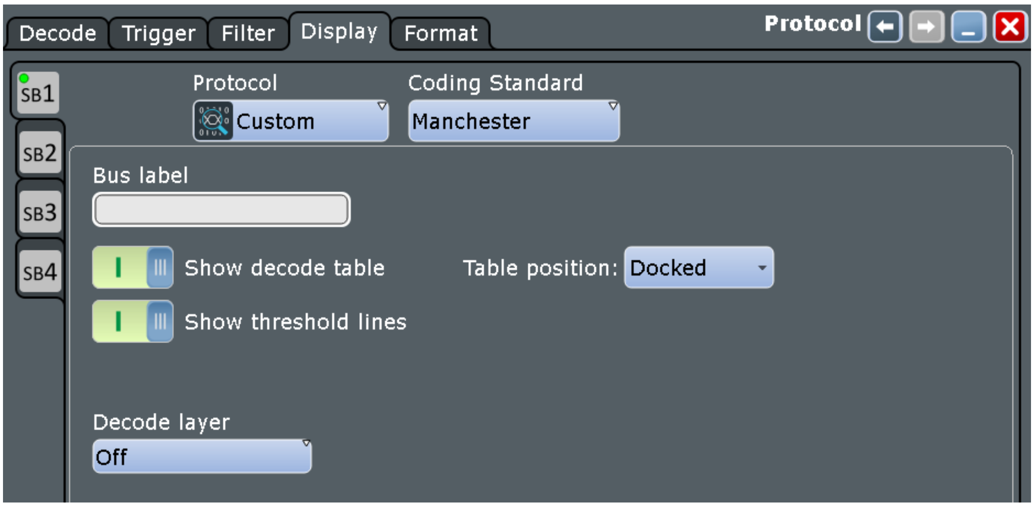
- Name (optional) is the "Cell Name", in string format.
- BitCount specifies the length of bits, in numerical format.
- Condition (optional) identifies the bit pattern to match, in string format.
- NumericFormat allows the following choices:
  – "Decimal"
  – "Hex"
  – "Octal"
  – "Binary"
- BitOrder allows two alternatives:
  – "MSB First" (most significant bit first)
  – "LSB First" (least significant bit first)
- Color allows you to set a user defined ARGB hexadecimal color value.
- Column is the "Result Column" with four options:
  – "-" (none, which is the default)

– "1"
– "2"
– "3"

12.14.2.4 Custom: Manchester / NRZ Display Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol = Custom" > "Display" tab

To enhance the decode possibilities of the custom serial protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Common display settings are explained in Chapter 12.1.3, "Display", on page 475.

Decode layer

Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Final"	...
"Edges"	...
"Binary"	...
"Synchronization"	...

12.14.2.5 Configuring Custom Manchester / NRZ Signals

For configuration, assign the lines to the input channels, define the active states and the logical thresholds, and specify frame format descriptions.

Serial bus setup

1. Press the [PROTOCOL] key on the front panel.

2. At the left-hand side, select the vertical tab of the serial bus (SB1–SB4) you want to set up.
3. Select the "Decode" tab.
4. Tap "Protocol" and select the protocol: "Custom".
5. Optionally, you can enter a "Bus label" in the "Display" tab.
6. Switch to the "Trigger" tab, tap "Source" and select "Serial bus".

   This prevents using digital waveforms (Math and Reference) as channel signals.

   Note: For triggering on a custom serial bus, analog input channels are required.
7. Switch back to the "Decode" tab.
8. Tap "Coding Standard" and select the coding ("Manchester", "Manchester II", "NRZ Clocked", or "NRZ Unclocked") you want to set up.
9. Select the polarity and phase of the data signal (and potentially of the clock signal).
10. Set the logical thresholds. See "Thresholds" on page 740.
11. Still in the protocol "Decode" tab, select "Decode" to activate the decode functionality.
12. Switch to the "Format" tab and open or create frame format descriptions.

For details on configuration settings, see Chapter 12.14.2.1, "Custom: Manchester / NRZ Configuration Settings", on page 735.

12.14.3 Custom: Manchester / NRZ Trigger

If you need information on how to get started with triggering on Custom serial bus signals, see Chapter 12.14.3.2, "Triggering on Custom Manchester / NRZ Serial Bus", on page 760. Otherwise proceed with the Custom serial bus trigger settings.

12.14.3.1 Custom: Manchester / NRZ Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = Custom"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected coding standard and trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

You can adjust the "Gap time" and "Bit rate" also in the "Trigger" tab if necessary.

For a list of supported trigger conditions, refer to the data sheet.

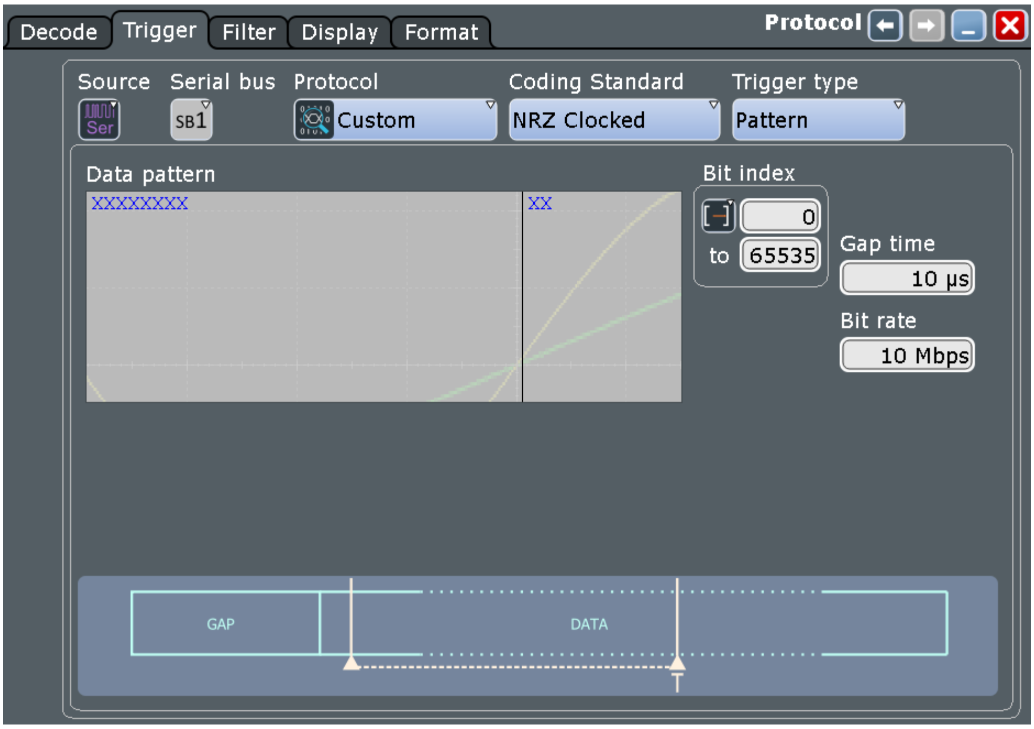


Figure 12-88: Custom serial bus trigger settings, here with NRZ Clocked and Pattern selected

Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Type

Defines the trigger type for custom serial bus analysis. The available trigger types are "Frame Start", "Pattern" and "Advanced".

Remote command:

`TRIGger<m>:CMSB:TYPE` on page 1710

Frame Start ← Type

For Manchester and NRZ Clocked coding standards, the frame start trigger is set to the end of the gap time. The start of frame (SOF) condition is the first bit after the gap (timeout).

For the NRZ Unclocked coding standard, the trigger requires that the signal contains a start bit. The frame start trigger follows the gap time and is set to the end of the start bit.

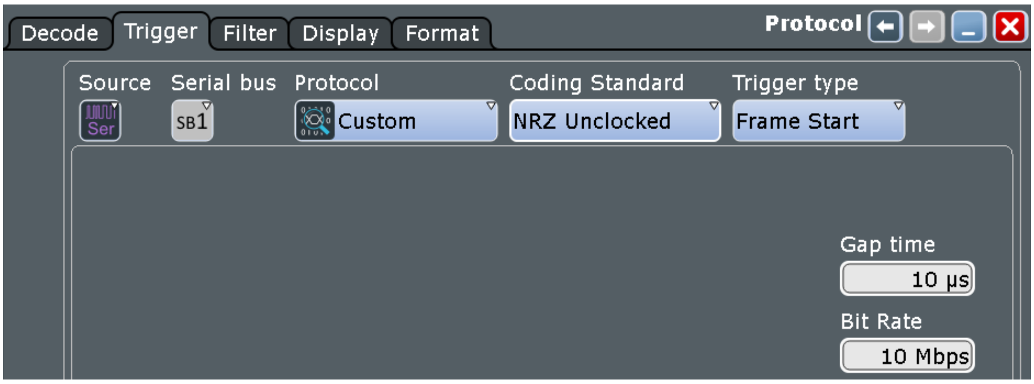


Figure 12-89: Custom serial bus: setting the trigger to frame start

Pattern ← Type

Specifies the pattern match conditions for a payload data check. The trigger is set to the first occurrence of a matching data bit pattern (which can be freely specified), starting after the minimum gap time, and after the detected start of the data frame. The trigger instant is after the last bit of the specified data pattern.

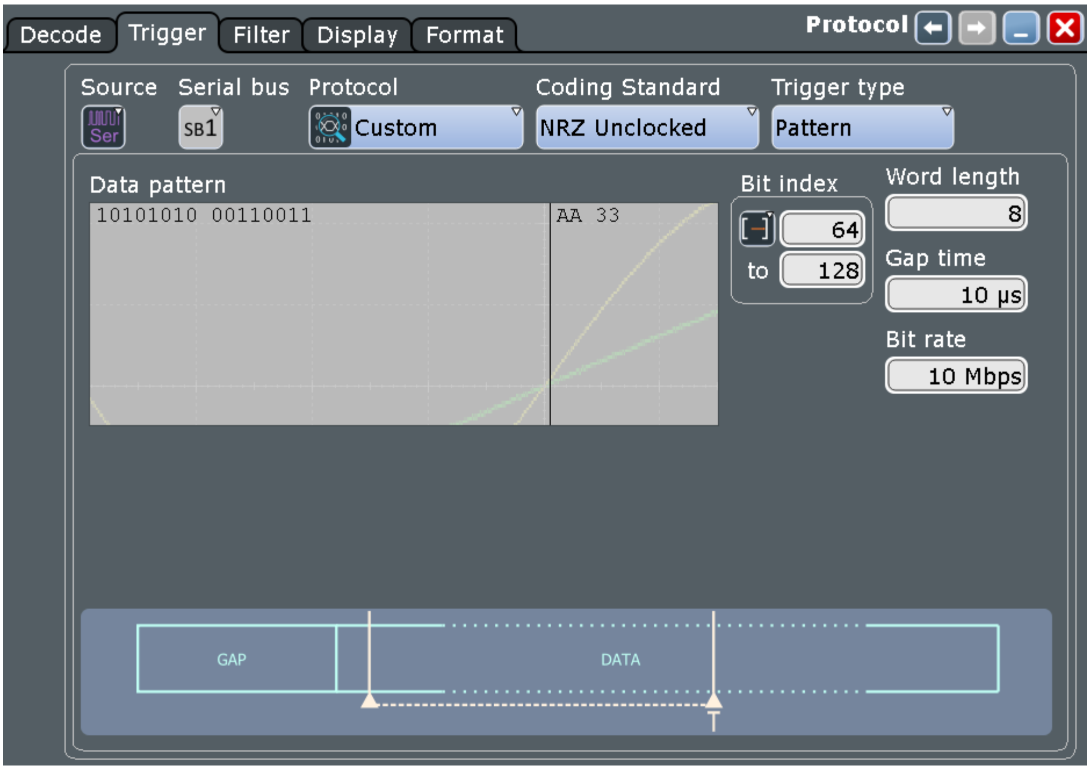


Figure 12-90: Custom serial bus: setting the pattern trigger, here for coding standard NRZ Unclocked

"Data Pattern"	Specifies the data pattern that is to be found and triggered. The pattern can be entered in binary or hexadecimal format, maximum pattern length is 256 binary characters or 64 hexadecimal characters.
"Bit index operator"	Sets the operator ("Equal", "Greater or equal", or "In range") to set a specific bit index (data position).

"Bit index" Sets the bit index (data position), or the start value of a bit index range. Default bit index value is 0, permitted values range from 0 to 65535.

"Bit index to" Sets the end value of a bit index range (data position range). Available only, if the "Bit index operator" is set to "In range". Default bit index end value is 65535, permitted values range from 0 to 65535.

"Word Length" Sets the number of bits in an NRZ Unclocked word (hence, the size of the data frame). Default word length is 8 bits, permitted lengths range from 0 bits to 31 bits .

Note: The NRZ Unclocked coding standard requires a signal that contains both a start bit and a stop bit:
- The start bit should be opposite in polarity to the idle state of the signal, and it is the first transition detected following the gap time.
- The stop bit should be the same polarity of the idle state, and it is the last bit in a data frame.

The end of the stop bit and the detection of the next frame's start bit constitutes the maximum gap time.

In order for the trigger to operate correctly, **you have to specify** the correct word length in the trigger menu. The trigger then counts the number of bits it decodes, and when the count matches the word length, the next bit is treated as the stop bit.

"Gap Time" Sets the minimum gap time for synchronization. The trigger is set to a position after the gap time, when the other trigger conditions are met. Default gap time is 10 µs, permitted gap times range from 1 ns to 1 s.

"Bit Rate" Sets the transmission speed for the data signal. Default bit rate is 10 Mbps, permitted bit rates range from 300 bps to 50 Mbps.

Remote command:
`TRIGger<m>:CMSB:PATTern` on page 1711
`TRIGger<m>:CMSB:ICONdition` on page 1711
`TRIGger<m>:CMSB:IMIN` on page 1711
`TRIGger<m>:CMSB:IMAX` on page 1712
`BUS<m>:CMSB:GAPTime:VALue` on page 1704
`TRIGger<m>:CMSB:NRZ:WRDLength` on page 1712

Advanced
Trigger on various frame types, fields in the frames, and data patterns.

Which settings are available, depends on what you have defined in the "Format" tab. For more information, see Chapter 12.14.2.2, "Frame Format Configuration", on page 743.

You can further refine the trigger criteria, selecting which frames you want to select for the triggering and what conditions their field values must fulfill.

Error types: Item name, Enable ← Advanced
The table lists the error types you can trigger on. Select the required errors in the "Enable" column. Available are "CRC" error and "Parity" error.

Remote command:
TRIGger<m>:CMSB:ADVanced:ERENable on page 1714
TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle on page 1714

Frame type ← Advanced
Selects the frame type for the custom bus trigger analysis.

You can define individual checking parameters for the fields listed in the "Trigger type dependent settings".

To specify these parameters, select a field from this list and define the data and/or index operators and values, or the bit state.

The trigger instant is the last criterion that is fulfilled.

Remote command:
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle on page 1712

Field name / Condition / Enable ← Advanced
This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Data", "BitState" and/or "Index" dialog (whichever applies). The condition is only applied, and the "Condition" is only shown in the table, if "Enable" is checked.

Remote command:
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle on page 1712
TRIGger<m>:CMSB:ADVanced:FIENable on page 1712

BitState ← Advanced
Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT on page 1713
TRIGger<m>:CMSB:ADVanced:BIT on page 1713

Data ← Advanced
Defines for the selected field, how a data check is executed.

"Condition"	Defining specific data or a data range requires to set the operator to one of the following conditions: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies data or sets the start value of a data range.
"Max"	Sets the end value of a data range, if "Condition" is set to "In range" or "Out of range".

Remote command:
TRIGger<m>:CMSB:ADVanced:DMAX on page 1713
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX on page 1713
TRIGger<m>:CMSB:ADVanced:DMIN on page 1714
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN on page 1714

TRIGger<m>:CMSB:ADVanced:DOPerator on page 1714
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator on page 1714

Index ← Advanced

Defines for the selected field, how an index check is executed.

"Condition"	Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.
"Min"	Specifies the index or sets the start value of an index range.
"Max"	Sets the end value of an index range, if "Condition" is set to "In range".

Remote command:
TRIGger<m>:CMSB:ADVanced:IMAX on page 1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX on page 1715
TRIGger<m>:CMSB:ADVanced:IMIN on page 1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN on page 1715
TRIGger<m>:CMSB:ADVanced:IOPerator on page 1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator on page 1715

12.14.3.2 Triggering on Custom Manchester / NRZ Serial Bus

Prerequisite: A bus is configured for the custom serial bus signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to Custom.
5. Tap "Type" and select the trigger type to be used for custom serial bus protocol analysis.
6. Depending on the selected custom serial bus coding standard, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 12.14.3.1, "Custom: Manchester / NRZ Trigger Settings", on page 755.

12.14.4 Custom Filter

With the filter function you can display only selected events in the acquired data. Only one single filter can be enabled on the protocol at a time.

12.14.4.1 Custom Filter Settings

Access: [PROTOCOL] > "Protocol = Custom" > "Filter" tab

Filter
Enables filtering on a custom serial bus.

Remote command:
BUS<m>:CMSB:FILTer:ENABle on page 1718

Frame type
Selects the frame type for the custom bus filtering.

You can define individual checking parameters for the fields listed in the "Trigger type dependent settings".

To specify these parameters, select a field from this list and define the data and/or index operators and values, or the bit state.

Remote command:
BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle on page 1719

Error types: Item name, Enable
The table lists the error types you can filter on. Select the required errors in the "Enable" column. Available are "CRC" error and "Parity" error.

Remote command:
BUS<m>:CMSB:FILTer:ERRor<n>:ENABle on page 1718

Field name / Condition / Enable
This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Data", "BitState" and/or "Index" dialog (whichever applies). The condition is only applied, and the "Condition" is only shown in the table, if "Enable" is checked.

Remote command:
BUS<m>:CMSB:FILTer:FIENable on page 1718

Bit State
Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
BUS<m>:CMSB:FILTer:BIT on page 1716

Data
Defines for the selected field, how a data check is executed.

"Condition"	Defining specific data or a data range requires to set the operator to one of the following conditions: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies data or sets the start value of a data range.
"Max"	Sets the end value of a data range, if "Condition" is set to "In range" or "Out of range".

Remote command:
`BUS<m>:CMSB:FILTer:DMAX` on page 1717
`BUS<m>:CMSB:FILTer:DMIN` on page 1717
`BUS<m>:CMSB:FILTer:DOPerator` on page 1717

Index

Defines for the selected field, how an index check is executed.

This function is avalible only for array fields, see "Condition" on page 747.

"Condition" Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.

"Min" Specifies the index or sets the start value of an index range.

"Max" Sets the end value of an index range, if "Condition" is set to "In range".

Remote command:
`BUS<m>:CMSB:FILTer:IMAX` on page 1719
`BUS<m>:CMSB:FILTer:IMIN` on page 1720
`BUS<m>:CMSB:FILTer:IOPerator` on page 1720

12.14.5 Custom: Manchester / NRZ Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Configuration" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" (and optionally "Show threshold lines"). For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the Frame Format Configuration settings.

The color-coding of the various Cells simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The Frame Format Configuration defines the cells and their Color scheme. The honeycomb display applies these settings according to the following rules:

- Each frame is displayed as a honeycomb frame with the frame type being displayed (in the 1st line of the honeycomb).
- Each cell (row) is displayed as a honeycomb cell with the name in the header line (2nd line in the honeycomb) and the formatted content in the value line (3rd line in the honeycomb).

Since the frame description is customizable, the result table has to be mostly generic. Due to the limited width of the result table, you have to select a limited number of results to be displayed by specifying it in the frame description. For example, the three results defined in the Result Column). This leads to a detailed view that shows all information on a per-frame base.

Examples

The example in Figure 12-91 shows decoded and binary signals of a custom serial bus. The format information of DALI is being used to display as a result.

Note that activating "Show details" in the decode table provides a more detailed analysis of decode results for one selected frame. This brings up a list of the states and values of all cells of the selected frame (in binary format). With this details dialog open, you can still click on the basic decode table, to change the selection of the frame to be displayed in detail.

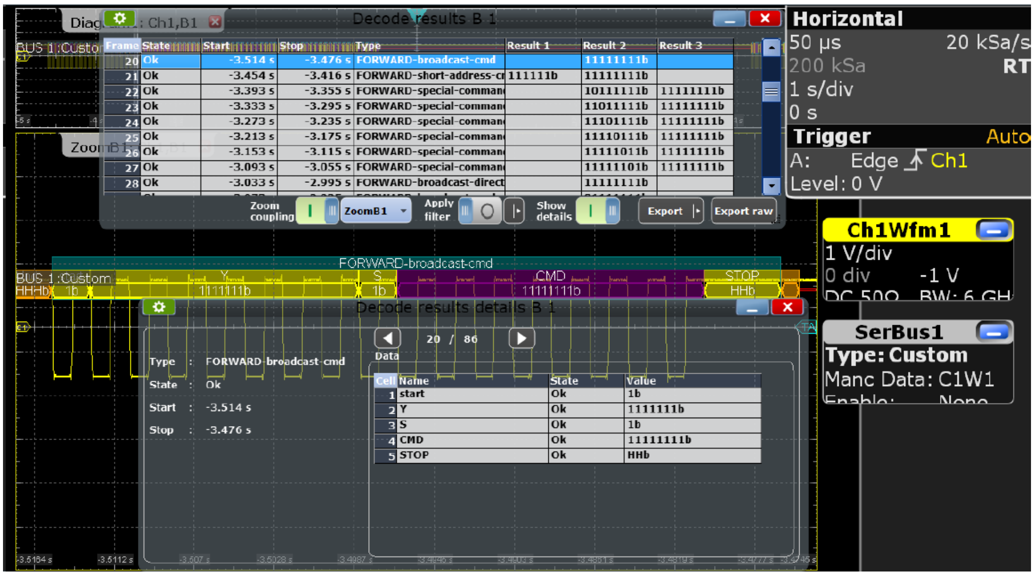


Figure 12-91: Decode results table and details of a "DALI" protocol sample waveform

In the honeycomb display, Cells are shown in the Color that have been set in the Frame Format Configuration.

For example, the result "array[297]" in "Frame 4", "Result 2" of the decoding table in Figure 12-91, is the short name for an array field display, and the number in the brackets indicates the length of the array, as described in Variable Length Array: [].

The tables "Decode results" and "Decode results details" in Figure 12-91 are described in Table 12-17 and Table 12-18:

Table 12-17: Content of the "Decode results" table

Column	Description
State	Overall state of the frame: either OK or the relevant error condition (e.g. preamble, length)
Start	Start time of the frame
Stop	Stop time of the frame
Type	Frame type as specified in the "Frame type" field of the "Frame Format" description dialog (see "Frames" on page 745)
Result 1	1st cell content as specified in the Result Column of the "Frame Format" description dialog (see "Frames" on page 745)

Column	Description
Result 2	2nd cell content (as above)
Result 3	3rd cell content (as above)

Table 12-18: Content of the "Decode results details" table

Column	Description
Name	Name of the cell (e.g. Start, Data) as specified in the Cell Name column of the "Frame Format" description dialog (see "Frames" on page 745)
State	Overall state of the cell: either OK or the relevant error condition (e.g. length error)
Value	Data content of the cell (e.g. 0x1, 1000LL00L)

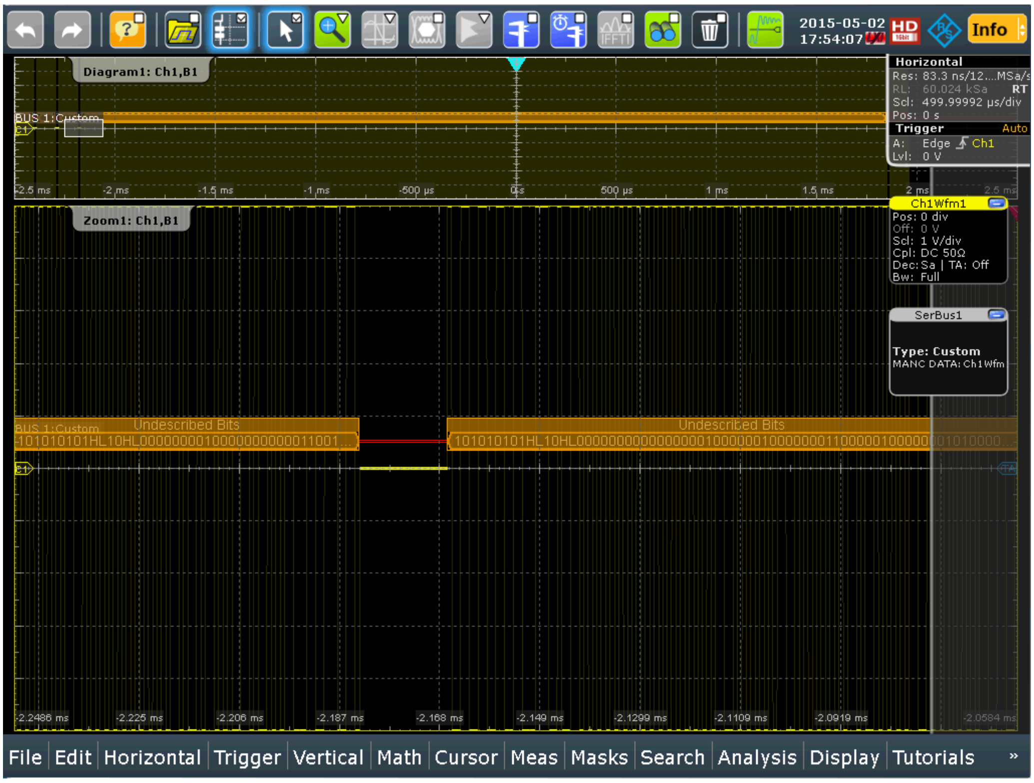


Figure 12-92: The function "Undescribed Bits" catches frames missed by the frame format descriptions

In the result presentation, frames labeled "Undescribed Bits" (as in Figure 12-92) show the bit patterns that are not matched by any user defined frame format description. Showing these raw bits is a functionality to help you develop suitable frame format descriptions.

The following commands are used to retrieve decode results in remote control. For an example on how to query the status of a frame, see Chapter 17.17.15.4, "Decode Results", on page 1720.

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:
- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:
- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.15.4, "Decode Results", on page 1720.

12.14.6 Search on Decoded Custom Manchester / NRZ Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

If you need information on how to get started with searching custom serial bus data, see Chapter 12.14.6.3, "Searching Custom Serial Bus Data", on page 767. Otherwise proceed with the custom serial bus search setup.

12.14.6.1 Custom Manchester / NRZ Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for "Custom"

Search criteria

Use the "Search criteria" dialog to define the event types to be searched. Available event types are "Frame" and "Error".

Individual search parameters, which do not depend on the coding standard and trigger settings of the custom serial bus, can be specified in the tabs below the "Search criteria" dialog.

Example:

Search dialog Custom decode shows an example of a custom decode search dialog setup. In this example, you search for "Frame Type 1" and "Frame Type 2". "Frame Type 1" has an additional condition for the range of "Data" and, since the "Data" field is an array, also to the range in the index. So this will only search, if the "Data" field in "Frame Type 1" has a value between 0x10 and 0x50 in the first 100 indexes.

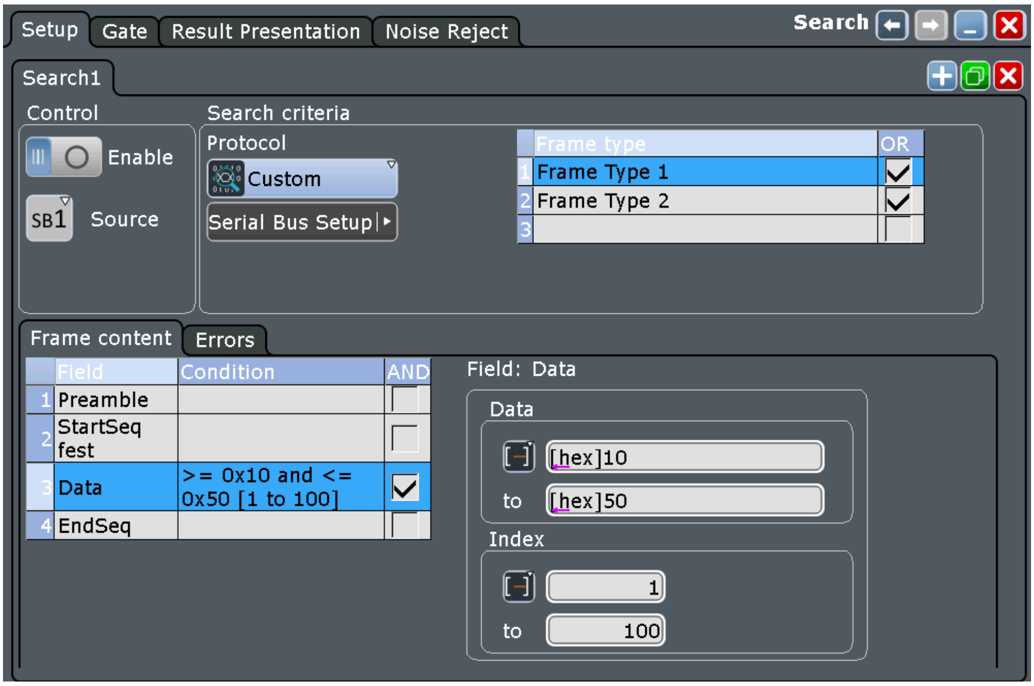


Figure 12-93: Search dialog Custom decode

Frame

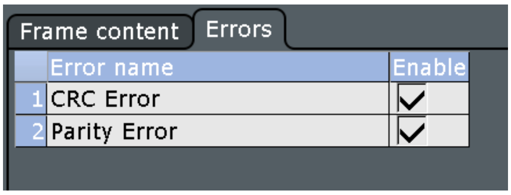
Searches for four different frame conditions.

Remote command:

SEARch:TRIGger:CMSB:ERRor<m>:ENABle on page 1727
SEARch:TRIGger:CMSB:FIENable on page 1727
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle on page 1730
SEARch:TRIGger:CMSB:FRENable on page 1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT on page 1728
SEARch:TRIGger:CMSB:BIT on page 1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX on page 1728
SEARch:TRIGger:CMSB:DMAX on page 1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN on page 1729

SEARch:TRIGger:CMSB:DMIN on page 1729
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator on page 1729
SEARch:TRIGger:CMSB:DOPerator on page 1729
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX on page 1730
SEARch:TRIGger:CMSB:IMAX on page 1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN on page 1731
SEARch:TRIGger:CMSB:IMIN on page 1731
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator on page 1731
SEARch:TRIGger:CMSB:IOPerator on page 1731

Error Condition
Searches for two error conditions: "CRC Error" or "Parity Error".



Remote command:
SEARch:TRIGger:CMSB:ERRor<m>:ENABle on page 1727

12.14.6.2 Custom Serial Bus Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

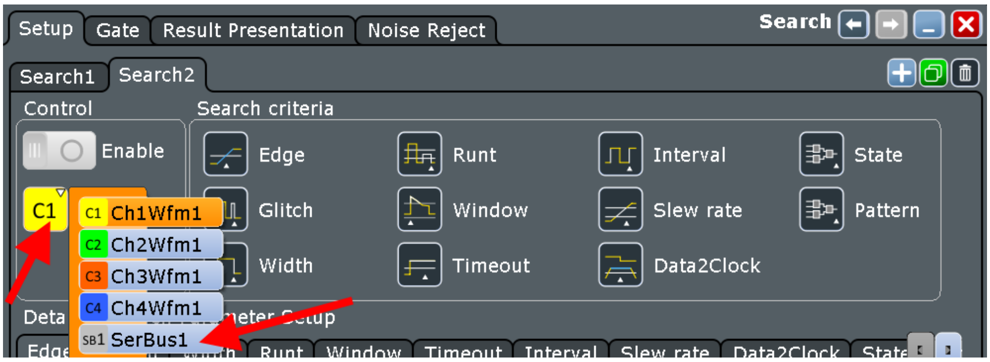
Remote commands are listed in Chapter 17.17.15.6, "Search Results", on page 1731

12.14.6.3 Searching Custom Serial Bus Data

Prerequisite: A serial bus is configured for the custom serial bus signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to "Custom" (e.g. "SB1", unless already selected).



The search dialog for custom serial bus protocol analysis is opened.

4. Specify search criteria according to Chapter 12.14.6.1, "Custom Manchester / NRZ Search Setup", on page 765.
5. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs a custom serial bus decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.15 MDIO (Option R&S RTE-K55)

The R&S RTE-K55 option enables the R&S RTE to analyse Management Data Input/Output (MDIO) protocols. The option is compatible with the Ethernet standard IEEE 802.3 (http://standards.ieee.org/findstds/standard/802.3-2012.html) and supports simplified triggering and decoding for both variants of MDIO: Clause 22 with basic addressing, and Clause 45 with advanced addressing that meets the requirements of 10 Gigabit Ethernet devices.

- The MDIO Protocol 769
- MDIO Configuration 770
- MDIO Trigger 773
- MDIO Label List 776
- MDIO Decode Results 777
- Search on Decoded MDIO Data 780

12.15.1 The MDIO Protocol

MDIO is used for bidirectional transfer of control and status information between the physical layer entity (PHY) and the station management entities (STA).

A major application of MDIO is fault detection by interrogating registers of physical devices. Hence, MDIO serial bus visualization helps debugging new products by giving developers a quick insight into the native data on the bus without using a special decoder.

On physical level, MDIO is a clocked non-return-to-zero (NRZ) code similar to SPI. According to the Ethernet standard, the protocol defines two threshold levels, 2 V and 0.8 V, which establish a hysteresis.

On logical level, MDIO is a fairly simple protocol with a fixed word length of 64 bits. The structure of MDIO frames is shown in the following tables:

Table 12-19: MDIO frame structure according to Clause 22

	Management Frame Fields							
Frame	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
WRITE	1...1	01	01	AAAAA	RRRRR	10	DDDDDDDDDDDDDDDD	Z
READ	1...1	01	10	AAAAA	RRRRR	Z0	DDDDDDDDDDDDDDDD	Z

Table 12-20: MDIO frame structure according to Clause 45

	Management Frame Fields							
Frame	PRE	ST	OP	PRTAD	DEVAD	TA	ADDRESS / DATA	IDLE
ADDRESS	1...1	00	00	PPPPP	EEEEE	10	AAAAAAAAAAAAAAAA	Z
WRITE	1...1	00	01	PPPPP	EEEEE	10	DDDDDDDDDDDDDDDD	Z
READ	1...1	00	11	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z
POST-READ increment address	1...1	00	10	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z

The following abbreviations are used in the tables:

- PRE = preamble, consisting of 32 logic "one" bits ("1...1")
- PRE = preamble, consisting of 32 logic "one" bits ("1...1")
- ST = start of frame code (2 bits), "01" for Clause 22, "00" for Clause 45, "0X" for any, no other options permitted
- OP = operation code or "OpCode" (2 bits). This is a frame type code specifying the type of transaction. For more details on the OpCode, see "OP" in Table 12-21, or `TRIGger<m>:MDIO:FRAMetype`.
- PHYAD = address of a physical layer entity (in Clause 22)
- PRTAD = address of a port (in Clause 45)
- REGAD = register address within a PHY (in Clause 22)
- DEVAD = device address within a port (in Clause 45)

- TA = turnaround time, a 2-bit time spacing between REGAD/DEVAD and DATA. The turnaround provides the slave some time to answer upon a read command. TA is hard-wired even in write commands, although it is not required there.
- ADDRESS / DATA = address or payload data, 16 bits
- IDLE = A single value (high-impedance state) indicating to the Physical Medium Attachment (PMA) that there is no data to convey

Instead of a specific hardware trigger, the option R&S RTE-K55 uses a predefined generic serial bus pattern trigger. It simply triggers on a bit pattern in the data stream. This is fast, but limited in the complexity of the conditions.

The MDIO trigger settings allow you to define the MDIO fields individually. The firmware concatenates the settings to a single search pattern that is then used by the serial bus pattern.

While this design is simple, it does not allow triggering on a data range or even inequality. This explains the much simpler structure compared to other protocols.

12.15.2 MDIO Configuration

12.15.2.1 MDIO Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *MDIO*

Make sure that the tab of the correct serial bus is selected on the left side.

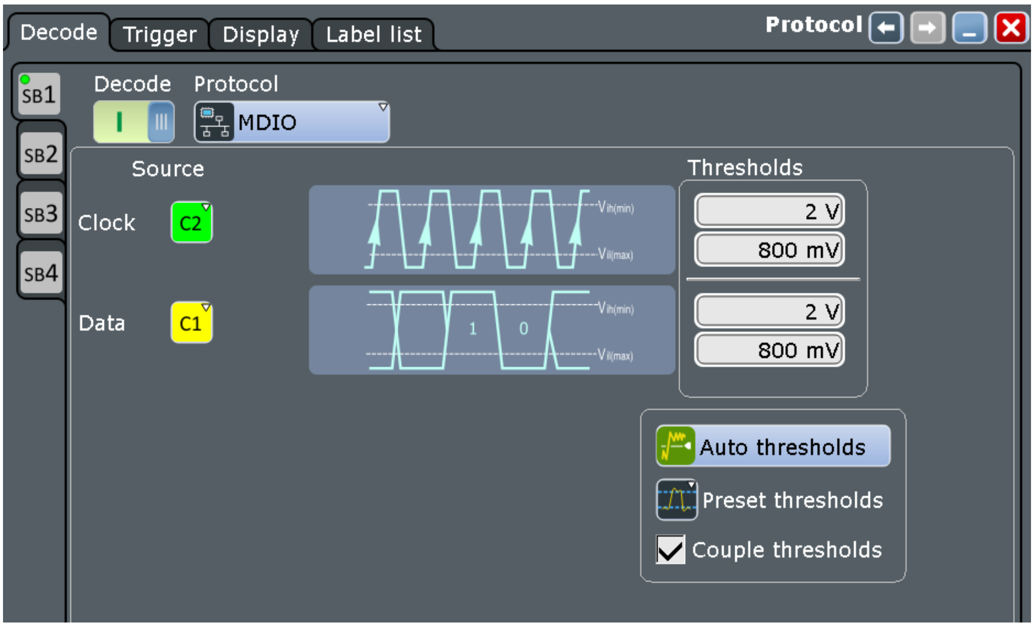


Figure 12-94: Serial bus MDIO protocol configuration dialog

For general information on how to configure protocol parameters, see also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Source
MDIO requires two source channels, one for clock and one for data.

"Clock" Defines the source settings for the clock line (management data clock, MDC). Typically, select any of the analog channels "C1" – "C4" or digital channels "D0" – "D15", depending on the test application. "Math" or "Ref" waveforms are also permitted.

"Data" Defines the source settings for the data signal. Typically, select any of the analog channels "C1" – "C4" or digital channels "D0" – "D15", depending on your application, but not the same as for "Clock". "Math" or "Ref" waveforms are also permitted.

Remote command:
`BUS<m>:MDIO:CLOCk:SOURce` on page 1738
`BUS<m>:MDIO:DATA:SOURce` on page 1738

Thresholds
MDIO defines two thresholds for each source line:
- Vih(min) is being used for the rising edge evaluation. This "h" (high) threshold is the minimum value for the signal to be identified as "1". If the signal value comes from a low state (hence, rising edge), the state remains to be considered as low ("0"), until it has risen above Vih(min).
- Vil(max) is being used for the falling edge evaluation. This "l" (low) threshold is the maximum level for the signal to be identified as "0". If the signal value comes from a high state (hence, falling edge), the state remains to be considered as high ("1"), until it has fallen below Vil(max).

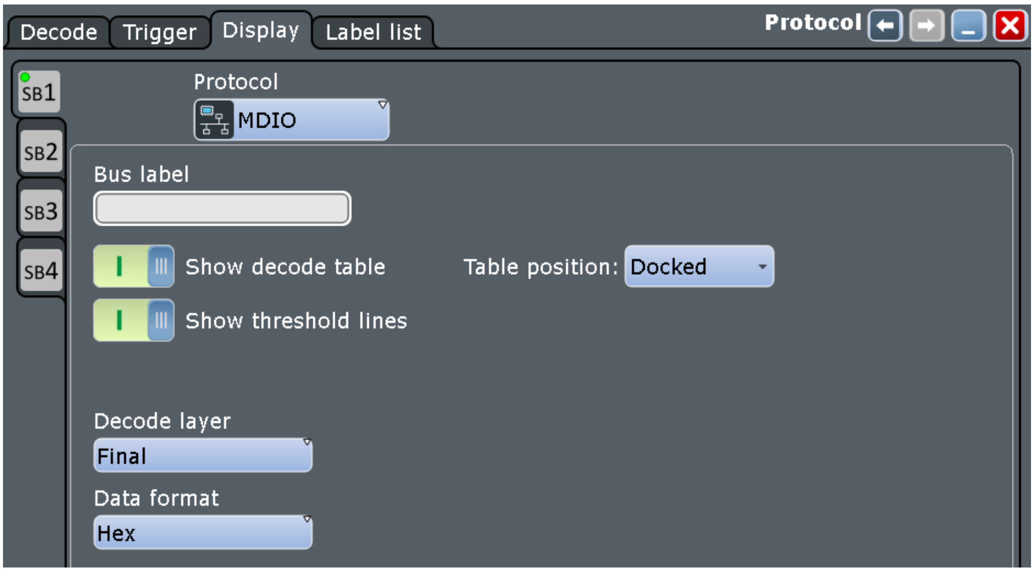
There are four ways to set the thresholds for the digitization of the signal lines:
- "Threshold"
  Enter the values directly in the fields.
- "Preset thresholds"
  Allows to select the default threshold settings according to the Ethernet standard: 2.0 V and 0.8 V.
- "Auto thresholds"
  Executes a measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Couple thresholds"
  Overwrites the data thresholds with the clock thresholds.

Remote command:
`BUS<m>:MDIO:CLOCk:THReshold:HIGH` on page 1739
`BUS<m>:MDIO:CLOCk:THReshold:LOW` on page 1739
`BUS<m>:MDIO:DATA:THReshold:HIGH` on page 1739
`BUS<m>:MDIO:DATA:THReshold:LOW` on page 1740
`BUS<m>:MDIO:PRESet` on page 1740
`BUS<m>:SETReflevels` on page 1383
`BUS<m>:MDIO:COUPling` on page 1740

12.15.2.2 MDIO Display Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol = MDIO" > "Display" tab

To enhance the decode possibilities of the MDIO protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Common display settings are explained in Chapter 12.1.3, "Display", on page 475.

Decode layer

Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Final"	...
"Edges"	...
"Binary"	...

12.15.2.3 Configuring MDIO Signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

1. Press the [PROTOCOL] key on the front panel.
2. At the left hand-side of the "Configuration" tab, select the vertical tab of the serial bus (SB1–SB4) you want to set up.
3. Tap "Protocol" and select the protocol: "MDIO".
4. Optionally, you can enter a "Bus label" in the "Display" tab.
5. Select the waveform for the "Clock" and "Data" lines.

6. Set the logical thresholds: Either according to technology definition with "Preset thresholds", or to the middle reference levels with "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields. Optionally, use "Couple thresholds" to couple the data thresholds to the clock thresholds.
7. In the protocol "Configuration" tab, select "Decode" to activate the decode functionality.

For details on configuration settings, see Chapter 12.15.2.1, "MDIO Configuration Settings", on page 770.

12.15.3 MDIO Trigger

12.15.3.1 MDIO Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = MDIO"

In this section, all trigger settings are described. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to the data sheet.

Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:
`TRIGger<m>:SOURce:SBSelect` on page 1387

Protocol
Defines the protocol type of the selected serial bus.

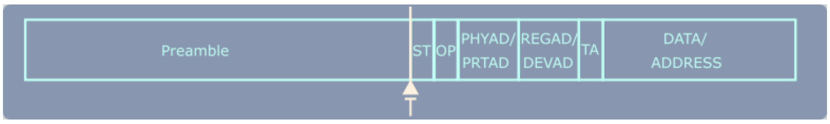
Remote command:
`BUS<m>:TYPE` on page 1382

MDIO Trigger Type

Selects the trigger type for MDIO analysis.

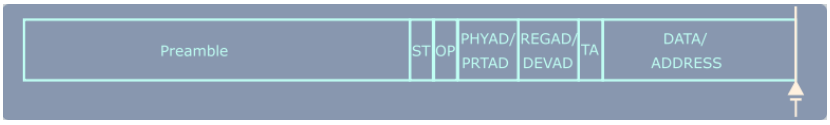
Figure 12-95: MDIO trigger event settings dialog

"Frame Start" Sets the trigger to the start of frame (SOF) field. The start of frame condition and the trigger instant is the end of the preamble.
Trigger pattern: preamble (32 bits "1")



There are no additional parameters to be specified.

"Frame Stop" Sets the trigger to the end of frame (EOF) field. The trigger instant is after the last data bit.
Trigger pattern: preamble (32 bits "1") + 32 bits "X"

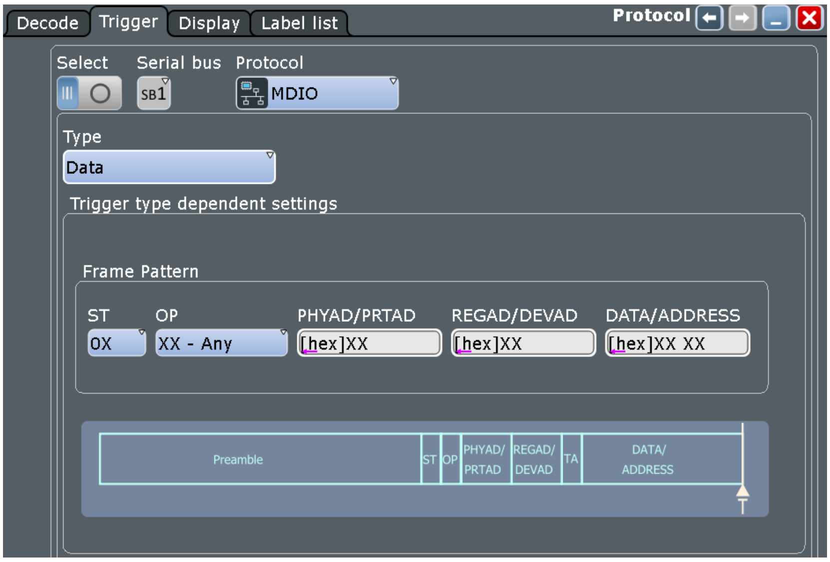


There are no additional parameters to be specified.

"Data" Sets the trigger to the data field. For more information on the data condition, see MDIO in the Ethernet standard. The trigger instant is at the end of the frame after the last data bit, as indicated in the GUI.

Note: All data triggers are always at the end of the frame, even if the specified pattern to trigger for is at a different position within the data word.

Trigger pattern: preamble (32 bits "1") + "ST" (2 bits, Start of Frame Code) + "OP" (2 bits, operation code or frame type code) + "PHYAD/PRTAD" (5 bits, Physical Layer Entity Address / Port Address) + "REGAD/DEVAD" (5 bits, Register Address / Device Address) + "TA" (2 "X" bits, turnaround time) + "DATA/ADDRESS" (16 bits)
For the parameters to be specified, see "ST" on page 775, "OP" on page 775, "PHYAD/PRTAD" on page 776, "REGAD/DEVAD" on page 776, and "DATA/ADDRESS" on page 776.



Remote command:
`TRIGger<m>:MDIO:TYPE` on page 1741

ST ← MDIO Trigger Type
Selects the start of frame code of the frame pattern; available only in trigger type "Data". Permissible frame patterns are: Clause 22, Clause 45, or Any

Remote command:
`TRIGger<m>:MDIO:ST` on page 1741

OP ← MDIO Trigger Type
Selects the type of frame code (or OP code, OpCode, operation code); available only in trigger type "Data". Available frame types are: Address, Write, Read, Post Read, or Any

Remote command:
`TRIGger<m>:MDIO:FRAMetype` on page 1742

PHYAD/PRTAD ← MDIO Trigger Type
Sets the physical address or port address (5 bits) of the frame pattern; available only in trigger type "Data".

Remote command:
`TRIGger<m>:MDIO:PHYS` on page 1742

REGAD/DEVAD ← MDIO Trigger Type
Sets the register address or device address (5 bits) of the frame pattern; available only in trigger type "Data".

Remote command:
`TRIGger<m>:MDIO:REGI` on page 1742

DATA/ADDRESS ← MDIO Trigger Type
Defines the payload data pattern or address pattern (16 bits); available only in trigger type "Data".

Remote command:
`TRIGger<m>:MDIO:DATA` on page 1742

12.15.3.2 Triggering on MDIO

Prerequisite: A serial bus is configured for the MDIO signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to MDIO, e.g.:

   The "Protocol" selection is then automatically set to "MDIO".
5. Tap "Trigger Type MDIO" and select the trigger type to be used for MDIO protocol analysis.
6. If the trigger type "Data" is selected, the frame pattern has to be specified.

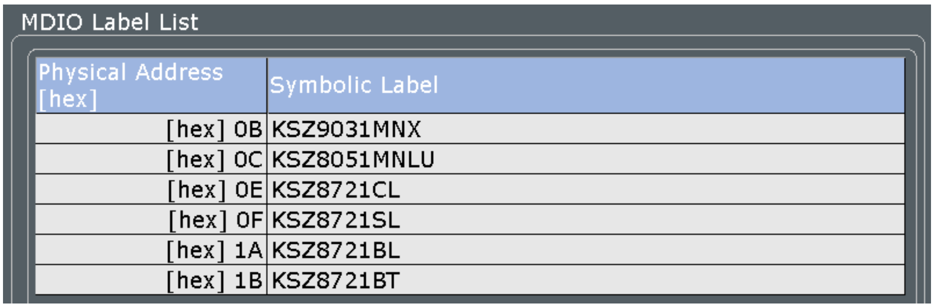
For information on how to proceed with the configuration settings, see Chapter 12.15.3.1, "MDIO Trigger Settings", on page 773.

12.15.4 MDIO Label List

Label lists are protocol-specific. A label list file for MDIO contains physical addresses and their symbolic names.

Example: MDIO label list CSV file

```
@PROTOCOL_NAME = mdio
0x0B, KSZ9031MNX
0x0C, KSZ8051MNLU
0x0E, KSZ8721CL
0x0F, KSZ8721SL
0x1A, KSZ8721BL
0x1B, KSZ8721BT
```


MDIO Label List

Physical Address [hex]	Symbolic Label
[hex] 0B	KSZ9031MNX
[hex] 0C	KSZ8051MNLU
[hex] 0E	KSZ8721CL
[hex] 0F	KSZ8721SL
[hex] 1A	KSZ8721BL
[hex] 1B	KSZ8721BT

For general information on the "Label List" tab, see Chapter 12.1.4, "Label Lists", on page 478.

Remote command:

- `BUS<m>:MDIO:WORD<n>:SYMBol?` on page 1746

12.15.5 MDIO Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Example

An example test waveform consisting of six frames is shown in Figure 12-96. The corresponding "Decode results" table for these frames can be seen in the foreground. The upper part of the screen, behind the table, represents the waveform in a honeycomb display, along with the binary decode results. In the lower part of the screen there is a zoom into frame #2, which is a "Write" frame, containing PRE, ST, OP, PRTAD, DEVAD, TA and DATA fields. The zoom shows both the honeycomb display and the binary decode results of that second frame.

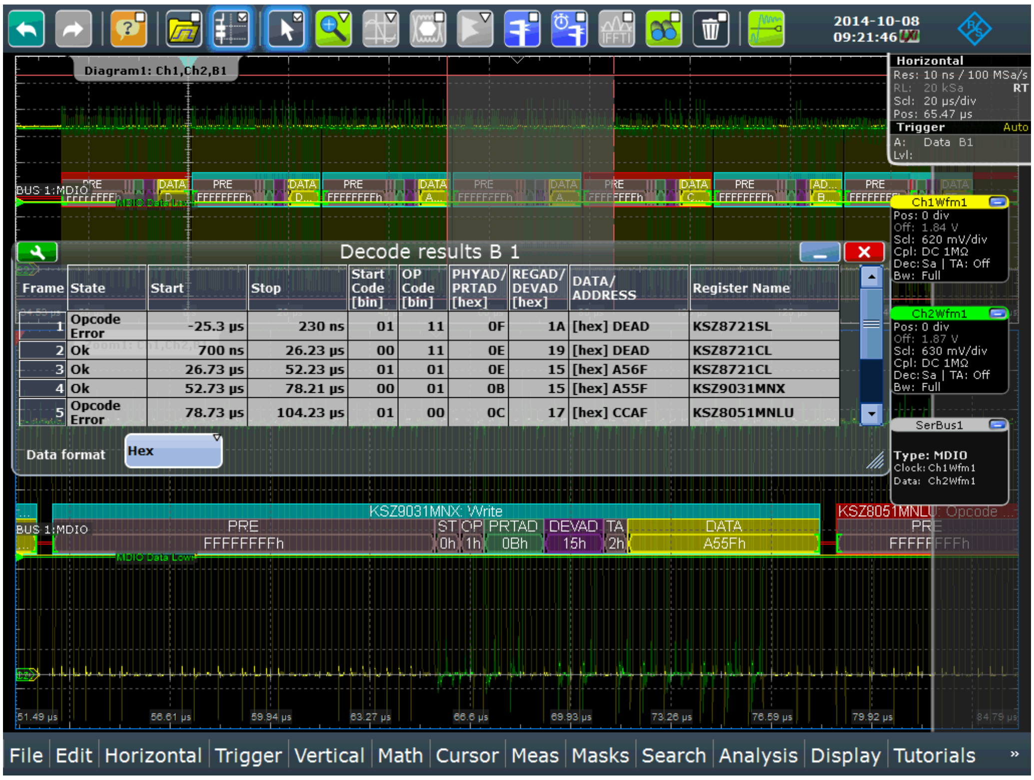


Figure 12-96: Decoded and binary MDIO signal trace, and decode results table

green brackets [...] = start and end of frame
blue frame = frame ok
red frame = frame containing an error
grey = preamble (PRE), start pattern (ST), operation code (OP = frame type), or turnaround (TA) fields
dark green = PHY address or port address (depending on clause)
dark purple = register address or device address field (depending on clause)
yellow = data field or address field (depending on clause)

Table 12-21: Content of the "Decode results" table in the previous figure

Column	Description
Frame	Number of the acquired frame
State	State of frame, available messages are: ● OK ● OPcode error ● Length error ● Incomplete frame ● Unsynchronized bits
Start	Start time of the frame
Stop	Stop time of the frame
ST	Start of frame code, 2 bits ● "01" (Clause 22) ● "00" (Clause 45)

Column	Description
OP	Operation code (= Frame type), 2 bits ● "00" = Address frame (in Clause 45, only) ● "01" = Write frame (in Clause 22 or Clause 45) ● "10" = Read frame (in Clause 22) or Post Read frame (in Clause 45) ● "11" = Read frame (in Clause 45)
PHYAD/PRTAD	Address field, shown as 2 hex characters (corresponding to 5 binary bits) ● PHY address (in Clause 22) ● Port address (in Clause 45)
REGAD/DEVAD	Address field, shown as 2 hex characters (corresponding to 5 binary bits) ● Register address (in Clause 22) ● Device address (in Clause 45)
DATA/ADDRESS	Payload data field (in Clause 22 or Clause 45), or Address field (in Clause 45, only), shown as 4 hex characters or 16 binary bits (see Figure 12-97).
Register Name	Displays a translation of the PHYAD/PRTAD address label in textual form

In the decode results table, the contents of column "DATA/ADDRESS" can also be displayed in alternative numerical formats, e.g. in binary format, as shown in Figure 12-97.

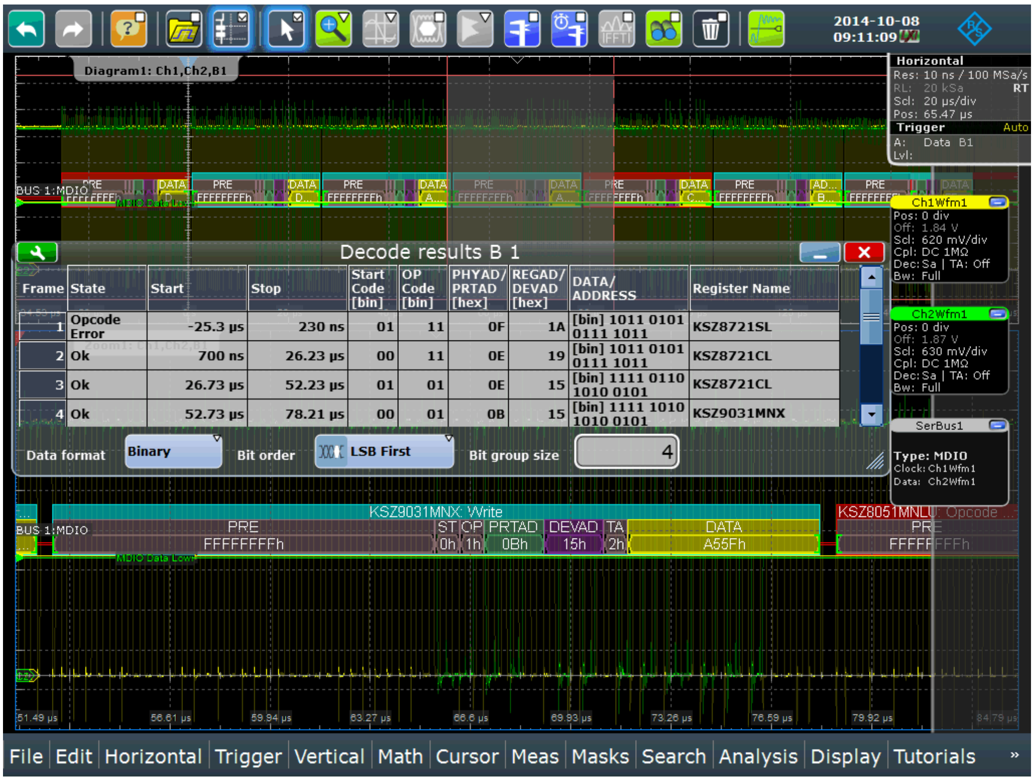


Figure 12-97: The same signal trace and decode results table as in the previous figure, but only with "DATA" in binary format

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.16.3, "Decode Results", on page 1743.

12.15.6 Search on Decoded MDIO Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

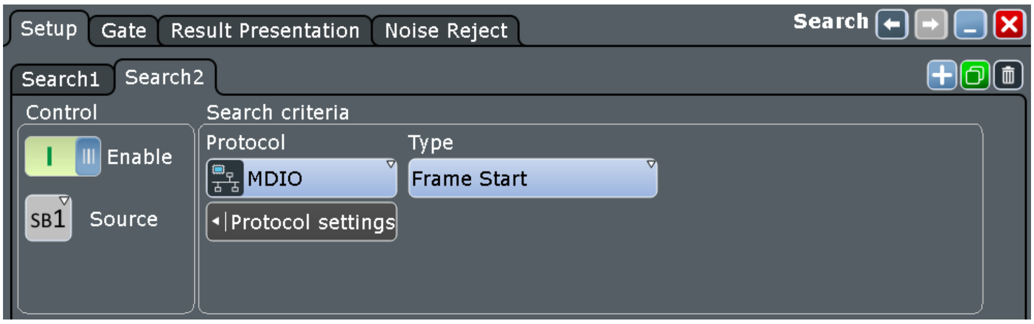
For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

If you need information on how to get started with searching MDIO data, see Chapter 12.15.6.3, "Searching MDIO Data", on page 782. Otherwise proceed with the MDIO search setup.

12.15.6.1 MDIO Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for MDIO

Search criteria
Use the "Search criteria" dialog to define the event types to be searched.



Available event types are "Frame Start", "Frame Stop", and "Data".

Only if search criteria type "Data" is selected, individual search parameters can be specified in the tabs below the "Search criteria" dialog. For these parameters, see "ST" on page 781, "OP" on page 781, "PHYAD/PRTAD" on page 781, "REGAD/DEVAD" on page 781, and "DATA/ADDRESS" on page 781.

Remote command:
`SEARch:TRIGger:MDIO:TYPE` on page 1748

ST
Selects the start of frame code of the frame pattern; available only in search criteria type "Data". Permissible frame patterns are: Clause 22, Clause 45, or Any

Remote command:
`SEARch:TRIGger:MDIO:ST` on page 1748

OP
Selects the Type of Frame code (or OP code, OpCode, operation code); available only in search criteria type "Data". Available frame types are: Address, Write, Read, Post Read, or Any

Remote command:
`SEARch:TRIGger:MDIO:FRAMetype` on page 1747

PHYAD/PRTAD
Sets the physical address or port address (5 bits) of the frame pattern; available only in search criteria type "Data".

Remote command:
`SEARch:TRIGger:MDIO:PHYS` on page 1748

REGAD/DEVAD
Sets the register address or device address (5 bits) of the frame pattern; available only in search criteria type "Data".

Remote command:
`SEARch:TRIGger:MDIO:REGI` on page 1748

DATA/ADDRESS
Defines the payload data pattern or address pattern (16 bits); available only in search criteria type "Data".

Remote command:
SEARch:TRIGger:MDIO:DATA on page 1747

12.15.6.2 MDIO Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:
- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

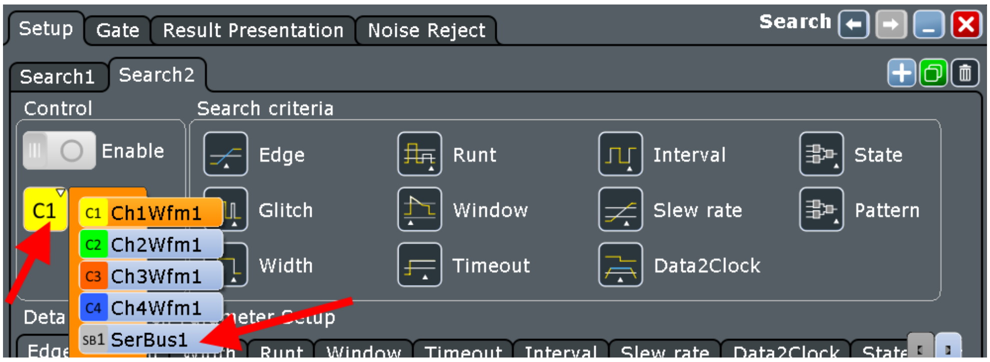
Remote commands:
- SEARch:RESult:MDIO:WCOunt? on page 1752
- SEARch:RESult:MDIO:WORD<m>:STATe? on page 1751
- SEARch:RESult:MDIO:WORD<m>:STARt? on page 1750
- SEARch:RESult:MDIO:WORD<m>:STOP? on page 1751
- SEARch:RESult:MDIO:WORD<m>:ST? on page 1750
- SEARch:RESult:MDIO:WORD<m>:TYPE? on page 1752
- SEARch:RESult:MDIO:WORD<m>:PHYS? on page 1749
- SEARch:RESult:MDIO:WORD<m>:REGI? on page 1750
- SEARch:RESult:MDIO:WORD<m>:DATA? on page 1749
- SEARch:RESult:MDIO:WORD<m>:SYMBol? on page 1752

12.15.6.3 Searching MDIO Data

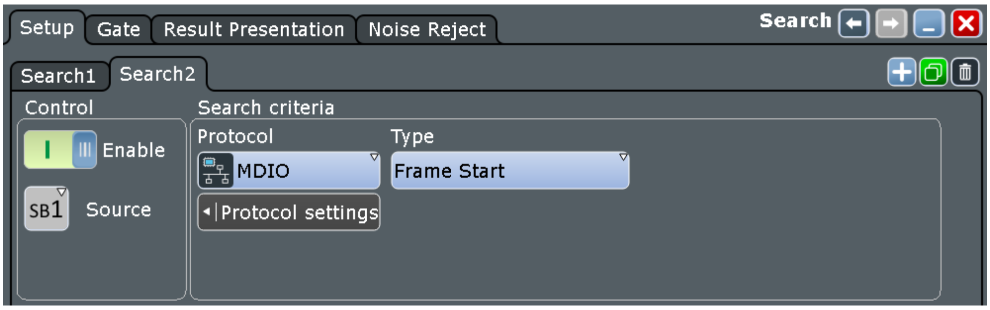
Prerequisite: A serial bus is configured for the MDIO signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to MDIO (e.g. "SB1", unless already selected).



The search dialog for MDIO protocol analysis is opened.



There are no additional search criteria to be specified.

4. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs an MDIO decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).

5. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.16 USB (Option R&S RTE-K60)

R&S RTE-K60 is a firmware option that enables the R&S RTE to analyse Universal Serial Bus (USB) protocols, by triggering and decoding them. The option is compatible with the standards USB 1.0, USB 1.1, USB 2.0 and USB HSIC (High-Speed Inter-Chip). R&S RTE-K60 supports the data rates "Low Speed" (1.5 Mbit/s), "Full Speed" (12 Mbit/s) and "High Speed" (480 Mbit/s, available in USB 2.0 and HSIC).

- The USB Protocol.. 784
- USB 2.0 Configuration.. 789
- USB 2.0 Trigger... 793
- USB 2.0 Decode Results... 803
- Search on Decoded USB 2.0 Data.. 806

12.16.1 The USB Protocol

The USB protocol was developed, starting in 1996, by the nonprofit organization USB Implementers Forum, Inc. (USB-IF), formed by Compaq, Hewlett-Packard, Intel, Lucent Technologies, Microsoft, NEC, and Philips. The purpose was to provide a common "plug-and-play" solution to replace a multitude of interfaces for the communication between computers and devices. It should allow even unskilled users to easily connect many devices to a PC. USB was originally used for devices that feature low signalling rates (up to 1.5 Mbit/s), especially human interface devices like mouse, joystick or keyboard. With the release of USB 2.0 in April 2000, devices such as printers, cameras and mass storage media were enabled to exchange data at faster rates (up to 480 Mbit/s). Also, the high-speed protocol HSIC (High Speed Inter Chip) is used for the communication between on-board devices.

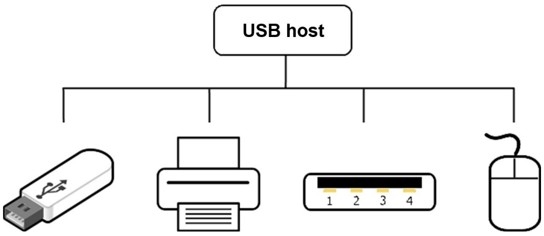


Figure 12-98: USB Topology: mass storage device, printer, USB hub, and mouse (human interface device) connected to a USB host

More information, including the USB specification, is available online within the web domain http://usb.org.

This chapter provides an overview of protocol characteristics, data transfer, packet structure, address and endpoint formats of USB and trigger possibilities.

USB characteristics

Main characteristics of USB are:

- Four-wire design: USB requires a shielded cable containing four wires. Two of them, called D+ and D-, form a twisted pair (for low speed, they may not be twisted). These data lines transmit differential data signals and single-ended signal states, both referenced to a third wire: the GND or ground. The fourth wire, called VBUS (voltage bus), carries a nominal 5 V supply, which may be used to power a device.
- Host-to-device communication: in USBs "speak-when-spoken-to" protocol, communication is always initiated by the host. Consequently, there is no direct communication between USB devices, apart from few exceptions.

- Addressing scheme: a maximum of 127 connected devices can be distinguished, because a packet's address field length is limited to 7 bits. USB devices have up to 16 OUT endpoints (from host to device) and up to 16 IN endpoints (from device to host).
- USB transactions consist of two or three packets: token, data, and typically handshake
- Packet type: a packet identifier (PID) is sent as a first byte within the packet and specifies the different packet types.
- NRZI (Non Return to Zero Inverted): a zero (0) is encoded as a transition of the physical level, whereas a one (1) has no transition, thus it is represented by a steady level.

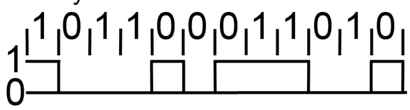


Figure 12-99: Example of an NRZI sequence

- Bit stuffing: a zero (0) is inserted after every 6 consecutive ones (111111). This ensures sufficient transitions to keep the phase-locked loop (PLL) synchronized and locked onto the data stream.
- Little Endian scheme: both multiple bits and multiple bytes are transmitted with the least significant bit/byte (LSB) sent first, while the most significant bit/byte (MSB) is sent last.
- HSIC (High-Speed Inter-Chip): an industry standard for USB chip-to-chip interconnection with a 2-signal (strobe, data) source synchronous serial interface, using 240 MHz DDR signaling to provide only high-speed (480 Mbps) data rate.

Table 12-22: Simplified symbolic representation from the USB standard

Bus State	Protocol	Levels
Differential **"1"**		D+ High, D- Low
Differential **"0"**		D+ Low, D- High
Data **"J"** State	Low speed	D+ Low, D- High (differential **"0"**)
	Full speed	D+ High, D- Low (differential **"1"**)
Data **"K"** State	Low speed	D+ High, D- Low (differential **"1"**)
	Full speed	D+ Low, D- High (differential **"0"**)

In the example in Table 12-22, High speed can be assumed to be like Full speed. For a complete overview of bus states and for an electrical definition of High and Low speed, refer to chapter 7 of the USB specification.

Data transfer

In contrast to a conventional bus, USB is more like a network protocol, using target addresses and endpoints. However, USB features a bus master, called the host. It transmits packets of data to all devices or hubs connected to the host (or connected to a device or hub, which is in turn connected to the host). All connected devices receive each data packet, but disregard it unless it carries the correct address. In reply, only the addressed device (one at a time) can send data upstream, to the host.

The USB architecture supports four different kinds of data transfer: control, bulk, interrupt or isochronous transfer. For example, a "bulk OUT" transfer (from host to device) would look like this:

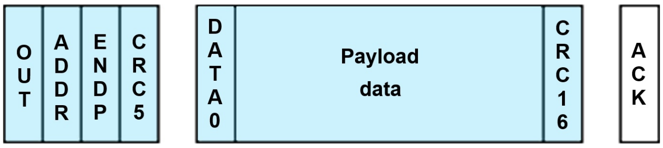


Figure 12-100: Example of a bulk OUT transfer. Blue: host speaks, white: device speaks

Packet structure

All packets must start with a **SYNC** field, also called **SOP** (start of packet), which indicates data transmission. It consists of "KJ" pairs, followed by one "KK". At low speed and full speed (USB 1.x), it is 8 bits long, encoded as "KJKJKJKK". At high speed (USB 2.0 and HSIC), it is up to 32 bits long, encoded as "KJKJKJKJKJKJ...KK".

The SYNC field is used to synchronize the clock of the receiver with that of the transmitter. The final 2 bits ("KK") indicate where the PID fields starts.

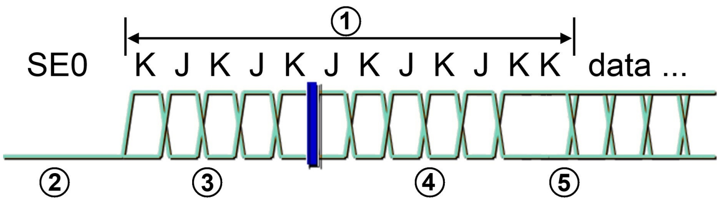


Figure 12-101: Start of a high-speed packet: after an SE0, the packet starts with a SYNC field (or SOP), followed by the data packet

SE0 = Single-ended zero, both D+ and D- wire are on low level
1 = High-speed SYNC field (or SOP), length up to 32 bits
2 = Receiver-squelched differential envelope, below 100 mV
3 = Differential envelope exceeds 150 mV, reception enabled with 4 symbol times
4 = Clock recovery must lock in time to detect end of SYNC
5 = End of SYNC detected, receiver begins data recovery

After the SYNC (or SOP), USB data packets consist of (in this order):

- **PID** (packet identifier): specifies the type of packet. Examples are:
  - Token packets (IN, OUT, SETUP) are always the first packet in a USB transaction, containing the address of the device and its endpoint.
  - Token SOF packets (start of frame) are there to schedule a data transfer.
  - Handshake packets provide the status of the transaction.
  - Data packets are dedicated for the data transfer payload.
  - Various special packets provide special characteristics: for instance, SPLIT packets are used to communicate with low or full speed devices on a high-speed link.

The PID's length is 8 bit, beginning (LSB) with the fields PID0, PID1, PID2 and PID3. Those 4 bits are repeated in PID4, PID5, PID6 and PID7, but in complementary form (for integrity checking). The PID is the only part of a packet not to be covered by the CRC (see below). Packets with invalid or non-supported PID fields are rejected. For valid PIDs, see Table 12-23.

- **ADDR** (address): contains the USB address of the target device. The length of this field is 7 bits, resulting in addresses 0 through 127 (with 0 reserved as the default address after reset).
- **ENDP** (endpoint): number of the endpoint. The length of this field is 4 bits, resulting in a maximum of 16 different endpoints.
- **Frame number**: this field with a length of 11 bits specifies the current frame number 0 to 2047. It is only sent in SOF packets.
- **Data field** (or "payload data"): the length of this field is not fixed, it can contain between 0 bytes and 1024 bytes.
- **CRC** (cyclic redundancy check): Finally, the CRC field ensures that the received data is correct. If an error is detected in the CRC calculation, the packet is rejected. Packets (in this case: IN, OUT, SOF, SETUP, PING, SPLIT) are followed by a CRC5 (which is 5 bits long) over ADDR and ENDP (or the frame number field, in case of SOF); data packets (DATA0, DATA1, MDATA, DATA2) are followed by a CRC16 (which is 16 bits long) over the data field. No CRC is used for handshake packets.
- **EOP** (end of package): the devices determine the length of the data field with the EOP signal, which indicates the end of a packet. In low and full speed, it consists of two SE0 bits, followed by a J state. For high speed, the current state is inverted and kept for 7-bit times, leading to a Bitstuff error that is used as an EOP indicator. This corresponds to the bit sequence 01111111 ahead of NRZI; for the SOF token, the bit sequence 0111111111111111111111111111111111111111 is transmitted (five NRZI bytes), which is used to recognize disconnects (since high-speed devices normally do not apply voltage to D+/D-). If no device is present, a higher voltage is measured. After that, the idle state (SE0) is occupied.

Table 12-23: Valid PIDs for different packet types

Packet type	Name	PID value	Meaning
Token	OUT	0xE1	Starts data transfer towards a device
	IN	0x69	Starts data transfer towards the host
	SOF	0xA5	Indicates start of frame
	SETUP	0x2D	Starts a setup transfer and sends information on this to the device
Data	DATA0	0xC3	Data packet with data-toggle bit 0
	DATA1	0x4B	Data packet with data-toggle bit 1
	DATA2	0x87	Data packet for high speed IN isochronous transfers that require high bandwidth
	MDATA	0x0F	Data packet for high-speed OUT isochronous transfers that require high bandwidth
Handshake	ACK	0xD2	Acknowledgement of a packet received without error

Packet type	Name	PID value	Meaning
	NAK	0x5A	Data not accepted, typically equivalent with some type of EAGAIN, meaning that the data should be resent later
	STALL	0x1E	A severe error has occurred, the target endpoint cannot be addressed until it is explicitly cleared again
	NYET	0x96	Only used in high-speed transfers, meaning ACK, but in the next inter-val no data can be received, therefore the host should first apply a PING
Special	PRE	0x3C	Starts a low speed transfer via a full speed bus
	ERR	0x3C	Indicates an error in an SPLIT transaction (using the same PID as PRE, however, these cannot be mistaken for each other)
	SPLIT	0x78	Starts an SPLIT transaction (thus: a low speed or full speed transfer via a high-speed bus)
	PING	0xB4	Used for monitoring high-speed data flow

- Token packets (IN, OUT, SETUP) and PING packets have the following format:

SYNC	PID	ADDR	ENDP	CRC5	EOP

- Start of frame packets (SOF) have the following format:

SYNC	PID	Frame number	CRC5	EOP

- Data packets have the following format:

SYNC	PID	Data	CRC16	EOP

- Handshake packets have the following format:

SYNC	PID	EOP

- SPLIT packets have the following format:

SYNC	PID	ADDR	SC	PORT	S	E	ET	CRC5	EOP

Trigger possibilities

Signals on the input channels CH1 - CH4 of the R&S RTE can be triggered by the option R&S RTE-K60. The following trigger types are available:

- ANY Packet - Packet Sync: Triggering on the first rising slope after transmission of the packet Sync. Various lengths according to standard (in USB 2.0: 32 bit)
- Any token, OUT, IN, SOF, SETUP, AND-ing with user defined PID check, address, endpoint, CRC5: For OUT, IN, SETUP, the endpoint and CRC5 follow from the bit order, therefore such patterns can be recognized.

- Data Selection: DATA0, DATA1, DATA2, MDATA, (for USB 1.x only: AND-ing with user defined PID check, payload and CRC values)
- Handshake Packet Setup: Triggering on handshake packet, trigger with specific settings: ACK, NAK, NYET, STALL or ERR handshake packet
- Protocol Error: Triggering on PID/check error, CRC5 error, CRC16 (for USB 1.x only), frame length error (for USB 1.x only)
- Bus Event: Triggering on reset, resume, or suspend

12.16.2 USB 2.0 Configuration

If you need information on how to get started with configuring the USB 2.0 setup, see Chapter 12.16.2.2, "Configuring USB Signals", on page 792. Otherwise proceed with the configuration settings.

12.16.2.1 USB 2.0 Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = *USB 2.0*

Make sure that the tab of the correct serial bus is selected on the left side.

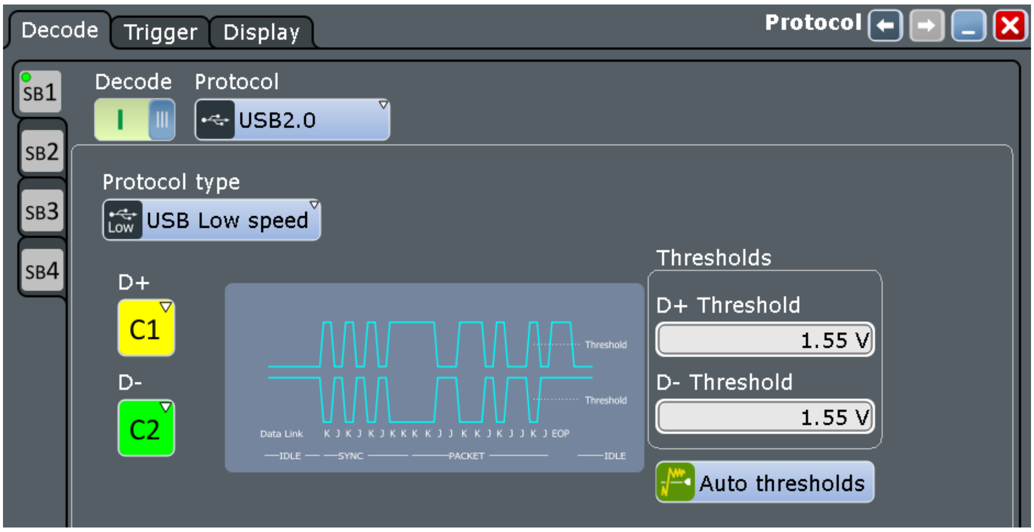


Figure 12-102: Serial bus protocol configuration dialog

For general information on how to configure protocol parameters, see also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Protocol type
Defines the USB protocol technology and transmission speed.

"USB Low speed"

Selects USB 2.0 low speed protocol (1.5 Mbit/s).

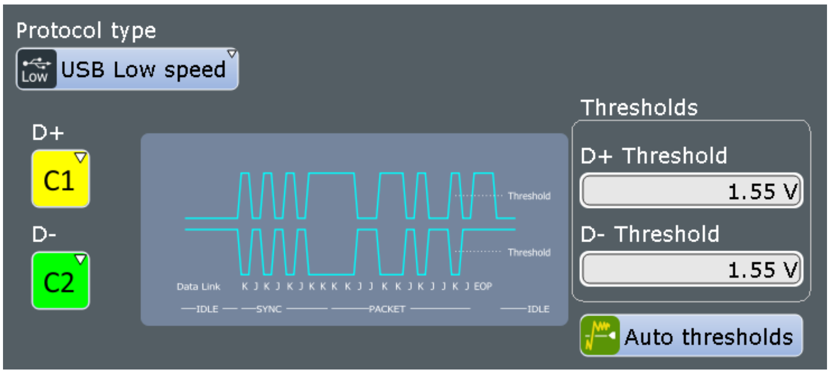


Figure 12-103: USB low speed protocol configuration

"USB Full speed"

Selects USB 2.0 full speed protocol (12 Mbit/s).

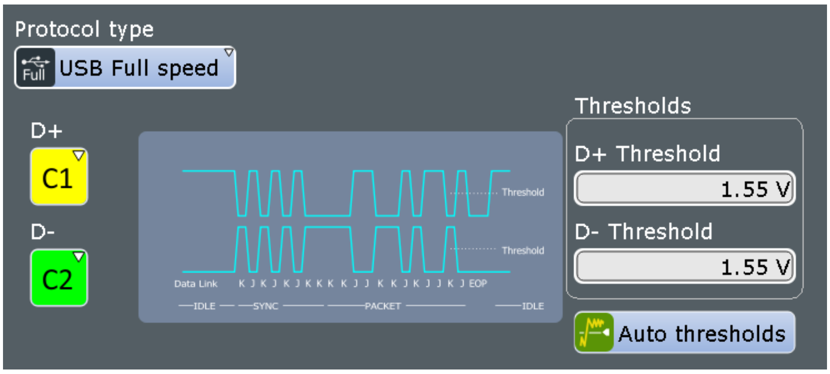


Figure 12-104: USB full speed protocol configuration

"USB High speed"

Selects USB high-speed protocol (480 Mbit/s). As the signal is differential, there is only one source and one threshold to be defined.

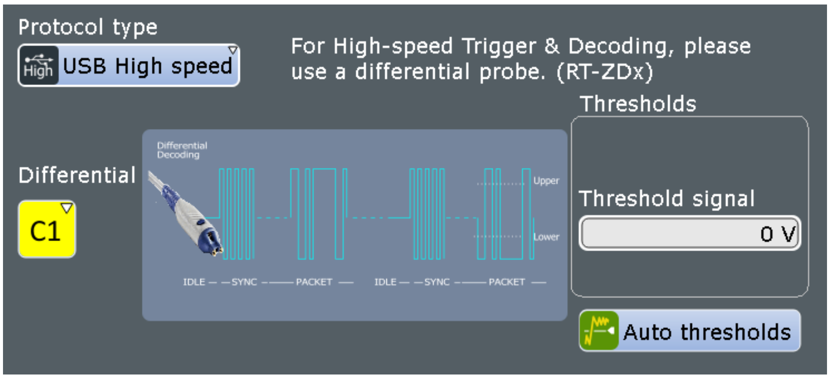


Figure 12-105: USB high-speed protocol configuration

"USB HSIC"

Selects USB high speed inter-chip (HSIC) protocol.

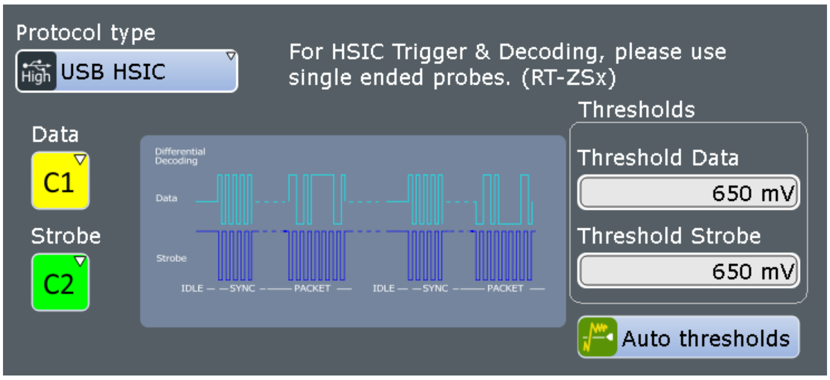


Figure 12-106: USB HSIC protocol configuration

Remote command:
BUS<m>:USB:TECHnology on page 1753

D+, D-

Define the source settings for the D+ and the D- data signals in USB low speed and USB full speed protocols. You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
BUS<m>:USB:DPLus:SOURce on page 1754
BUS<m>:USB:DMINus:SOURce on page 1754

Differential

Defines the source settings for the differential data signal (in USB high-speed protocol, only). You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
BUS<m>:USB:DIFFerential:SOURce on page 1754

Data, Strobe

Define the source settings for the data and the strobe signals in USB HSIC protocol. You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
BUS<m>:USB:DATA:SOURce on page 1754
BUS<m>:USB:STRobe:SOURce on page 1755

Thresholds

Sets the threshold value for the digitization of each signal line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, if the signal value is below the threshold, the signal state is considered low.

There are three ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
BUS<m>:USB:DPLus:THReshold on page 1755
BUS<m>:USB:DMINus:THReshold on page 1755
BUS<m>:USB:DIFFerential:THReshold on page 1756
BUS<m>:USB:DATA:THReshold on page 1756
BUS<m>:USB:STRobe:THReshold on page 1756
BUS<m>:SETReflevels on page 1383

12.16.2.2 Configuring USB Signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

Serial Bus Setup

1. Press the [PROTOCOL] key on the front panel.
2. At the left hand-side, select the vertical tab of the serial bus (SB1–SB4) you want to set up.
3. Select the "Configuration" tab.
4. Tap "Protocol" and select the protocol: "USB 2.0".
5. Optionally, you can enter a "Bus label" in the "Display" tab.
6. Switch to the "Trigger Setup" dialog, tap "Source" and select "Serial bus".

   This prevents using math waveforms, reference waveforms and tracks as channel signals.

   Note: For triggering on a serial bus, analog input channels are required.
7. Switch back to the "Serial Bus Setup" dialog.
8. Tap "Protocol type" and select the protocol type ("USB Low speed", "USB Full speed", "USB High speed", or "USB HSIC") you want to set up.
9. Depending on the protocol type, select the waveform for each of the available "D+", "D-", "Differential", "Data", and "Strobe" lines.
10. Set the logical thresholds: Either to the middle reference levels with "Auto thresholds", or enter a user-defined value directly in the "Threshold" fields.
11. In the protocol "Configuration" tab, select "Decode" to activate the decode functionality.

For details on configuration settings, see Chapter 12.16.2.1, "USB 2.0 Configuration Settings", on page 789.

12.16.3 USB 2.0 Trigger

If you need information on how to get started with triggering on USB 2.0 signals, see Chapter 12.16.3.2, "Triggering on USB 2.0", on page 802. Otherwise proceed with the USB 2.0 trigger settings.

12.16.3.1 USB 2.0 Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = USB 2.0"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected USB 2.0 protocol type and trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to data sheet.

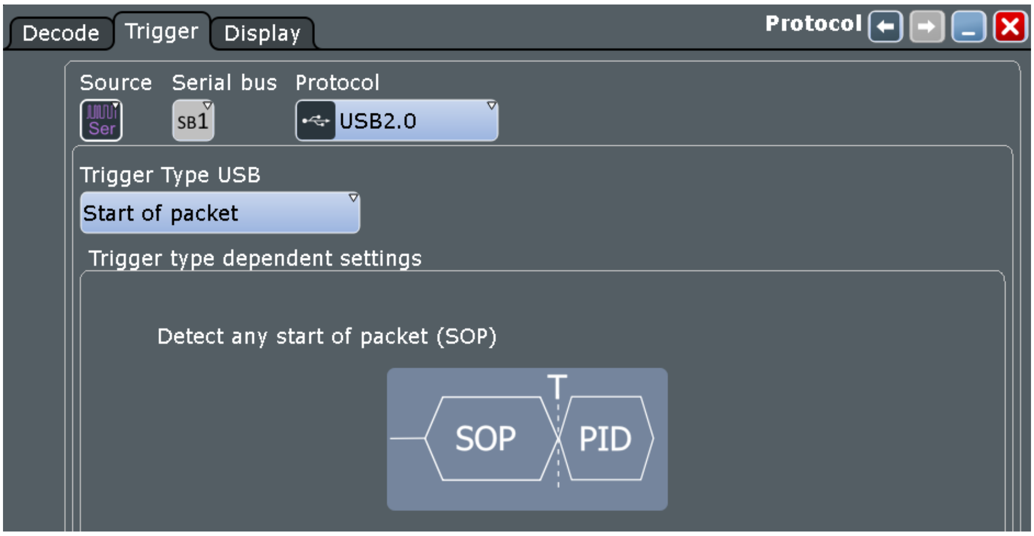


Figure 12-107: USB trigger event settings dialog

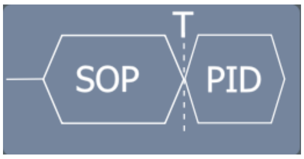
Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Trigger type USB
Selects the trigger type for USB 2.0 analysis. The available trigger types depend on the USB 2.0 protocol type that is selected in the configuration setup, see "Protocol type" on page 789.

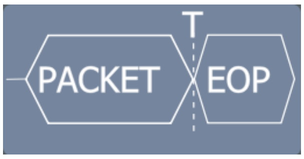
Remote command:
`TRIGger<m>:USB:TYPE` on page 1757

Start of packet ← Trigger type USB
Sets the trigger to the SOP (start of packet) field. The start of packet condition is the end of the SYNC field. The trigger instant is the end of the SOP field.



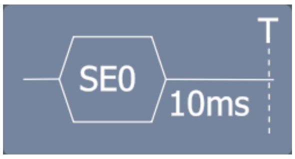
End of packet ← Trigger type USB
Sets the trigger to the EOP (end of packet) field. The trigger instant is the beginning of the EOP field.

Only available for USB low speed and USB 2.0 full speed protocol types.



Reset ← Trigger type USB
Sets the trigger to the Reset field. For more information on the reset condition, see the USB standard. The trigger instant is the end of the 10 ms period after the SE0 field.

Only available for USB low speed and USB 2.0 full speed protocol types.



Suspend ← Trigger type USB
Sets the trigger to the Suspend field. For more information on the suspend condition, see the USB standard. The trigger instant will be declared after the defined 3 ms time-out.

Only available for USB low speed and USB 2.0 full speed protocol types.

Resume ← Trigger type USB
Sets the trigger to the Resume field. For more information on the resume condition, see the USB standard. The trigger instant will be declared after the defined 20 ms timeout.

Only available for USB low speed and USB 2.0 full speed protocol types.

Token ← Trigger type USB
Sets the trigger to one out of four different token trigger types: OUT, IN, SOF, or SETUP.

See "Token" on page 795

Data ← Trigger type USB
Sets the trigger to one out of four different data trigger types: DATA0, DATA1, DATA2, or MDATA.

See "Data" on page 796

Handshake ← Trigger type USB
Sets the trigger to one out of four different handshake trigger types: ACK, NAK, STALL, or NYET.

See "Handshake" on page 797

Special PID ← Trigger type USB
Sets the trigger to one out of four different Special PID trigger types: PREamble, ERR, SPLIT, or PING.

See "Special PID" on page 798

Error condition ← Trigger type USB
Sets the trigger to one out of eight different error condition trigger types: Any error, PID error, CRC5 error, CRC16 error, Bitstuff error, Unexpected PID error, SE1 error, or Glitch error.

See "Error condition" on page 799

Token
Sets the trigger to one out of four different token types:
- "OUT"
- "IN"
- "SOF"
- "SETUP"

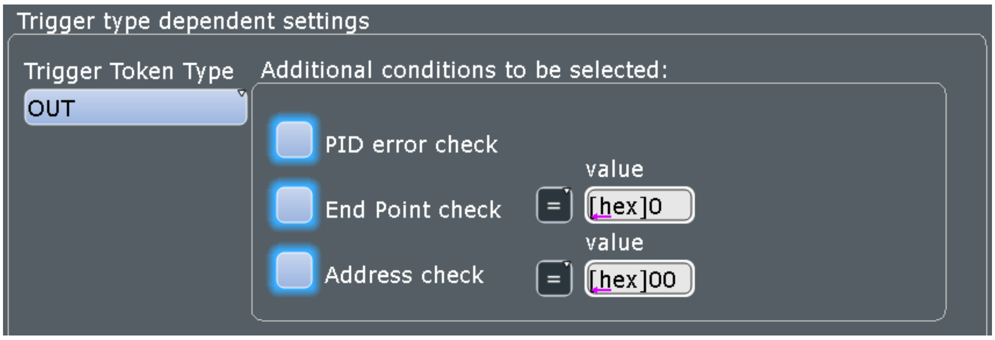
If no additional conditions are set, the trigger instant is after the PID.

If the PID error check is selected:
- If an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID
- If no PID error is found, no trigger is set, even if other selected conditions are met.

Otherwise, the trigger instant is defined by the first occurrence of any of the specified additional conditions.

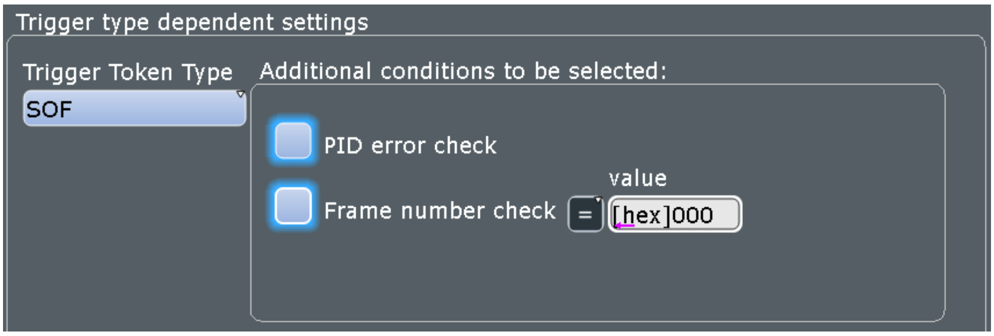
Remote command:
`TRIGger<m>:USB:TOKen` on page 1766

OUT, IN or SETUP ← Token
For the trigger token types "OUT", "IN" or "SETUP", the following conditions can be set:



You can refine the trigger condition:
- See "PID error check" on page 799
- See "End Point check" on page 799
- See "Address check" on page 800

SOF ← Token
For the trigger token type "SOF", the following conditions can be set:



You can refine the trigger condition:
- See "PID error check" on page 799
- See "Frame number check" on page 801

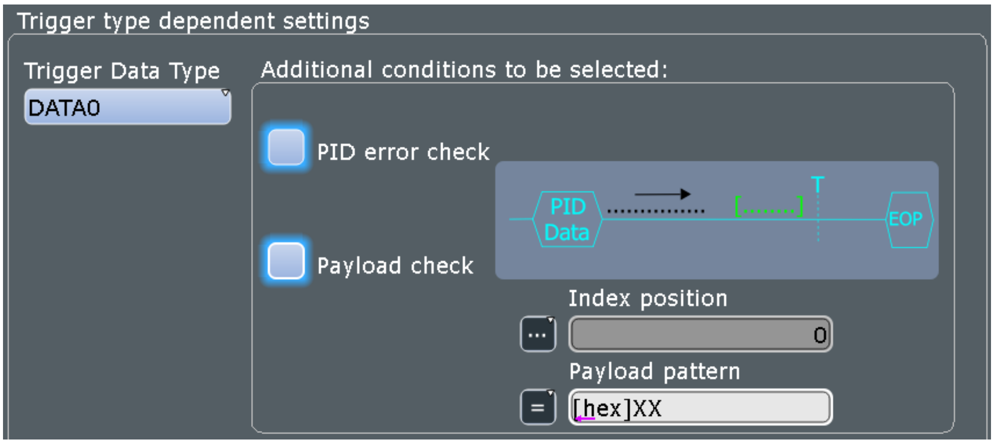
Data
Sets the trigger to one out of four different data types:
- "DATA0"
- "DATA1"
- "DATA2"
- "MDATA"

If no additional conditions are set, the trigger instant is after the PID.

If the PID error check is selected:

- If an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID
- If no PID error is found, no trigger is set, even if other selected conditions are met.

Otherwise, the trigger instant is defined by the first occurrence of any of the specified additional conditions.



You can refine the trigger condition:

- See "PID error check" on page 799
- See "Payload check" on page 800

Remote command:
`TRIGger<m>:USB:DATA` on page 1759

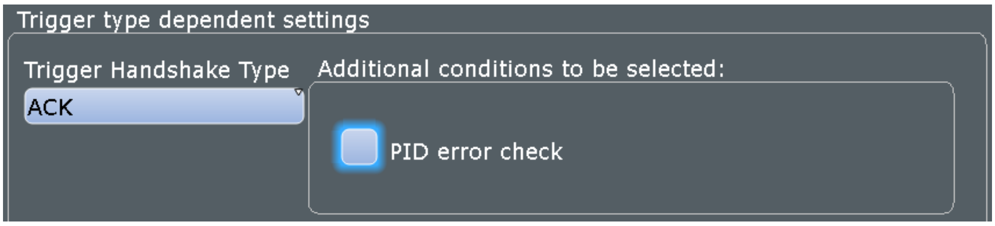
Handshake
Sets the trigger to one out of four different handshake types:

- "ACK"
- "NAK"
- "STALL"
- "NYET"

If no handshake condition is set, the trigger instant is after the PID.

If the PID error check is selected:

- If an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID.
- If no PID error is found, no trigger is set.



See "PID error check" on page 799

Remote command:
`TRIGger<m>:USB:HAND` on page 1762

Special PID

Sets the trigger to one out of four different "Special PID" types:

- "PREamble"
- "ERR"
- "SPLIT"
- "PING"

If no additional conditions are set, the trigger instant is after the PID.

If the PID error check is selected:

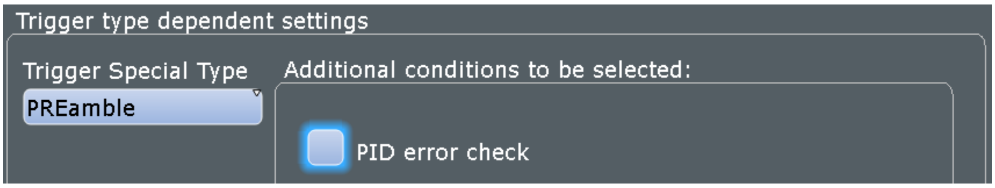
- if an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID
- if no PID error is found, no trigger is set, even if other selected conditions are met.

Otherwise, the trigger instant is defined by the first occurrence of any of the specified additional conditions.

Remote command:
`TRIGger<m>:USB:SPEC` on page 1764

PREamble or ERR ← Special PID

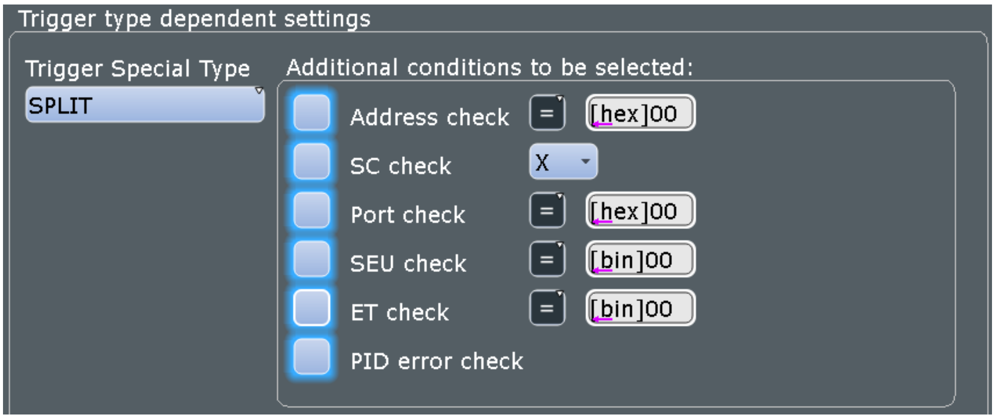
For the Trigger Special Types "PREamble" and "ERR", the following condition can be set:



See "PID error check" on page 799

SPLIT ← Special PID

For the Trigger Special Type "SPLIT", the following conditions can be set:



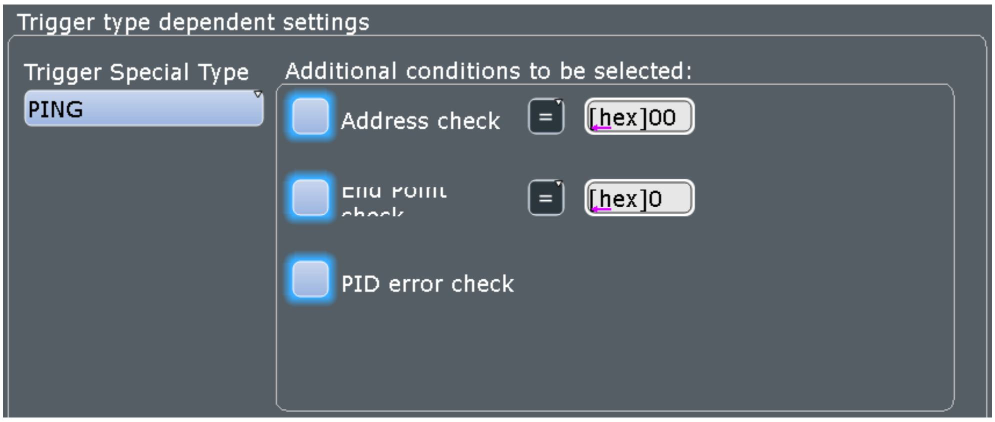
You can refine the trigger condition:

- See "Address check" on page 800
- See "SC check" on page 801
- See "Port check" on page 801
- See "SEU check" on page 801
- See "ET check" on page 802

- See "PID error check" on page 799

PING ← Special PID

For the Trigger Special Type "PING", the following conditions can be set:



You can refine the trigger condition:

- See "Address check" on page 800
- See "End Point check" on page 799
- See "PID error check" on page 799

Error condition

Sets the trigger in case of one of the following eight different error condition types.

- Any error: triggers on any of the errors listed below.
- PID error: triggers on any packet identifier error.
- CRC5 error: triggers on any CRC5 error event
- CRC16 error: triggers on any CRC16 error event
- Bitstuff error: triggers in the event of an erroneous or missing bit stuffing sequence (see USB standard).
- Unexpected PID error: triggers on any illegal PID. This is a PID that is not allowed in USB low speed and USB full speed protocols, especially PID's announcing packets such as SPLIT, DATA2, MDATA, or other noncompliant packets.
- SE1 error: triggers on the illegal bus state Single Ended 1 (SE1 = both lines high).
- Glitch error: triggers on an error in the bit period (see USB standard for the definition of glitch).

The trigger instant is the first occurrence of the specified error.

Remote command:

`TRIGger<m>:USB:ERRC` on page 1761

PID error check

Defines, whether a packet ID error check is executed or not.

Remote command:

`TRIGger<m>:USB:WPID` on page 1767

End Point check

Defines, whether an endpoint check that meets specific conditions is executed or not.

"Condition"	Defining a specific endpoint or an endpoint range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies an endpoint, or sets the start value of an endpoint range.
"Max"	Sets the end value of an endpoint range if "Condition" (TRIGger<m>:USB:ECONdition) is set to INRange or OORange.

Remote command:
TRIGger<m>:USB:WEND on page 1766
TRIGger<m>:USB:ECONdition on page 1760
TRIGger<m>:USB:EMIN on page 1761
TRIGger<m>:USB:EMAX on page 1761

Address check
Defines, whether an address check that meets specific conditions is executed or not.

"Condition"	Defining a specific address or an address range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies an address, or sets the start value of an address range.
"Max"	Sets the end value of an address range if "Condition" (TRIGger<m>:USB:ACONdition) is set to INRange or OORange.

Remote command:
TRIGger<m>:USB:WADD on page 1766
TRIGger<m>:USB:ACONdition on page 1759
TRIGger<m>:USB:AMAX on page 1759
TRIGger<m>:USB:AMIN on page 1759

Payload check
Defines, whether a payload check that meets specific conditions is executed or not.

"Condition"	Sets the operator "any" or "equal" that allows to trigger for payload data at any position or at a specified position.
"Position"	Available only if "Condition" (TRIGger<m>:USB:DPOPerator) is set to *equal*. Specifies the position in which a special data pattern is to be triggered within the payload data packet.
"Data Condition"	Sets the operator ("equal" or "unequal", TRIGger<m>:USB:DCONdition) to set a specific payload data pattern.
"Payload pattern"	Specifies the payload data pattern that is to be triggered.

Remote command:
TRIGger<m>:USB:WPAY on page 1767
TRIGger<m>:USB:DPOPerator on page 1760
TRIGger<m>:USB:DPOSition on page 1760
TRIGger<m>:USB:DCONdition on page 1760
TRIGger<m>:USB:PATT on page 1763

Frame number check
Defines, whether a frame number check that meets specific conditions is executed or not.

"Condition" Defining a specific frame number or a frame number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a frame number, or sets the start value of a frame number range.

"Max" Sets the end value of a frame number range if "Condition" (TRIGger<m>:USB:FCONdition) is set to INRange or OORange.

Remote command:
TRIGger<m>:USB:WFRN on page 1767
TRIGger<m>:USB:FCONdition on page 1762
TRIGger<m>:USB:FMIN on page 1762
TRIGger<m>:USB:FMAX on page 1762

SC check
Defines, whether a Start / Complete SPLIT transaction check is executed or not.

Remote command:
TRIGger<m>:USB:WSTC on page 1768
TRIGger<m>:USB:STCO on page 1765

Port check
Defines, whether a port check that meets specific conditions is executed or not.

"Condition" Defining a specific port number or a port number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a port number, or sets the start value of a port number range.

"Max" Sets the end value of a port number range if "Condition" (TRIGger<m>:USB:PCONdition) is set to INRange or OORange.

Remote command:
TRIGger<m>:USB:WPOR on page 1767
TRIGger<m>:USB:PCONdition on page 1763
TRIGger<m>:USB:PMIN on page 1763
TRIGger<m>:USB:PMAX on page 1764

SEU check
Defines, whether an SEU check that meets specific conditions is executed or not. S and E represent the Start and End of a start-split transaction, U represents the reserved/Unused bit of a complete-split transaction. Permissible binary SEU values are 00, 01, 10, and 11.

"Condition" Defining a specific SEU value or an SEU value range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an SEU value, or sets the start value of an SEU value range.

"Max" Sets the end value of an SEU value range if "Condition" (`TRIGger<m>:USB:SCONdition`) is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USB:WSEU` on page 1767
`TRIGger<m>:USB:SCONdition` on page 1764
`TRIGger<m>:USB:SMIN` on page 1764
`TRIGger<m>:USB:SMAX` on page 1764

ET check
Defines, whether an Endpoint Type (ET) check that meets specific conditions is executed or not.

"Condition" Defining a specific endpoint type or an ET range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an ET, or sets the start value of an ET range.

"Max" Sets the end value of an ET range if "Condition" (`TRIGger<m>:USB:TCONdition`) is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USB:WETCheck` on page 1767
`TRIGger<m>:USB:TCONdition` on page 1765
`TRIGger<m>:USB:TMIN` on page 1766
`TRIGger<m>:USB:TMAX` on page 1766

12.16.3.2 Triggering on USB 2.0

Prerequisite: A bus is configured for the USB 2.0 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to USB 2.0.
5. Tap "Trigger Type USB" and select the trigger type to be used for USB protocol analysis.
   Available trigger types depend on the USB 2.0 protocol type that is activated in "Protocol type" on page 789.

6. Depending on the selected USB 2.0 protocol type and trigger type, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 12.16.3.1, "USB 2.0 Trigger Settings", on page 793.

12.16.4 USB 2.0 Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Examples

The example in Figure 12-108 shows a simulated USB full speed message. A Token OUT packet has been decoded, followed by an ACK packet with an erroneous complementary PID value (PID Error). The next event is a PRE packet, then a DATA1 packet with two bytes of data transmitted, and with a valid CRC16. The trigger instant is at the PID of the DATA1 packet.

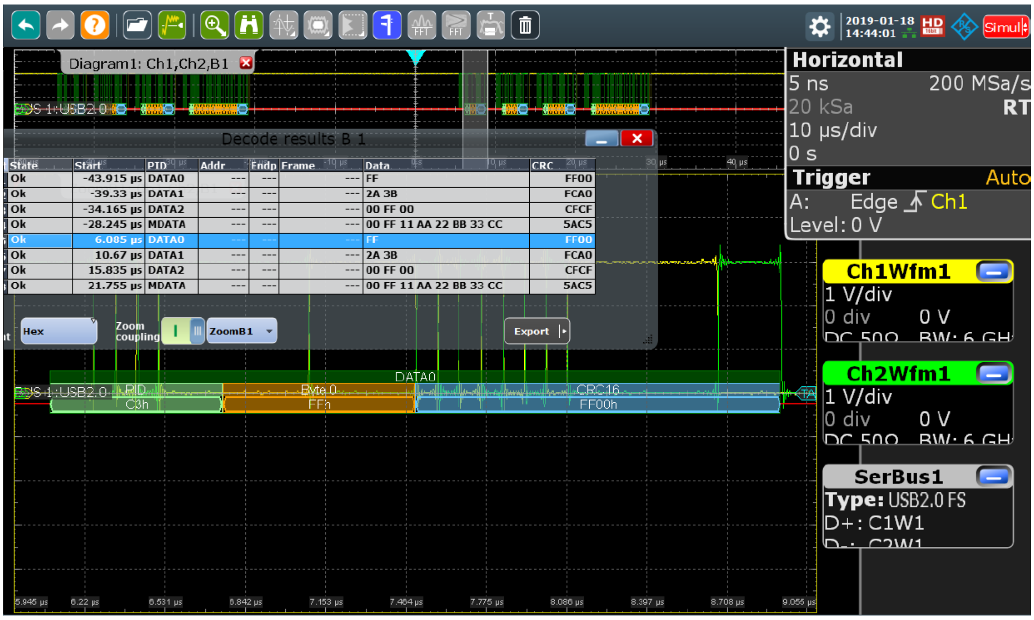


Figure 12-108: USB full speed protocol: decoded and binary signal, and decode results

green brackets [...] = start and end of packet
blue packet = packet ok
green packet = data packet ok
red packet = error condition
yellow = PID
dark blue = address
blue = endpoint
purple = CRC5/16
grey = payload data bytes

The example in Figure 12-109 shows a simulated USB high-speed message. A Token SETUP packet has been decoded, which contains a CRC5 error. The next event is a Token IN packet and an incomplete MDATA packet. Note that an incomplete packet is also decoded, as long as sample data are available. In such a case, no error is shown, since the remaining CRC16 cannot be computed. The trigger instant is on the CRC5 error.

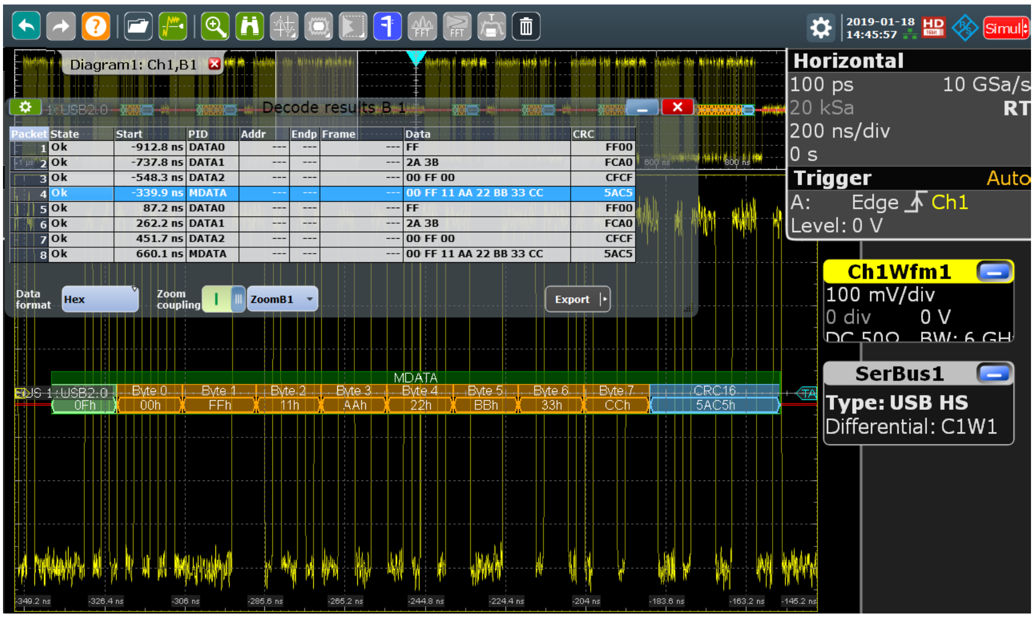


Figure 12-109: USB high-speed protocol: decoded and binary signal, and decode results

green brackets [...] = start and end of packet
blue packet = packet ok
green packet = data packet ok
red packet = error condition
yellow = PID
dark blue = address
blue = endpoint
purple = CRC5/16
grey = payload data bytes

The example in Figure 12-110 shows a simulated USB HSIC sequence, in which Data (ch1) and Strobe (ch2) are combined. A Token OUT packet, a DATA0 packet with an erroneous CRC16 value and a STALL packet have been decoded. The next events are a Token SETUP packet with erroneous CRC5 value, and a Token IN packet. In this scenario, the trigger instant is on the SOP (start of packet).

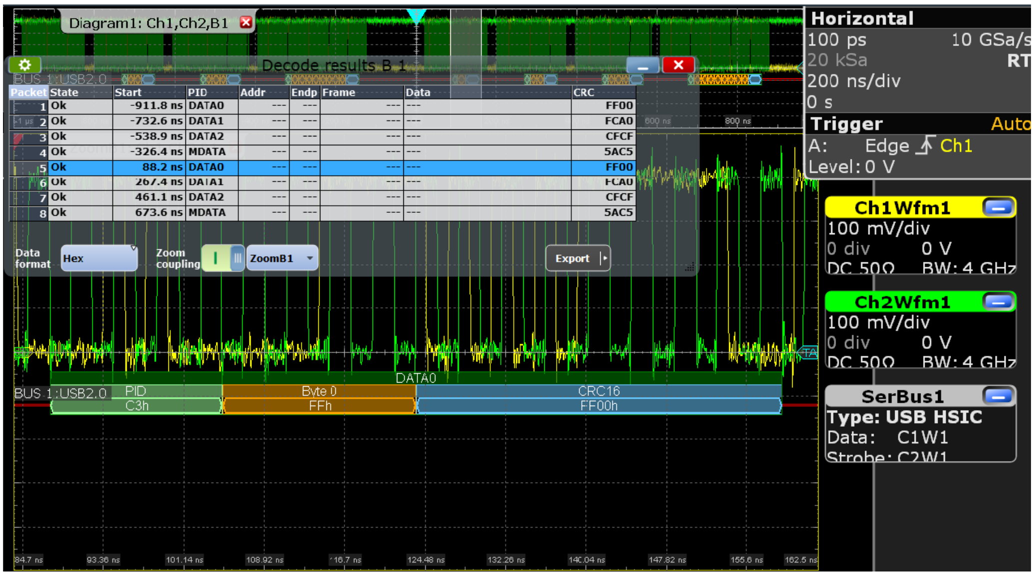


Figure 12-110: USB HSIC protocol: decoded and binary signal, and decode results

green brackets [...] = start and end of packet
blue packet = packet ok
green packet = data packet ok
red packet = error condition
yellow = PID
dark blue = address
blue = endpoint
purple = CRC5/16
grey = payload data bytes

Table 12-24: Content of the "Decode results" table in the previous figures

Column	Description
State	Overall state of the packet: either OK or the relevant error condition (CRC, glitch-ing, ...)
Start	Start time of the packet.
PID	PID type (OUT, IN, DATA0, ...)
Addr	Address of the recipient
Endp	Endpoint of the recipient
Frame	Frame number (in SOF packet)
Data	Values of the payload data bytes. The data format is selected below the table.
CRC	Either CRC5 or CRC16 (data packet PID)

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.17.3, "Decode Results", on page 1768.

12.16.5 Search on Decoded USB 2.0 Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

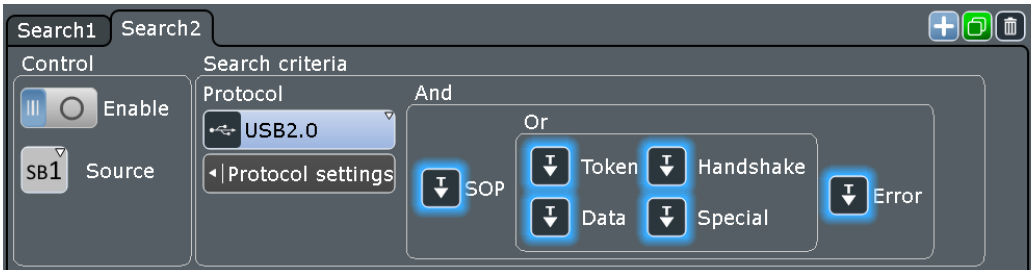
For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

If you need information on how to get started with searching USB 2.0 data, see Chapter 12.16.5.3, "Searching USB Data", on page 814. Otherwise proceed with the USB 2.0 search setup.

12.16.5.1 USB 2.0 Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for USB 2.0

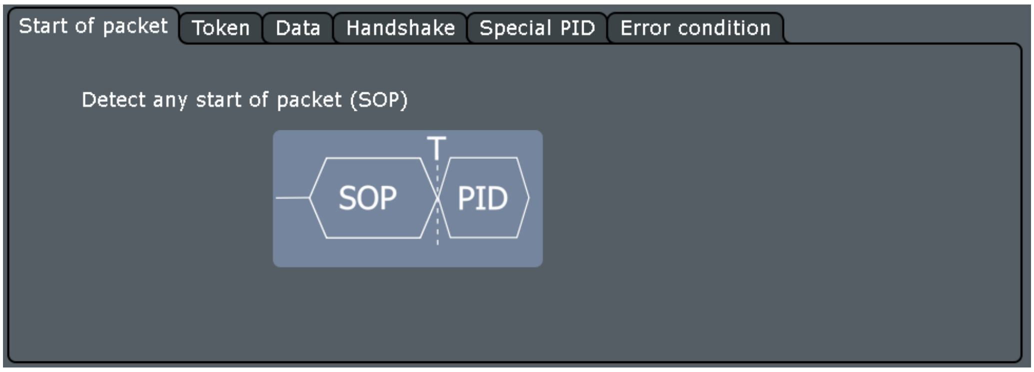
Search criteria
Use the "Search criteria" dialog to define the event types to be searched.



Individual search parameters (which do not depend on the USB protocol type and trigger settings), can be specified in the tabs below the "Search criteria" dialog.

Remote command:
`SEARch:TRIGger:USB:SSOP` on page 1783
`SEARch:TRIGger:USB:STOKen` on page 1785
`SEARch:TRIGger:USB:SDATa` on page 1782
`SEARch:TRIGger:USB:SHANdshake` on page 1783
`SEARch:TRIGger:USB:SSPE` on page 1783
`SEARch:TRIGger:USB:SERRor` on page 1783

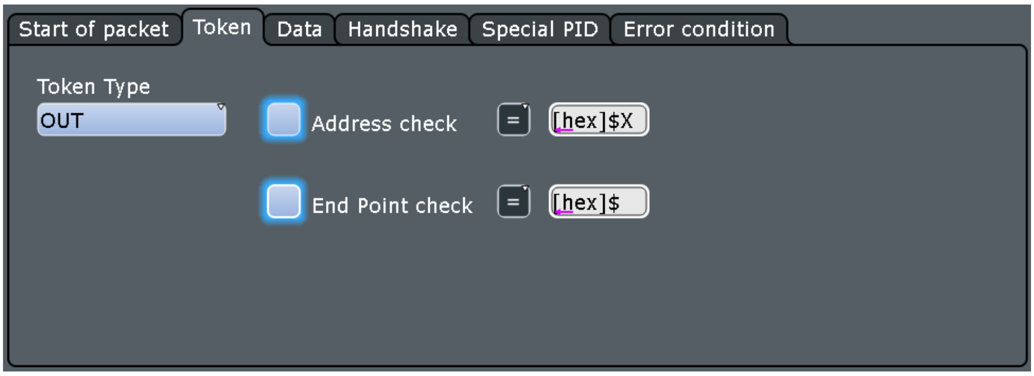
SOP
Searches for any start of packet. There are no additional parameters to be defined.



Token
Searches for four different token types: "OUT", "IN", "SOF", or "SETUP", and "Any" token.

For "Any" token, there are no additional parameters to be defined.

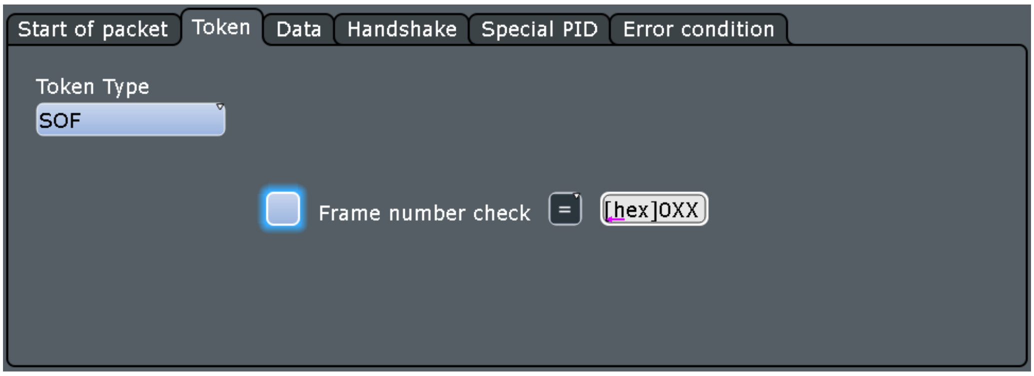
For "OUT", "IN", or "SETUP" tokens, additional search parameters are "Address check" and "End Point check".



You can refine the search condition:

- See "Address check" on page 811
- See "End Point check" on page 811

For "SOF" tokens, the additional search parameter is "Frame number check".



You can refine the search condition:

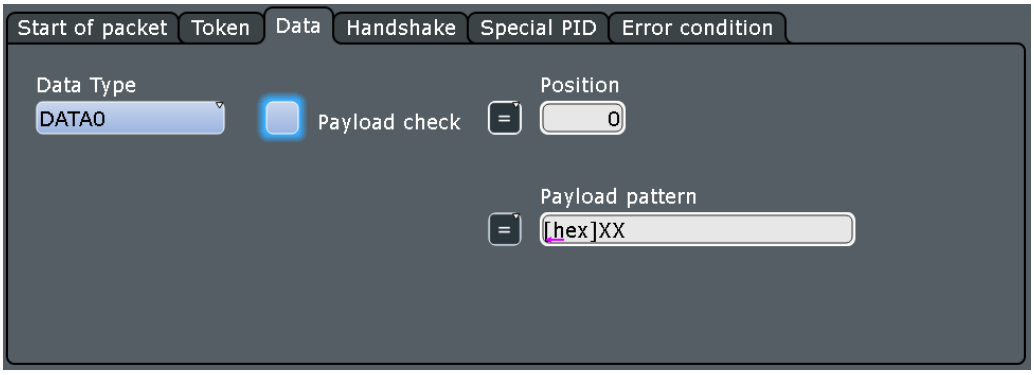
- See "Frame number check" on page 811

Remote command:

`SEARch:TRIGger:USB:TOKen` on page 1786

Data

Searches for data packets of the following types: DATA0, DATA1, DATA2, or MDATA, and "Any" data packet.



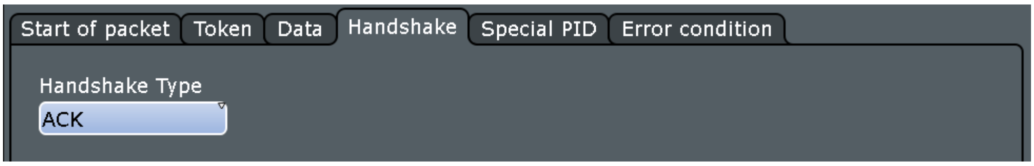
To search for payload in any data packet type, a data pattern and optionally a packet position have to be specified.

You can refine the search condition:

- See "Payload check" on page 812

Remote command:
`SEARch:TRIGger:USB:DATA` on page 1776

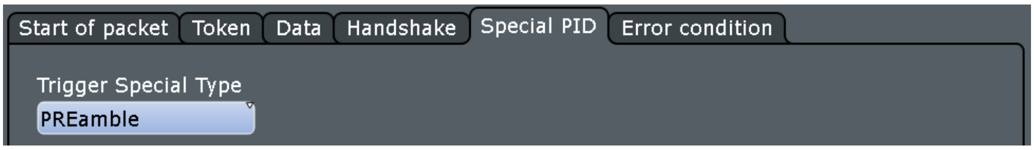
Handshake
Searches for four different handshake packet types: "ACK", "NAK", "STALL", or "NYET", and "Any" handshake packet. There are no additional parameters to be defined.



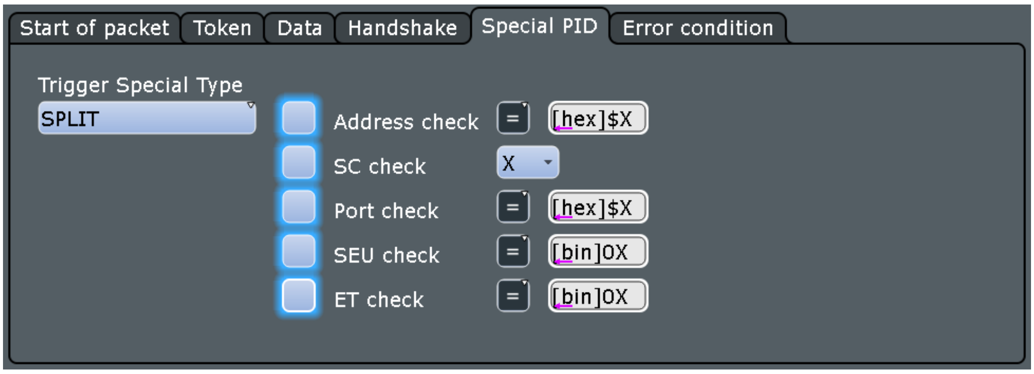
Remote command:
`SEARch:TRIGger:USB:HAND` on page 1780

Special PID
Searches for four different special packet identifier types: "PREamble", "ERR", "SPLIT", or "PING", and "Any" special PID.

For "Any", "PREamble", or "ERR", there are no additional parameters to be defined.



For "SPLIT", additional search parameters are "Address check", "SC check", "Port check", "SEU check", and "ET check".

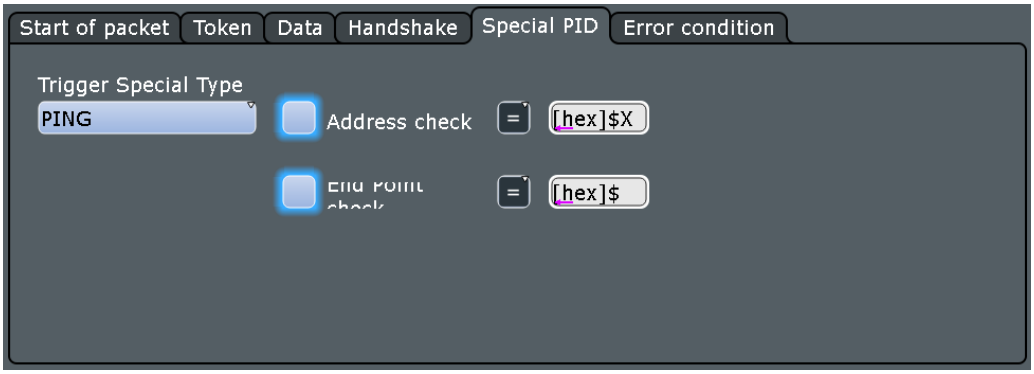


You can refine the search condition:

- See "Address check" on page 811
- See "SC check" on page 812
- See "Port check" on page 812
- See "SEU check" on page 813

- See "ET check" on page 813

For "PING", additional search parameters are "Address check" and "End Point check".



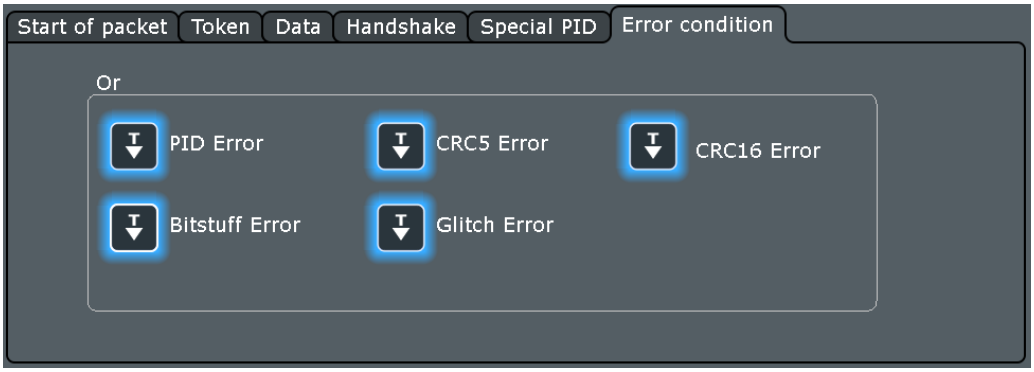
You can refine the search condition:

- See "Address check" on page 811
- See "End Point check" on page 811

Remote command:
SEARch:TRIGger:USB:SPEC on page 1784

Error condition

Searches for the following error conditions:



"PID Error" Searches for packet identifier errors.

Remote command:
SEARch:TRIGger:USB:PIDerror on page 1781

"CRC5 Error" Searches for any CRC5 error event.

Remote command:
SEARch:TRIGger:USB:CRC5error on page 1776

"CRC16 Error" Searches for any CRC16 error event.

Remote command:
SEARch:TRIGger:USB:CRC16error on page 1776

"Bitstuff Error" Searches for bitstuff errors, thus an erroneous or missing bit stuffing sequence (see USB standard).

Remote command:
SEARch:TRIGger:USB:BITSterror on page 1775

"Glitch Error" Searches for glitch errors (errors in the bit period, see USB standard for the definition of glitch).

Remote command:
SEARch:TRIGger:USB:GLITcherror on page 1779

Address check

Enables the search for an address that meets specific conditions.

"Condition" Defining a specific address or an address range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an address, or sets the start value of an address range.

"Max" Sets the end value of an address range if "Condition" (SEARch:TRIGger:USB:ACONdition) is set to INRange or OORange.

Remote command:
SEARch:TRIGger:USB:WADD on page 1786
SEARch:TRIGger:USB:ACONdition on page 1775
SEARch:TRIGger:USB:AMIN on page 1775
SEARch:TRIGger:USB:AMAX on page 1775

End Point check

Enables the search for an endpoint that meets specific conditions.

"Condition" Defining a specific endpoint or an endpoint range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an endpoint, or sets the start value of an endpoint range.

"Max" Sets the end value of an endpoint range if "Condition" (SEARch:TRIGger:USB:ECONdition) is set to INRange or OORange.

Remote command:
SEARch:TRIGger:USB:WEND on page 1787
SEARch:TRIGger:USB:ECONdition on page 1778
SEARch:TRIGger:USB:EMIN on page 1778
SEARch:TRIGger:USB:EMAX on page 1778

Frame number check

Enables the search for a frame number that meets specific conditions.

"Condition" Defining a specific frame number or a frame number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a frame number, or sets the start value of a frame number range.

"Max" Sets the end value of a frame number range if "Condition" (SEARch:TRIGger:USB:FCONdition) is set to INRange or OORange.

Remote command:
SEARch:TRIGger:USB:WFRN on page 1787
SEARch:TRIGger:USB:FCONdition on page 1779
SEARch:TRIGger:USB:FMIN on page 1779
SEARch:TRIGger:USB:FMAX on page 1779

Payload check
Enables the search for a payload data pattern that meets specific conditions.

"Condition" Sets the operator "any" or "equal" that allows to search for payload data at any position or at a specified position.

"Position" Available only if "Condition" (SEARch:TRIGger:USB:DPOPerator) is set to *equal*. Specifies the position in which a special data pattern is to be searched within the payload data packet.

"Data Condition" Sets the operator ("equal" or "unequal", SEARch:TRIGger:USB:DCONdition) to set a specific payload data pattern.

"Payload pattern" Specifies the payload data pattern that is to be searched.

Remote command:
SEARch:TRIGger:USB:WPAY on page 1787
SEARch:TRIGger:USB:DPOPerator on page 1777
SEARch:TRIGger:USB:DPOSition on page 1777
SEARch:TRIGger:USB:DCONdition on page 1777
SEARch:TRIGger:USB:PATT on page 1780

SC check
Searches for the selected Start (0) or Complete (1) SPLIT transaction endpoint, or X (do not care).

Remote command:
SEARch:TRIGger:USB:WSTC on page 1788
SEARch:TRIGger:USB:STCO on page 1784

Port check
Enables the search for a port that meets specific conditions.

"Condition" Defining a specific port number or a port number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a port number, or sets the start value of a port number range.

"Max" Sets the end value of a port number range if "Condition" (SEARch:TRIGger:USB:PCONdition) is set to INRange or OORange.

Remote command:
SEARch:TRIGger:USB:WPOR on page 1788
SEARch:TRIGger:USB:PCONdition on page 1780
SEARch:TRIGger:USB:PMIN on page 1781
SEARch:TRIGger:USB:PMAX on page 1781

SEU check

Enables the search for an SEU that meets specific conditions. (For SEU, see "SEU check" on page 801.)

"Condition" Defining a specific SEU value or an SEU value range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an SEU value, or sets the start value of an SEU value range.

"Max" Sets the end value of an SEU value range if "Condition" (SEARch:TRIGger:USB:SCONdition) is set to INRange or OORange.

Remote command:
SEARch:TRIGger:USB:WSEU on page 1788
SEARch:TRIGger:USB:SCONdition on page 1781
SEARch:TRIGger:USB:SMIN on page 1782
SEARch:TRIGger:USB:SMAX on page 1782

ET check

Enables the search for an Endpoint Type (ET) that meets specific conditions.

"Condition" Defining a specific endpoint type or an ET range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an ET, or sets the start value of an ET range.

"Max" Sets the end value of an ET range if "Condition" (SEARch:TRIGger:USB:TCONdition) is set to INRange or OORange.

Remote command:
SEARch:TRIGger:USB:WETCheck on page 1787
SEARch:TRIGger:USB:TCONdition on page 1785
SEARch:TRIGger:USB:TMIN on page 1785
SEARch:TRIGger:USB:TMAX on page 1786

12.16.5.2 USB 2.0 Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- SEARch:RESult:USB:PCOunt? on page 1793
- SEARch:RESult:USB:PACKet<m>:STATus? on page 1793

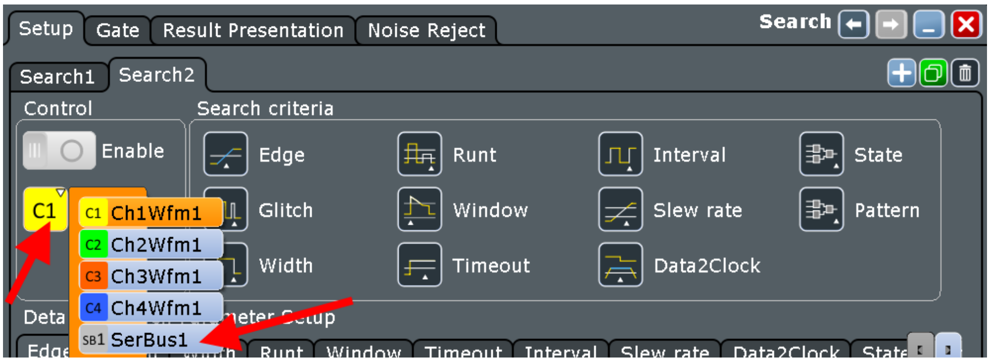
- SEARch:RESult:USB:PACKet<m>:STARt? on page 1792
- SEARch:RESult:USB:PACKet<m>:STOP? on page 1793
- SEARch:RESult:USB:PACKet<m>:ADDRess? on page 1789
- SEARch:RESult:USB:PACKet<m>:DATA? on page 1790
- SEARch:RESult:USB:PACKet<m>:CRC? on page 1789
- SEARch:RESult:USB:PACKet<m>:ENDPoint? on page 1790
- SEARch:RESult:USB:PACKet<m>:ET? on page 1790
- SEARch:RESult:USB:PACKet<m>:FRAMe? on page 1790
- SEARch:RESult:USB:PACKet<m>:PID? on page 1791
- SEARch:RESult:USB:PACKet<m>:PORT? on page 1791
- SEARch:RESult:USB:PACKet<m>:SC? on page 1792
- SEARch:RESult:USB:PACKet<m>:SEU? on page 1792

12.16.5.3 Searching USB Data

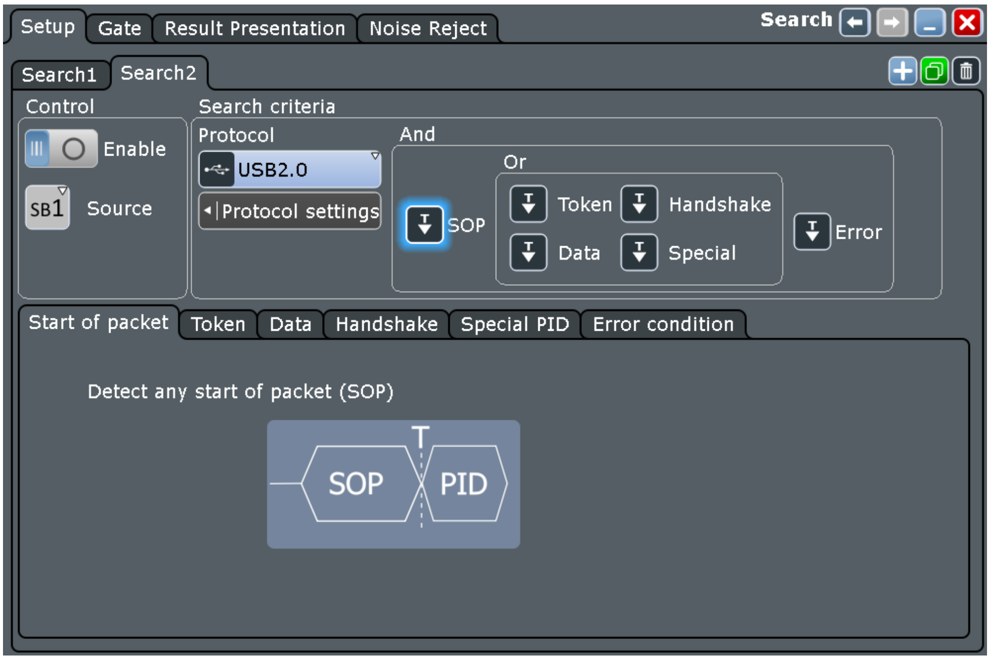
Prerequisite: A serial bus is configured for the USB signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to USB 2.0 (e.g. "SerBus1", unless already selected).



The search dialog for USB 2.0 protocol analysis is opened.



4. Specify search criteria according to Chapter 12.16.5.1, "USB 2.0 Search Setup", on page 806.
5. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs a USB 2.0 decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.17 USBPD (Option R&S RTE-K63)

R&S RTE-K63 is a firmware option that enables the R&S RTE to analyze Universal Serial Bus Power Delivery (USBPD) signals.

For analysis, USBPD encoded signals can be triggered, decoded and searched.

This chapter describes:

- The USB Power Delivery Protocol....816
- USBPD Configuration....817
- USBPD Trigger....820
- USBPD Decode Results....823
- Search on Decoded USBPD Data....827

12.17.1 The USB Power Delivery Protocol

The requirements on the USB have changed in the last years with the need of providing power through the USB port additionally to the data transfer. The USBPD specification aims to define standard for optimizing the power usage through the USB for the needs of the users.

USBPD characteristics

Main characteristics of USBPD are:
- Power direction is not fixed
- Negotiation of required power amount between devices
- Alternate modes can be defined through vendor defined messages, which allows for USB connector pins to be used for purposes other than USB

Message types

In the USBPD protocol, a power delivery connection can be made between a port that supplies power (source) and a port that consumes power (sink). They communicate with each other through messages. The USBPD specification defines three message types:

- Control messages: 16-bit messages used to control the messages between the port partners or transfer messages with no extra data. A control message consists of a message header and a CRC.
- Data messages: 48 bit to 240 bit messages used to transfer information between port partners. A data message consists of a message header and several data objects. The information that a data object carries is defined by the message type of the message header, see Table 12-25.
- Extended messages: can have a different length up to the defined maximum length of an extended message. It is used to transfer information between port partners. The information that the extended message carries is defined by the message type of the message header, see Table 12-25.

Frame packet types

All frame types are listed below. The frames listed above consist of individual sets of fields. Some frames only contain one field, others are much longer.

The frame types are sorted according to the message type.

Table 12-25: Frame packet types

SCPI	Description	Message type
ALRT	Alert message	Data
BATT	Battery status	Data
BIST	Built in self-test	Data
RQST	Request	Data
SINK	Sink capabilities message	Data

SCPI	Description	Message type
SRC	Source capabilities message	Data
VEND	Vendor defined message	Data
CTRL	Control	Control
DATA	Data	Data
LOWP	Low power	Low power
TEST	Test frame	Test
RESet	Reset frame	Reset
XBAC	Battery capabilities	Extended
XFRS	Firmware update response	Extended
XFRQ	Firmware update request	Extended
XGBC	Get battery cap	Extended
XGMI	Get manufacturer info	Extended
XGBS	Get battery status	Extended
XMFI	Manufacturer info	Extended
XMSG	Message	Extended
XSRC	Sources capabilities message	Extended
XSRS	Security response	Extended
XSRQ	Security request	Extended
XSTA	Status	Extended

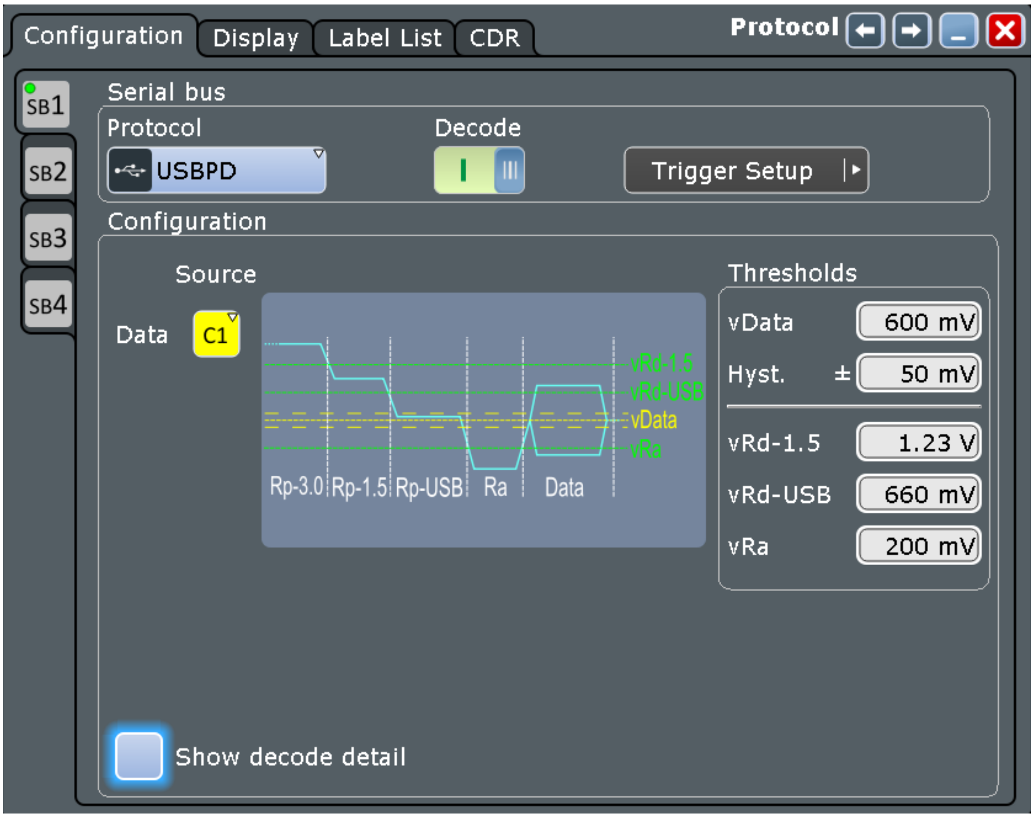
More information on the USBPD protocol, including all specifications down to the field contents, is available in the "Universal Serial Bus Power Delivery Specification" documentation. Refer to the online resources at www.usb.org.

12.17.2 USBPD Configuration

12.17.2.1 USBPD Configuration Settings

Access: [PROTOCOL] key > "Decode" tab > "Protocol" = "USBPD"

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Data

Defines the source settings for the data signal. You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
`BUS<m>:USBPd:SOURce` on page 1794

vData

Sets the threshold value of the data.

Remote command:
`BUS<m>:USBPd:THReshold` on page 1795

Hyst

Sets a value for the hysteresis of the data.

Remote command:
`BUS<m>:USBPd:HYSTeresis` on page 1794

Current advertisment thresholds

The signal level provides information about the current advertisement between the bursts. These thresholds determine the levels at which the current advertisement modes are defined.

"vRd-1.5" Sets the threshold at USB Type-C current of 1.5 A.

"vRd-USB" Sets the threshold at default USB Type-C current.

"vRa" Sets the threshold for the low current.

Remote command:
BUS<m>:USBPd:THRBottom on page 1795
BUS<m>:USBPd:THRMid on page 1795
BUS<m>:USBPd:THRTop on page 1796

Show decode detail
If enabled, the data words are broken down into sub-frames. If not enabled the data words are displayed as 32-bit data words.

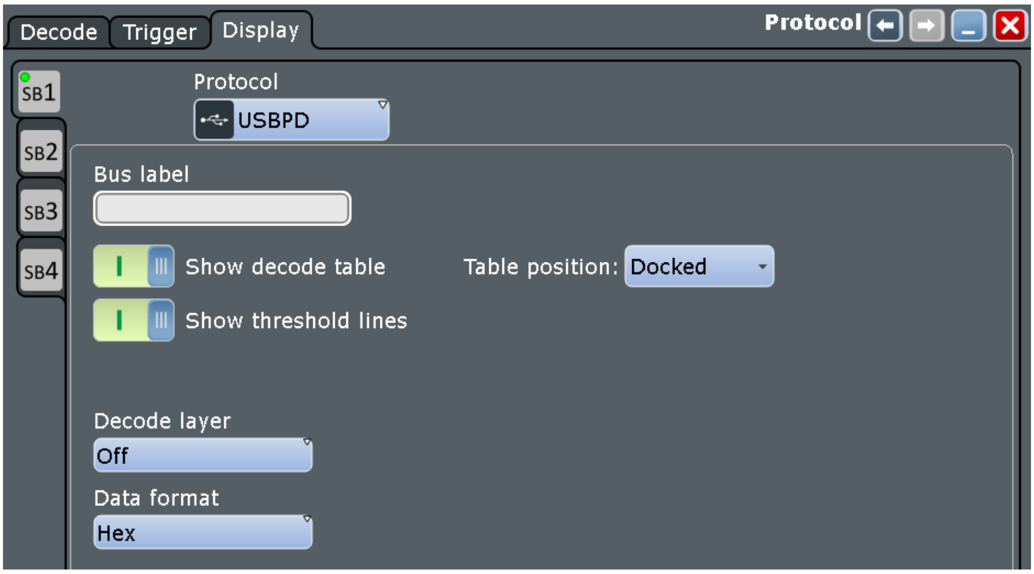
If the "Show decode detail" is enabled, you cannot do a search and trigger on USBPD frames.

Remote command:
BUS<m>:USBPd:DETail on page 1794

12.17.2.2 Display Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol = USBPD " > "Display" tab

To enhance the decode possibilities of the USBPD protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Common display settings are explained in Chapter 12.1.3, "Display", on page 475.

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Final" ...

"Edges"	...
"Bits"	...
"4b5b Symbols"	...

12.17.2.3 Configuring the USBPD Signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Configuration" tab.
4. Tap the "Protocol" button and select the protocol: "USBPD".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Select the data source.
7. Enter the "Thresholds".
8. Enable "Decode", if available.

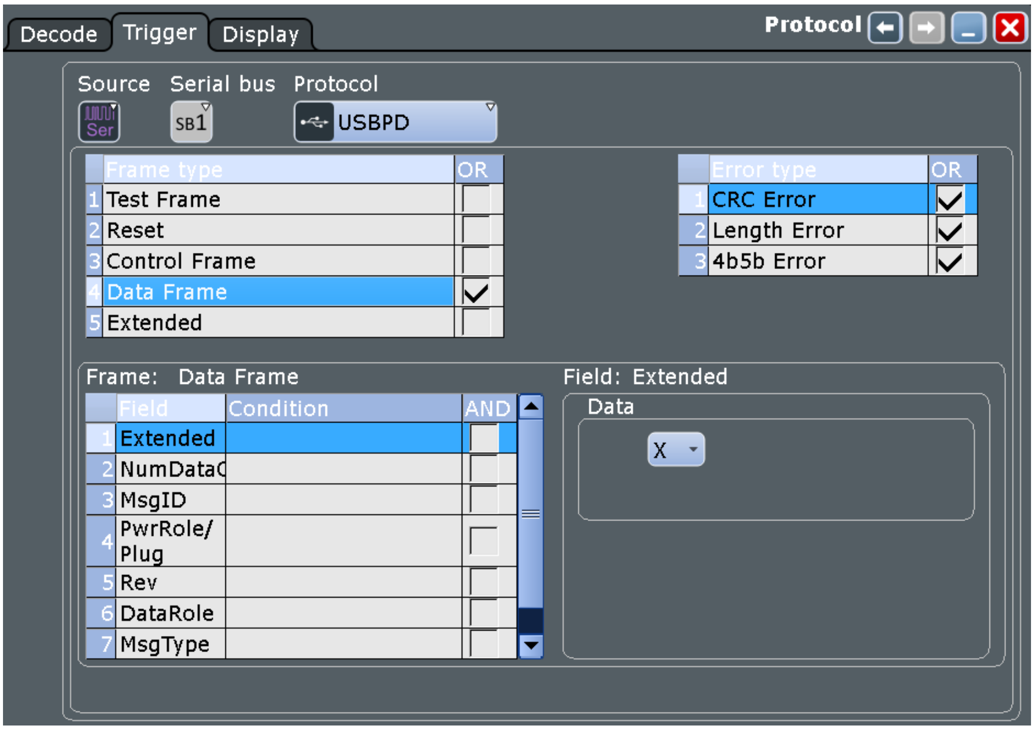
12.17.3 USBPD Trigger

If you need information on how to get started with triggering on USBPD signals, see Chapter 12.17.3.2, "Triggering on USBPD", on page 823. Otherwise proceed with the USBPD trigger settings.

12.17.3.1 USBPD Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = USBPD"

In this section, all trigger settings for USBPD are described. The user interface of the instrument guides you through the trigger setup.



Make sure that:

- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Frame Type

Selects the frame type for the USBPD analysis. For the available packet types, see "Frame packet types" on page 816.

You can define individual checking parameters for the fields listed in the "Trigger type dependent settings".

To specify these parameters, select a field from this list and define the data and/or index operators and values, or the bit state.

The trigger instant is the last criterion that is fulfilled.

Remote command:

`TRIGger<m>:USBPd:FRENable` on page 1798

`TRIGger<m>:USBPd:FRAMe<n>:ENABle` on page 1798

Field name / Condition Summary / Enable

This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Data", "BitState" and/or "Index" dialog (whichever applies). The condition is only applied, and the "Condition Summary" is only shown in the table, if "Enable" is checked.

For an overview of frames and fields, see "Frame packet types" on page 816.

Remote command:
`TRIGger<m>:USBPd:FIENable` on page 1799
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle` on page 1799

Error > Item name /Enable

The table lists the error types you can trigger on: "CRC Error", "Length Error", "4b5b Error". Enable any error that you want to trigger on in the "OR" column.

Remote command:
`TRIGger<m>:USBPd:ERENable` on page 1798
`TRIGger<m>:USBPd:ERRor<n>:ENABle` on page 1798

Data

Defines for the selected field, how a data check is executed.

"Condition"	Defining specific data or a data range requires to set the operator to one of the following conditions: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies data or sets the start value of a data range.
"Max"	Sets the the end value of a data range, if "Condition" is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USBPd:DOPerator` on page 1800
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator` on page 1800
`TRIGger<m>:USBPd:DMAX` on page 1800
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX` on page 1800
`TRIGger<m>:USBPd:DMIN` on page 1800
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN` on page 1800

Data BitState

Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (don't care).

Remote command:
`TRIGger<m>:USBPd:BIT` on page 1799
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT` on page 1799

Index

Defines for the selected field, how an index check is executed.

"Condition"	Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.
"Min"	Specifies the index or sets the start value of an index range.
"Max"	Sets the the end value of an index range, if "Condition" is set to `INRange`.

Remote command:
`TRIGger<m>:USBPd:IOPerator` on page 1802
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator` on page 1802
`TRIGger<m>:USBPd:IMAX` on page 1801
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX` on page 1801
`TRIGger<m>:USBPd:IMIN` on page 1801
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN` on page 1801

12.17.3.2 Triggering on USBPD

Prerequisite: A bus is configured for the USBPD signal to be analyzed.

In order to be able to trigger on an USBPD data, the "Show decode detail" field in the "Configuration" tab of the protocol setup should be disabled.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to USBPD, e.g.:
   The "Protocol" selection is then automatically set to "USBPD".
5. If you trigger on errors, enable the error types you want to find.
6. Select the frame types you want to trigger on.
7. For some frame types, you can define the frame fields.
   For information on how to proceed with the configuration settings, see Chapter 12.17.3.1, "USBPD Trigger Settings", on page 820.

12.17.4 USBPD Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Examples

The example in USBPD protocol: decoded and binary signal shows a simulated USBPD message. Among the long line of decoded frames, the zoom has selected a "Data" frame.

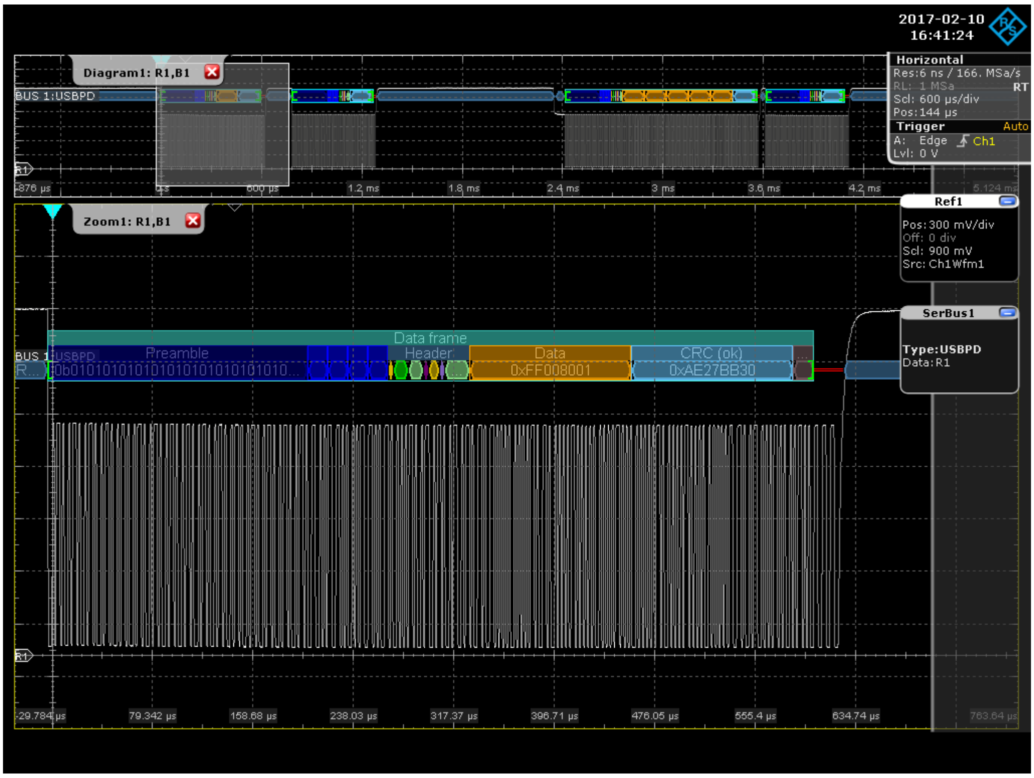


Figure 12-111: USBPD protocol: decoded and binary signal

Green brackets [...]	= Start and end of frame
Dark green field	= Header
Light blue fields	= CRC checksum
Orange fields	= Data bits

The example in USBPD protocol: decoded and binary signal, and decode results shows the same simulated USBPD message as in Figure 12-111, but with overlaid decode results (upper table, showing frames) and decode results details (table below, showing decoded fields of the selected frame).

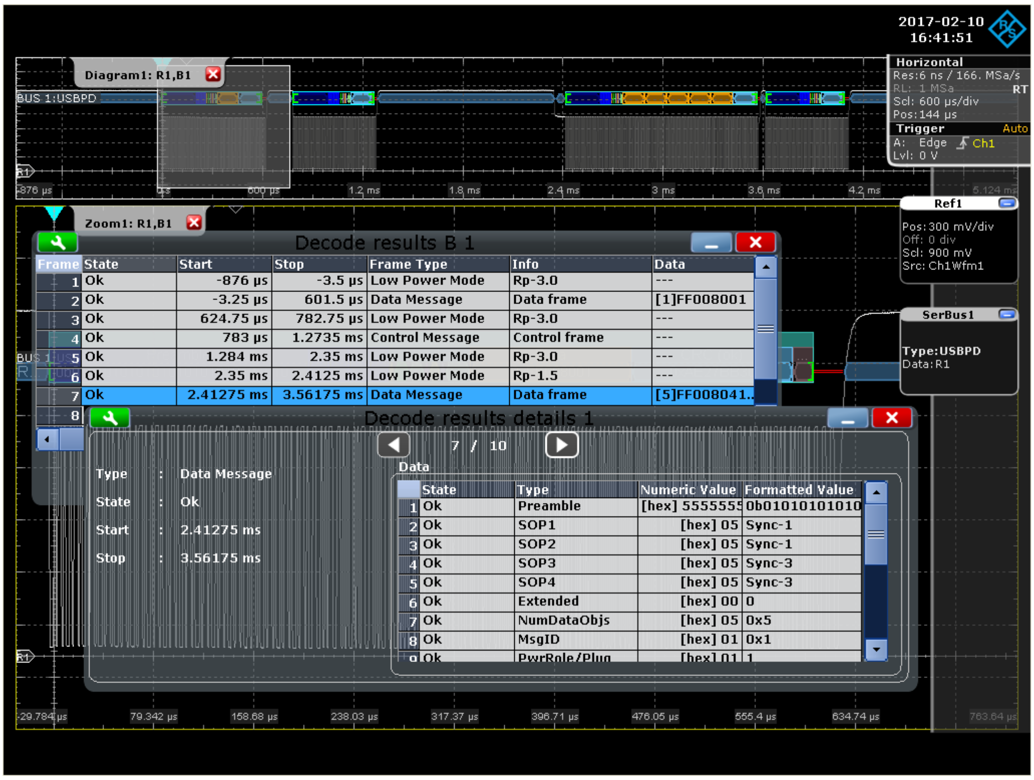


Figure 12-112: USBPD protocol: decoded and binary signal, and decode results

Table 12-26: Content of the "Decode results" table

Column	Description
State	Overall state of the frame
Start	Time of frame start
Stop	Time of frame stop
Frame Type	Type of the frame
Info	Information about the frame
Data	Data of the frame

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

Table 12-27: Content of the "Decode results details" table

Column	Description
State	Overall state of the field
Start	Start time of the field
Stop	Stop time of the field
Type	Type (name) of field

Column	Description
Numeric value	The numeric value of the field
Status	Status of the field: OK or error

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- BUS<m>:FORMat on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- BUS<m>:ZCOupling on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.

3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.18.3, "Decode Results", on page 1802.

- BUS<m>:USBPd:RESult:FCOunt? on page 1802
- BUS<m>:USBPd:RESult:FRAMe<n>:DATA? on page 1803
- BUS<m>:USBPd:RESult:FRAMe<n>:INFO? on page 1805
- BUS<m>:USBPd:RESult:FRAMe<n>:STARt? on page 1806
- BUS<m>:USBPd:RESult:FRAMe<n>:STATe? on page 1806
- BUS<m>:USBPd:RESult:FRAMe<n>:STOP? on page 1806
- BUS<m>:USBPd:RESult:FRAMe<n>:TYPE? on page 1807
- BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL? on page 1803
- BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus? on page 1803

- BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt? on page 1804
- BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP? on page 1804
- BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE? on page 1805
- BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL? on page 1805

12.17.5 Search on Decoded USBPD Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

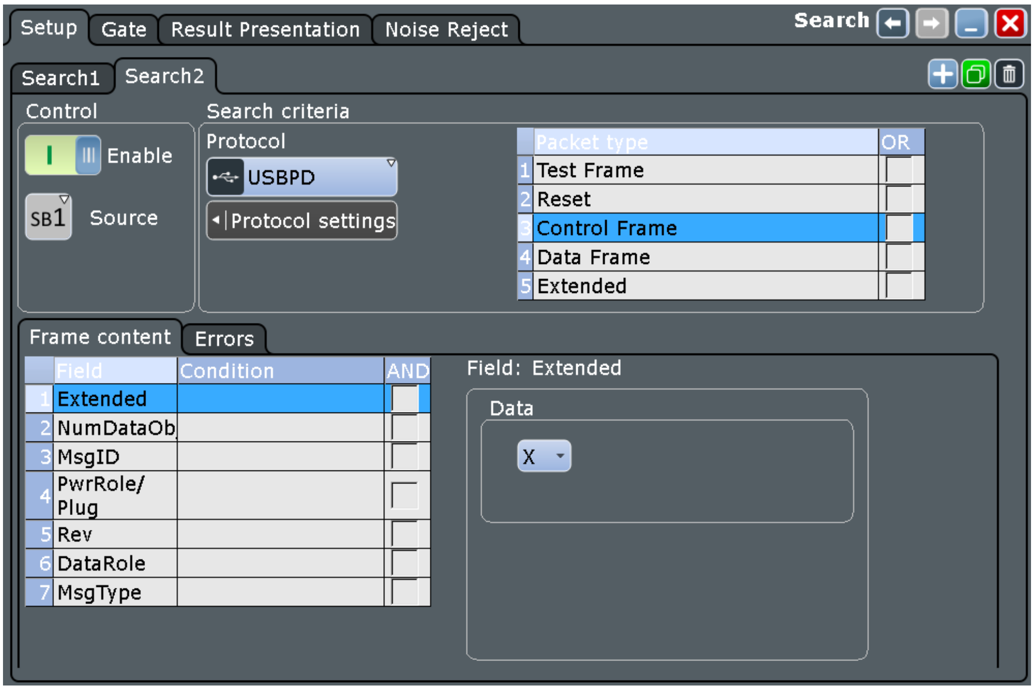
For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

If you need information on how to get started with searching USBPD data, see Chapter 12.17.5.3, "Searching USBPD Data", on page 830. Otherwise proceed with the USBPD search setup.

12.17.5.1 USBPD Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for USBPD

Search criteria
Use the "Search criteria" dialog to define the frame type in which data is to be searched.



You can define individual search parameters for individual fields in the table below the "Search criteria" dialog. To specify these parameters, select a field in this table and define the data and/or index operators and values, or the bit state.

For a description of how to set the search conditions, see Chapter 12.17.3.1, "USBPD Trigger Settings", on page 820.

Remote command:
`SEARch:TRIGger:USBPd:ERENable` on page 1808
`SEARch:TRIGger:USBPd:ERRor<m>:ENABle` on page 1808
`SEARch:TRIGger:USBPd:BIT` on page 1809
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT` on page 1809
`SEARch:TRIGger:USBPd:DMAX` on page 1810
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX` on page 1810
`SEARch:TRIGger:USBPd:DMIN` on page 1810
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN` on page 1810
`SEARch:TRIGger:USBPd:DOPerator` on page 1811
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator` on page 1811
`SEARch:TRIGger:USBPd:IMAX` on page 1811
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX` on page 1811
`SEARch:TRIGger:USBPd:IMIN` on page 1812
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN` on page 1812
`SEARch:TRIGger:USBPd:IOPerator` on page 1812
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator` on page 1812

12.17.5.2 USBPD Search Results

To get search results, "Enable" the search in the "Control" section of the "Search Setup" dialog. You can minimize, shift or close the search dialog to better see the "Search Results" table.

If the "Show decode detail" field in the "Configuration" tab of the protocol setup is enabled, then the "Enable" search button is disabled. Disable "Show decode detail" first, to be able to start the search.

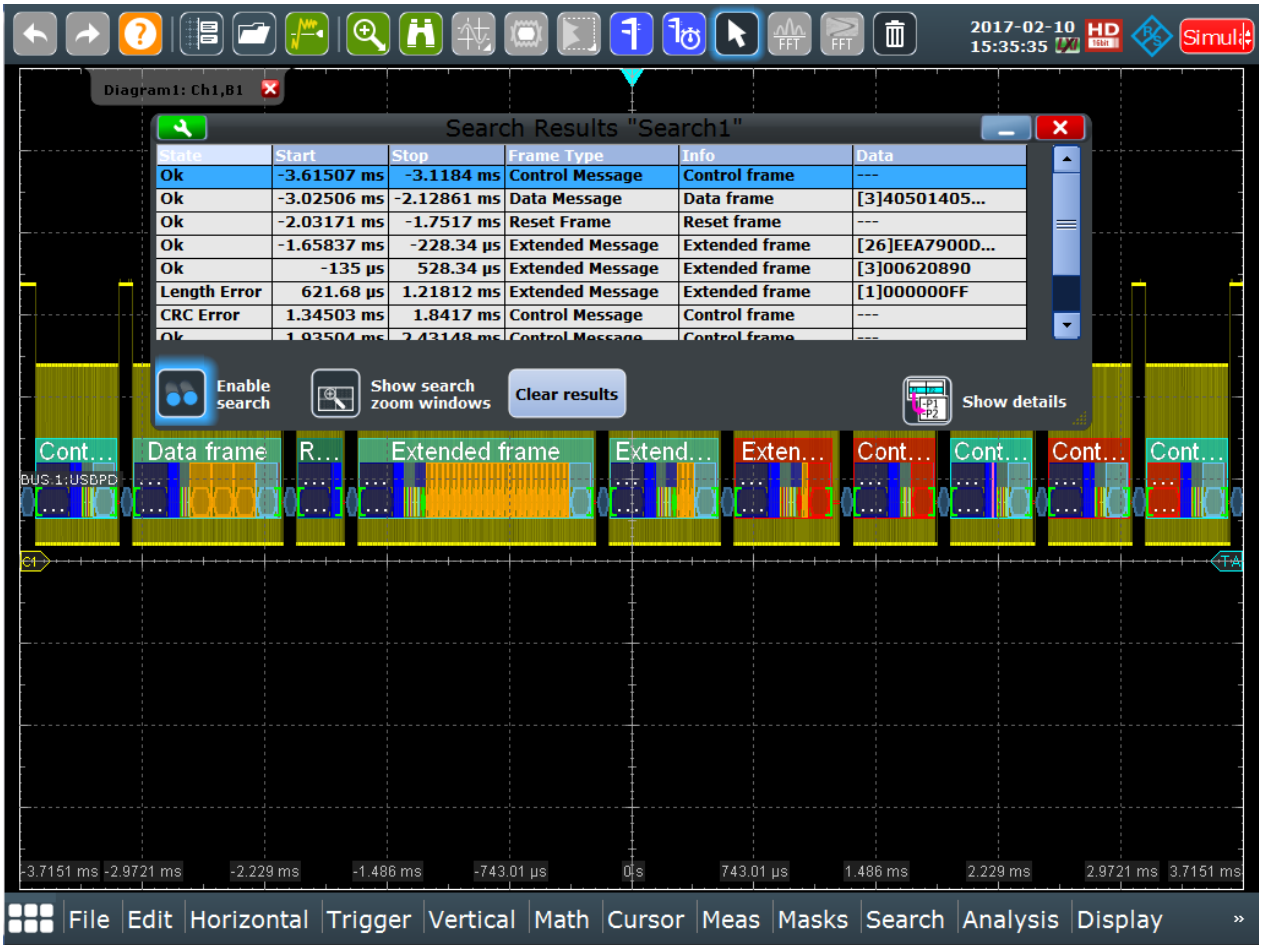


Figure 12-113: Search on USBPD frame

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- SEARch:RESult:USBPd:FCOunt? on page 1813
- SEARch:RESult:USBPd:FRAMe<m>:DATA? on page 1813
- SEARch:RESult:USBPd:FRAMe<m>:TYPE? on page 1817
- SEARch:RESult:USBPd:FRAMe<m>:INFO? on page 1816
- SEARch:RESult:USBPd:FRAMe<m>:STATe? on page 1817
- SEARch:RESult:USBPd:FRAMe<m>:STARt? on page 1816
- SEARch:RESult:USBPd:FRAMe<m>:STOP? on page 1817
- SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE? on page 1815
- SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus? on page 1814
- SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt? on page 1815
- SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP? on page 1815
- SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL? on page 1816
- SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL? on page 1814

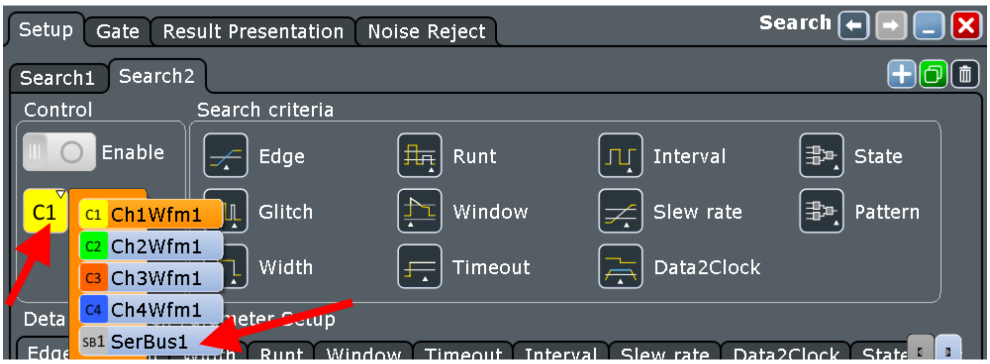
12.17.5.3 Searching USBPD Data

Prerequisite: A serial bus is configured for the USBPD signal to be decoded and analyzed.

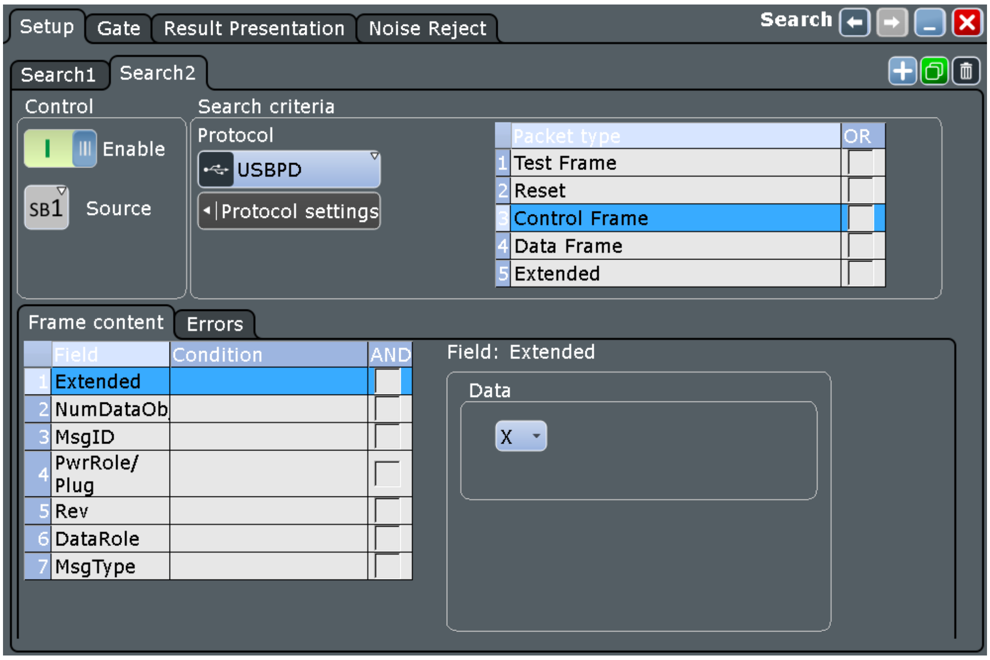
To be able to search for a USBPD data, the "Show decode detail" field in the "Configuration" tab of the protocol setup should be disabled. If "Show decode detail" is enabled, the "Enable" search button is disabled.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to USBPD (e.g. "SerBus1", unless already selected).



The search dialog for USBPD protocol analysis is opened.



4. Specify search criteria according to Chapter 12.17.5.1, "USBPD Search Setup", on page 827.
5. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs a USBPD decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.18 SpaceWire (Option R&S RTE-K65)

The SpaceWire is a communication network standard used for spacecrafts. It is based on the IEEE 1355 standard of comunications and coordinated by the European Space Agency (ESA).

● SpaceWire Basic.....832
● SpaceWire Configuration.....833
● SpaceWire Trigger.....836
● SpaceWire Decode Results.....839
● Search on Decoded SpaceWire Data.....842

12.18.1 SpaceWire Basic

The SpaceWire links are a Point-to-point (P2P) connection between a node and another node or a rooter. The link is full-duplex bidirectional serial data link.

The SpaceWire has two types of characters:

- Data characters containing a parity bit, a data control flag and eight bits of data.

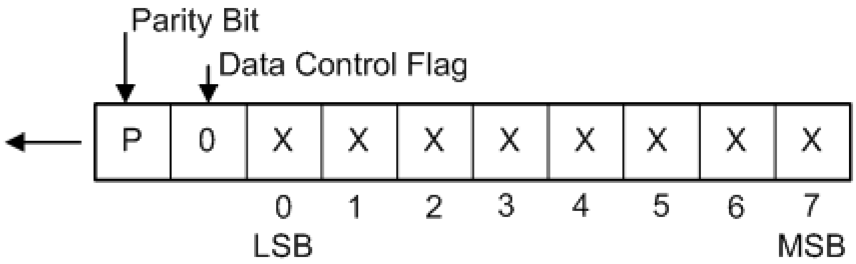


Figure 12-114: SpaceWire data characters

- Control characters containing a parity-bit, a data-control flag and the 2-bit control code. The data control flag is set to 1 and indicates that this is a control character.

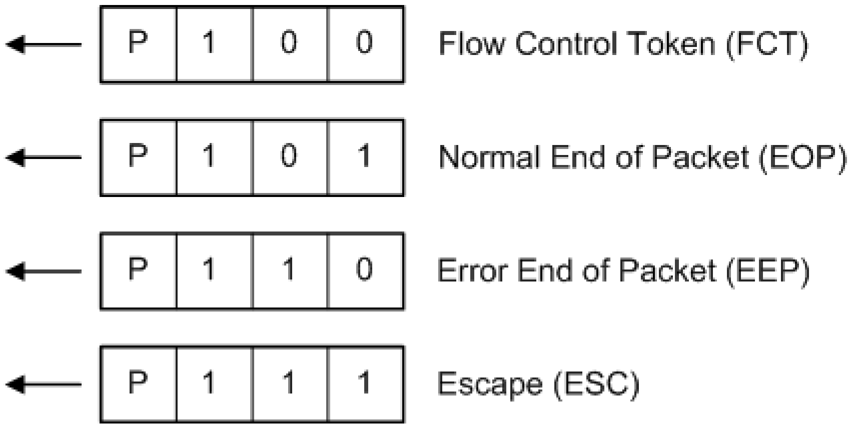


Figure 12-115: SpaceWire control characters

Also there are two control codes:

- NULL code consisting of an Escape (ESC) and a Flow Control Token (FCT)
- Time Code consisting of an ESC followed by a single data character

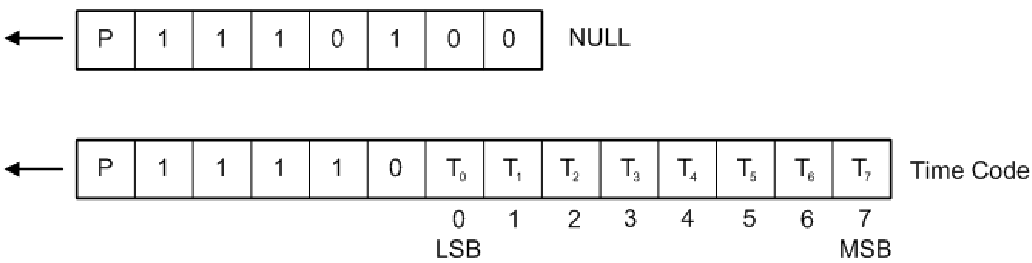


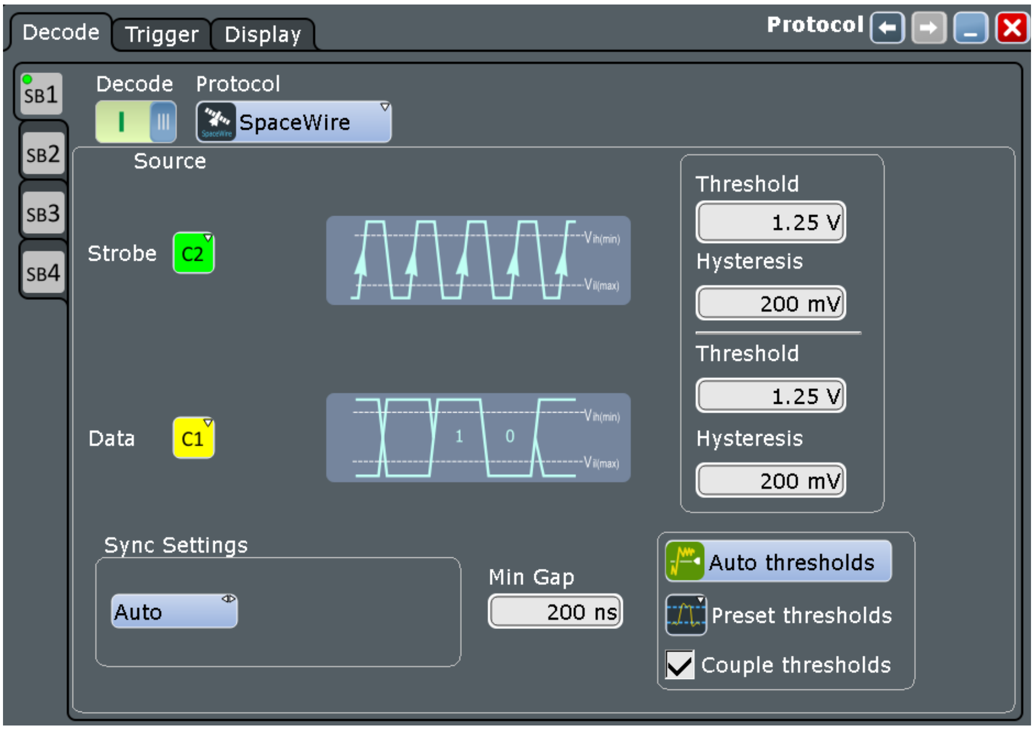
Figure 12-116: SpaceWire control codes

12.18.2 SpaceWire Configuration

12.18.2.1 SpaceWire Configuration Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol = SpaceWire"

Make sure that the tab of the correct serial bus is selected on the left side.



See also: Chapter 12.1.1, "Decode - General Settings", on page 474.

Strobe
Selects the source for the strobe signal.

Remote command:
BUS<m>:SWIRe:STRBe:SOURce on page 1820

Data
Selects the source for the data signal.

Remote command:
BUS<m>:SWIRe:DATA:SOURce on page 1819

Threshold setup
Sets the threshold value for the strobe/data signal.

There are several ways to set the threshold:
- "Threshold"
  Enter individual values for each line directly in the fields.
- "Preset thresholds"
  Selects the default threshold voltage from a list. The value is set to "Manual" if the threshold was set with "Auto thresholds", or was entered directly.
- "Auto thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
  This option is only available for analog sources.
- "Couple thresholds"
  Enables coupling, i.e. the same threshold and hysteresis value is used for the strobe and data signal.

Remote command:
`BUS<m>:SWIRe:COUPling` on page 1821
`BUS<m>:SWIRe:STRBe:THReshold` on page 1820
`BUS<m>:SWIRe:DATA:THReshold` on page 1819
`BUS<m>:SWIRe:PRESet` on page 1821
`BUS<m>:SETReflevels` on page 1383

Hysteresis
Sets a value for the hysteresis of the strobe/data signal.

Remote command:
`BUS<m>:SWIRe:STRBe:HYSTeresis` on page 1820
`BUS<m>:SWIRe:DATA:HYSTeresis` on page 1819

Sync Settings
Sets the mode for the synchronization of the signal. In the auto mode, the decoder automatically does the packet align. In the manual mode, you can set the align point manually with the "Bit Position" setting.

Remote command:
`BUS<m>:SWIRe:SYSLect` on page 1820

Bit Position ← Sync Settings
Sets the bit position, the align position for the manual synchronization mode. This can be useful when parity errors exist in the signal, and parity check is the main indicator for the decoder to do packet alignment.

Remote command:
`BUS<m>:SWIRe:BPOSition` on page 1818

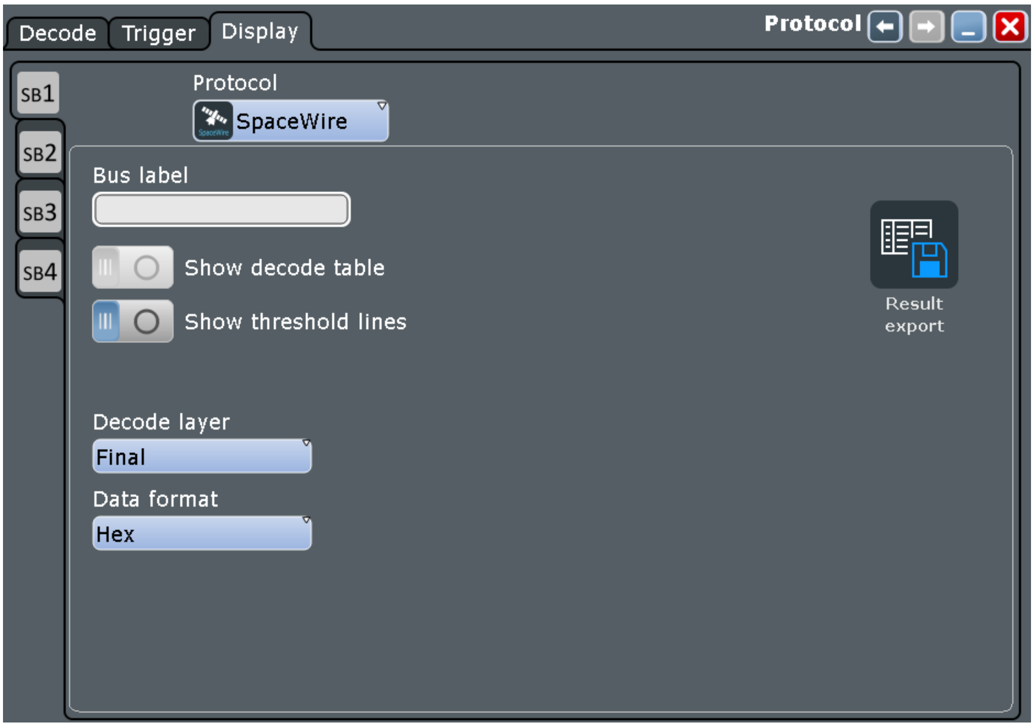
Min Gap
SpaceWire can have idle phases where strobe and data signals are not being sent. These "gaps" are identified to resume decoding after this idle time.

"Min Gap" sets the minimum duration of a gap. Any inactivity greater than this time is interpreted as a gap and lead to a resynchronization to the signal.

Remote command:
`BUS<m>:SWIRe:MGAP` on page 1819

12.18.2.2 Display Settings

Access: [PROTOCOL] > "Decode" tab > "Protocol" = "SpaceWire" > "Display" tab



Decode Layer

Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Final"	...
"Edges"	...
"Binary"	...

12.18.2.3 Configuring the SpaceWire Signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Decode" tab.
4. Tap the "Protocol" button and select the protocol: "SpaceWire".
5. Optionally, you can enter a "Bus label" on the "Display" tab.

6. Select the source and polarity for the strobe and data signals.
7. Enter the "Threshold" and the "Hysteresis" for the strobe and data signals.
8. Set the "Sync Settings" and the "Bit Position" if necessary.
9. Enable "Decode", if available.

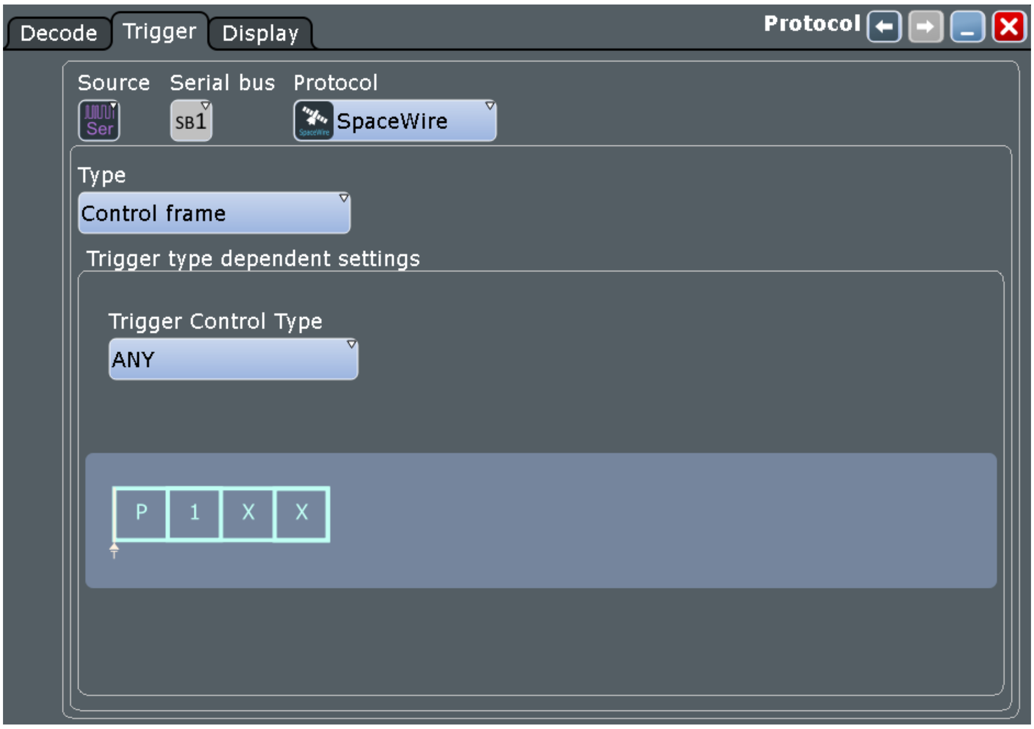
12.18.3 SpaceWire Trigger

12.18.3.1 SpaceWire Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = SpaceWire"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected USB protocol type and trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to data sheet.

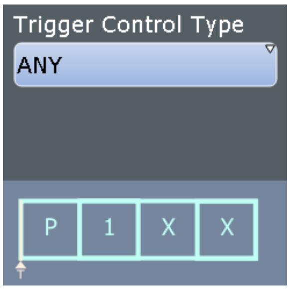


Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

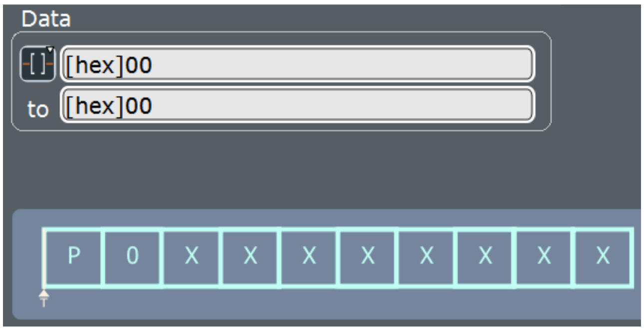
Type

Selects the trigger type for the SpaceWire analysis.

"Control frame" Sets the trigger to the selected control type frame.

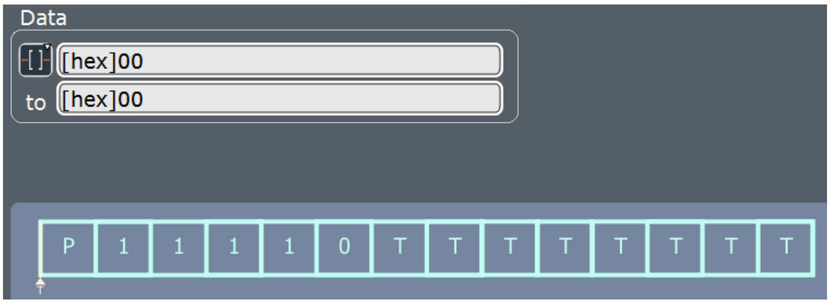


"Data pattern" Sets the trigger to a defined data pattern or pattern range.



"NULL frame" Sets the trigger to a null frame, a frame without usable data.

"Time Code" Sets the trigger to a time-code control code. You can define the data pattern of the time code to be triggered on.



"ERRORs" Triggers on an enabled error type.

Remote command:
`TRIGger<m>:SWIRe:TYPE` on page 1824

Trigger Control Type

Triggers on a specific control type character.

"ANY" Any control type character

"FCT" Flow Control Token character

"EOP" Normal End of Packet character

"EEP" Error End of Packet character

Remote command:
`TRIGger<m>:SWIRe:CTYPe` on page 1822

Data (Time Code)

Sets the specified data type for the time code to be triggered on. The data type setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.

"Data Min/ Data" Defines the bit pattern of the data pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:SWIRe:TIME:CONDition` on page 1823
`TRIGger<m>:SWIRe:TIME:MAX` on page 1824
`TRIGger<m>:SWIRe:TIME:MIN` on page 1824

Data(Data Pattern)

Sets the specified data type for the data pattern to be triggered on. The data type setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.

"Data Min/ Data" Defines the bit pattern of the data pattern.
In binary format, use the following characters: 1; 0; or X (do not care).
The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:SWIRe:DATA:CONDition` on page 1822
`TRIGger<m>:SWIRe:DATA:MAX` on page 1822
`TRIGger<m>:SWIRe:DATA:MIN` on page 1823

Parity Error
Checks the parity of every frame and triggers if the parity is even.

Remote command:
`TRIGger<m>:SWIRe:ERRor:PARity` on page 1823

ESC Error
Triggers on an escape error.

Remote command:
`TRIGger<m>:SWIRe:ERRor:ESC` on page 1823

12.18.3.2 Triggering on SpaceWire

Prerequisite: A bus is configured for the SpaceWire signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to SpaceWire.
5. Select the "Trigger Type" to be used for SpaceWire protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.
   For details, see Chapter 12.18.3.1, "SpaceWire Trigger Settings", on page 836.

12.18.4 SpaceWire Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Example

The example in Figure 12-117 shows a decoded signal with ambiguous bits (the earliest, least significant, parity bit among all surviving tracks). In the honeycomb the ambiguous bits are marked in pink. Additionally they are reflected in the result table.

Figure 12-117: Decoded SpaceWire signal with ambiguous bits

The example in Figure 12-118 shows a decoded signal with existing parity errors. The errors are marked with red on the honeycomb and reflected in the results table.

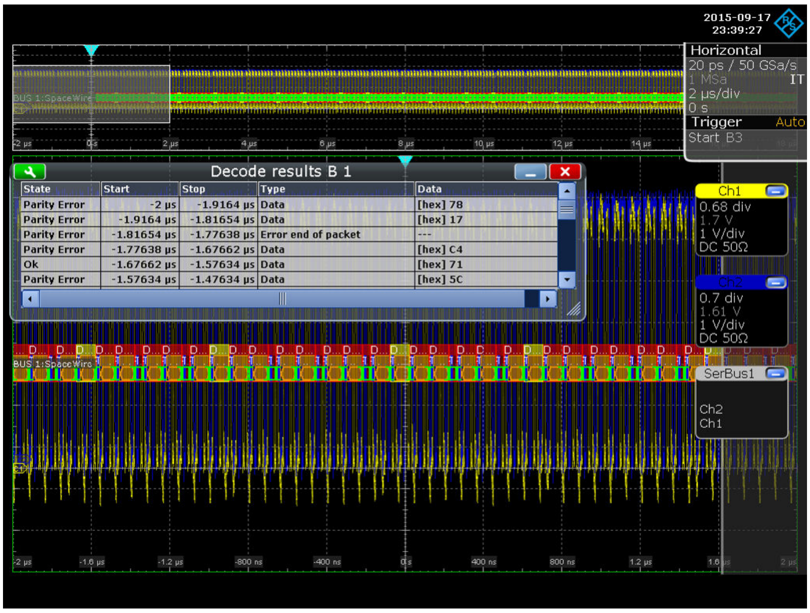


Figure 12-118: Decoded SpaceWire signal with a parity error

Table 12-28: Content of the decode result table

Column	Description
State	Overall state of the frame
Start	Time of frame start in relation to the trigger point
Stop	Time of frame stop in relation to the trigger point

Column	Description
Type	Frame type
Data	Data value

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

Table 12-29: Content of the "Details" table

Column	Description
State	State of the field
Type	Field type
Start	Time of field start in relation to the trigger point
Stop	Time of field stop in relation to the trigger point
Hex-Value	Hexadecimal value of the field
Formatted	Formatted content of the field

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.
2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands to retrieve decode results are described in Chapter 17.17.19.3, "Decode Results", on page 1824.

12.18.5 Search on Decoded SpaceWire Data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

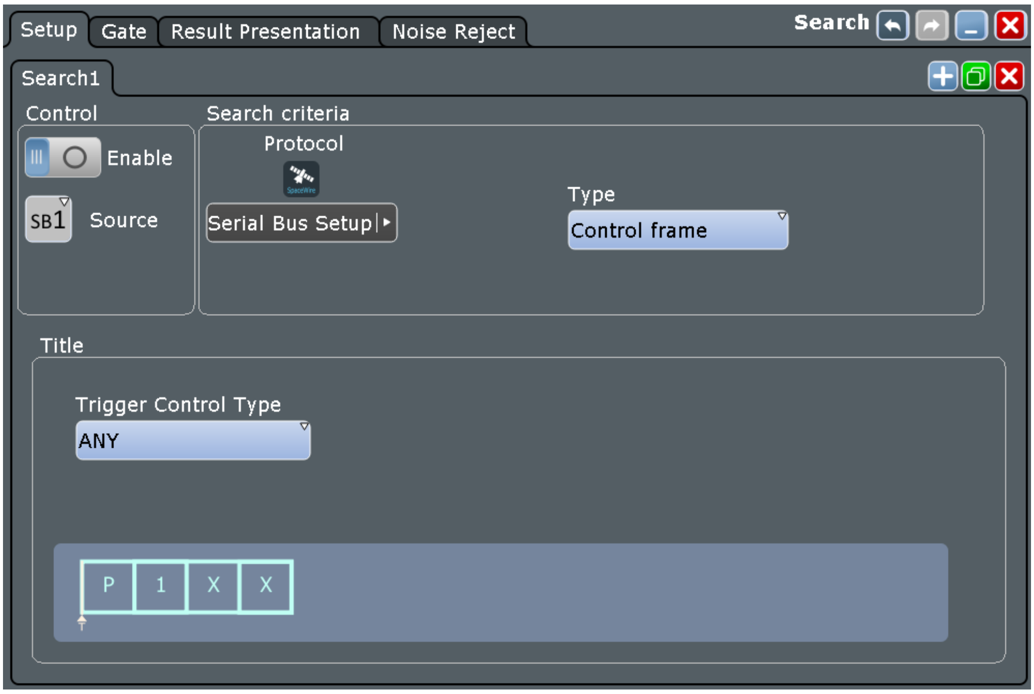
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 10, "Search Functions", on page 406.

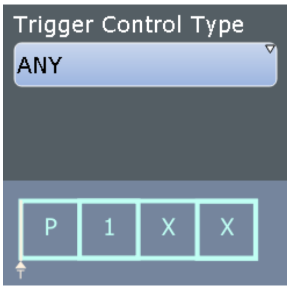
12.18.5.1 SpaceWire Search Setup

Access: [SEARCH] > "Setup" tab > "Source" = Serial bus configured for SpaceWire

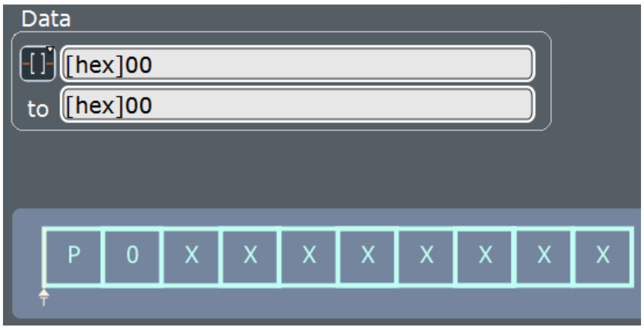


Type
Sets the search type for the SpaceWire analysis.

"Control frame" Searches for the selected control type frame.

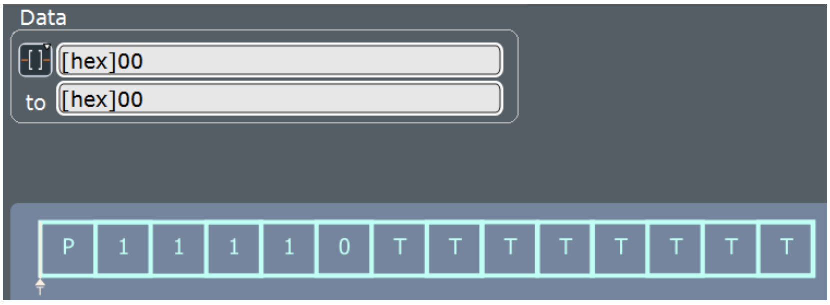


"Data pattern" Searches for a defined data pattern or pattern range.



"NULL frame" Searches for a null frame, a frame without usable data.

"Time Code" Searches for a time-code control code. You can define the data pattern of the time code to be searched for.



"ERRORs" Searches for an enabled error type.

Remote command:
SEARch:TRIGger:SWIRe:TYPE on page 1833

Trigger Control Type

Searches for a specific control type character.

"ANY" Any control type character

"FCT" Flow Control Token character

"EOP" Normal End of Packet character

"EEP" Error End of Packet character

Remote command:
SEARch:TRIGger:SWIRe:CTYPe on page 1830

Data (Time Code)
Sets the specified data type for the time code to be searched for. The data type setup consists of the condition and one or two data patterns.

"Condition"	Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.
"Data Min/ Data"	Defines the bit pattern of the data pattern. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"Data Max"	The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SWIRe:TIME:CONDition` on page 1832
`SEARch:TRIGger:SWIRe:TIME:MAX` on page 1832
`SEARch:TRIGger:SWIRe:TIME:MIN` on page 1832

Data(Data Pattern)
Sets the specified data type for the data pattern to be searched for. The data type setup consists of the condition and one or two data patterns.

"Condition"	Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.
"Data Min/ Data"	Defines the bit pattern of the data pattern. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"Data Max"	The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SWIRe:DATA:CONDition` on page 1830
`SEARch:TRIGger:SWIRe:DATA:MAX` on page 1831
`SEARch:TRIGger:SWIRe:DATA:MIN` on page 1831

Parity Error
Checks the parity of every frame and searches for even parity.

Remote command:
`SEARch:TRIGger:SWIRe:ERRor:PARity` on page 1831

ESC Error
Searches for an escape error.

Remote command:
`SEARch:TRIGger:SWIRe:ERRor:ESC` on page 1831

12.18.5.2 SpaceWire Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:

- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

Remote commands:

- SEARch:RESult:SWIRe:FCOunt? on page 1833
- SEARch:RESult:SWIRe:FRAMe<m>:DATA? on page 1833
- SEARch:RESult:SWIRe:FRAMe<m>:STARt? on page 1834
- SEARch:RESult:SWIRe:FRAMe<m>:STATe? on page 1834
- SEARch:RESult:SWIRe:FRAMe<m>:STOP? on page 1834
- SEARch:RESult:SWIRe:FRAMe<m>:TYPE? on page 1835

12.18.5.3 Searching SpaceWire

Prerequisite: A serial bus is configured for the SpaceWire signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press [SEARCH] or tap "Search" > "Setup" in the menu bar.
2. If the dialog box does not contain a search entry, tap the icon to create one, as described in "To create a user-defined search" on page 421.
3. Tap "Source" and select the serial bus that is set to SpaceWire (e.g. "SB1", unless already selected).
4. Specify search criteria according to Chapter 12.18.5.1, "SpaceWire Search Setup", on page 842.
5. To acquire a waveform, press [RUN N× SINGLE].

   The R&S RTE performs a SpaceWire decode according to the thresholds and protocol settings of the associated serial bus source.
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTE displays the "Search Results" box that lists the detected events. For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 427 and "Navigating search results" on page 408.

12.19 CXPI (Option R&S RTE-K76)

The Clock Extension Peripheral Interface (CXPI) protocol defines a communication standards for the vehicles electric system. It is developed by the Society of Automotive Engineers of Japan, Inc. (JSAE).

12.19.1 The CXPI Protocol

This chapter provides an overview of the protocol characteristics, frame types and frame fields.

CXPI characteristics

Main characteristics of CXPI are:

- Carrier Sense Multiple Access (CSMA) and Collision Resolution (CR)
- The master sends pulse width modulated clock (PWMC)
- Support of the sleep/wake function to reduce power consumption
- Support of two methods for frame transfer management, the event trigger method and the polling method.
  The event trigger method can be used when the focus is placed on the responsiveness of the slave node communication. The polling method puts the focus on the communication periodicity.
- Maximum baud rate of 20kbit/s
- Up to 16 nodes connected to a communication bus

Frame types

The CXPI protocol defines three types of frames:

- **Normal frame**: it varies in length according to the frame transfer method. The PTYPE field is only used for the polling method.
  The normal frame is used to transfer compact data.

Figure 12-119: CXPI normal frame

- **Sleep frame**: it has a fixed length and fixed values for the PID and data fields. It is sent by the master node to command the slave nods to change into sleep state.

Figure 12-120: CXPI sleep frame

- **Long frame**: it varies in length according to the frame transfer method. The PTYPE field is only used for the polling method.
  The long frame is used to transfer large amount of data.

Figure 12-121: CXPI long frame

During measurements further type of frames may also occur: PTYPE only, PID only and PTYPE + PID only.

Frame fields

The different frames may contain the following fields:

- **PID**: consists of a frame identifier (7 bits) and one parity bit.
- **Protected TYPE (PTYPE)**: a special PID field with a frame identifier of 00 (hex) and a parity of 1. It is used only in the polling method.
- **Frame information**: it consists of the following fields:
  – **Data Length Code (DLC)**: 4 bits that indicate the length of the data byte of a normal frame.
  – **Network Management (NM)**: 2 bits indicating if the frame is in wakeup or sleep mode.
  – **Counter (CT)**: 2 bits indicating the continuity of the frame.
  – **extension DLC**: 1 byte that is added to the frame information for long frames. It indicates the length of the data byte field of a long frame.
- **Data byte**: the length of the data byte field depends on the frame type. It contains the actual information of the frame.
- **Cyclic Redundancy Check (CRC)**: 1 byte for normal and sleep frames and 2 bytes for long frames. CRC errors can be detected to check if the received data is correct.

12.19.2 CXPI Configuration

12.19.2.1 CXPI Configuration Settings

Access: [PROTOCOL] key > "Decode" tab > "Protocol" = "CXPI"

Make sure that the tab of the correct serial bus is selected on the left side.

Source
Sets the source channel for the signal.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
BUS<m>:CXPI:SDATa on page 1838

Polarity
Selects if the polarity of the transmitted waveform is normal (high = 1) or inverted (low = 1).

Remote command:
BUS<m>:CXPI:POLarity on page 1837

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high. If the signal value on the line is below the threshold, the signal state is low. The interpretation of high and low is defined by the Polarity.

There are two ways to set the threshold:

- "Threshold"
  Enter the value directly in the field.
- "Auto threshold"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
BUS<m>:CXPI:THReshold on page 1837
BUS<m>:FAUToset on page 1383

Hysteresis
Sets a value for the hysteresis.

The hysteresis is applied as ("Threshold" +"Hysteresis") and ("Threshold" - "Hysteresis").

Remote command:
BUS<m>:CXPI:HYSTeresis on page 1837

Display order
Selects the order in which the signal is displayed in the honeycomb.

"As transmitted"
The signal is displayed in the order it occurs.

"Logical" The signal is displayed according to the definition of the standard (MSB bit order).

Remote command:
BUS<m>:CXPI:DORD on page 1837

Bit rate/ Average bit rate
If enabled, sets the number of transmitted bits per second. If disabled, the bit rate is measured and the average bit rate in the acquisition window is displayed.

Remote command:
`BUS<m>:CXPI:BITRate:ENABle` on page 1836
`BUS<m>:CXPI:BITRate:VALue` on page 1836
`BUS<m>:CXPI:RESult:BITRate?` on page 1836

Expected IBS length
Sets a range for the expected inter-byte-space (IBS) length.

The standard has a permissible IBS length. Violation of these limits lead to decoding ambiguities. To test different limits you can set the expected IBS length value.

Remote command:
`BUS<m>:CXPI:IBS:MAX` on page 1838
`BUS<m>:CXPI:IBS:MIN` on page 1838

Expected IFS length
Sets a range for the inter-frame-space.

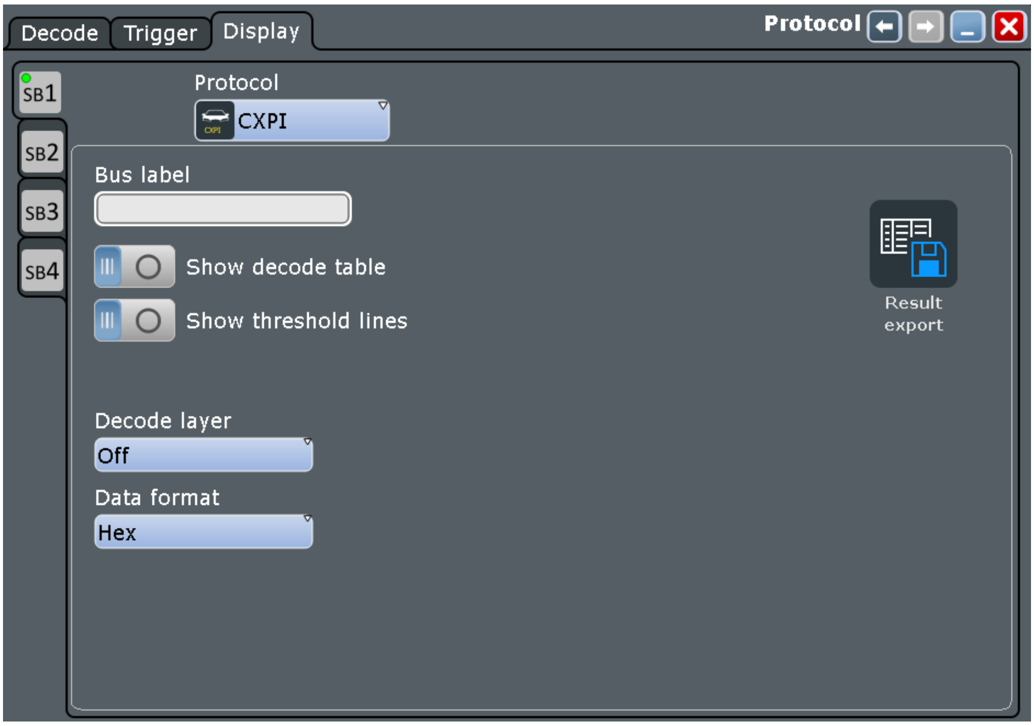
The standard has a permissible IFS length. Violation of these limits lead to decoding ambiguities. To test different limits you can set the expected IFS length value.

Remote command:
`BUS<m>:CXPI:IFS:MAX` on page 1839
`BUS<m>:CXPI:IFS:MIN` on page 1839

12.19.2.2 Display Settings

Access: [PROTOCOL] > "Configuration" tab > "Protocol = CXPI" > "Display" tab

To enhance the decode possibilities of the CXPI protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Common display settings are explained in Chapter 12.1.3, "Display", on page 475.

Decode layer

Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Edges"	...
"Final"	...
"Binary"	...
"Bits"	...

12.19.2.3 Configuring the CXPI Signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

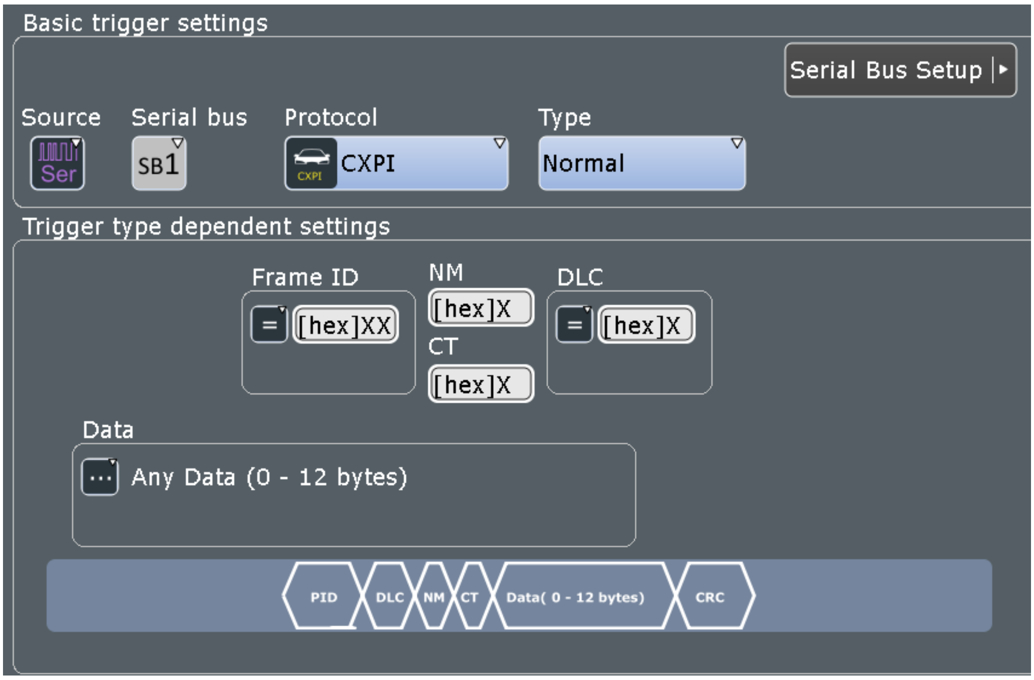
1. Press the [PROTOCOL] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Configuration" tab.
4. Tap the "Protocol" button and select the protocol: "CXPI".
5. Optionally, you can enter a "Bus label" on the "Display" tab.
6. Tap the "Variant" button, and select the protocol.

7. Select the source and the polarity.
8. Enter the "Threshold" and the "Hysteresis".
9. If necessary, enable and set the "Bit rate".
10. Enable "Decode", if available.

12.19.3 CXPI Trigger

12.19.3.1 CXPI Trigger Settings

Access: [PROTOCOL] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = CXPI"



Make sure that:
- The data source(s) of the serial bus are channel signals: [PROTOCOL] > "Decode" tab.
- The trigger sequence is set to "A only": [TRIGGER] > "Sequence" tab.
- The trigger source is "Serial bus": [TRIGGER] > "Events" tab.
- The correct serial bus is selected: [TRIGGER] > "Events" tab.
- The correct protocol is selected: [TRIGGER] > "Events" tab.

Trigger type
Selects the trigger type for CXPI analysis.

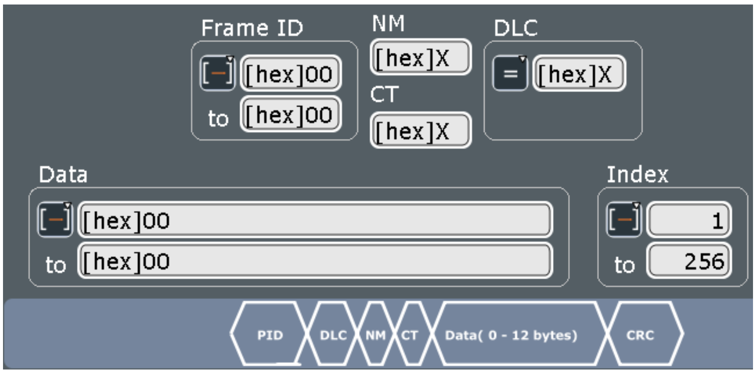
Remote command:
`TRIGger<m>:CXPI:TYPE` on page 1846

Frame Start ← Trigger type
Triggers on a frame start.

Normal ← Trigger type
Triggers on a normal frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 856
- Set an NM value expected in the frame.
  See "NM" on page 858.
- Set a CT value expected in the frame.
  See: "CT" on page 858.
- Set a DLC pattern or DLC pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC" on page 856.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 857.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 858.

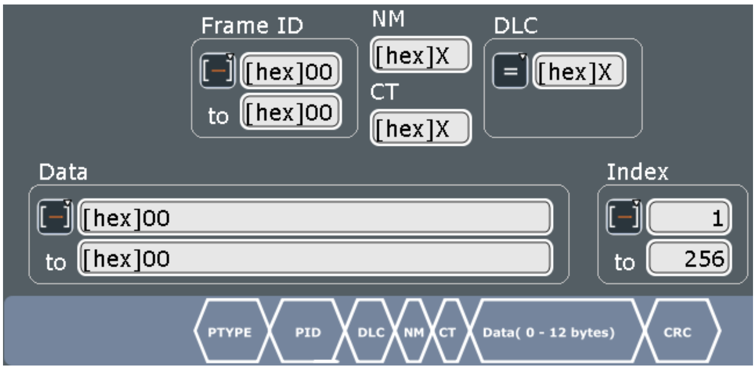


Normal Poll ← Trigger type
Triggers on a normal poll frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 856
- Set an NM value expected in the frame.
  See "NM" on page 858.
- Set a CT value expected in the frame.
  See: "CT" on page 858.

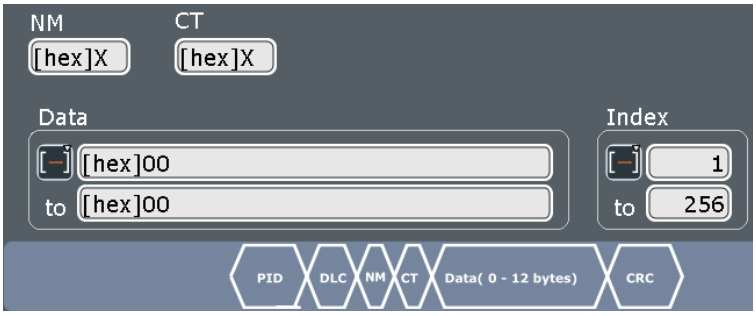
- Set a DLC pattern or DLC pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC" on page 856.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 857.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 858.



Sleep ← Trigger type
Triggers on a sleep frame.

You can refine the trigger condition:

- Set an NM value expected in the frame.
  See "NM" on page 858.
- Set a CT value expected in the frame.
  See: "CT" on page 858.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 857.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 858.

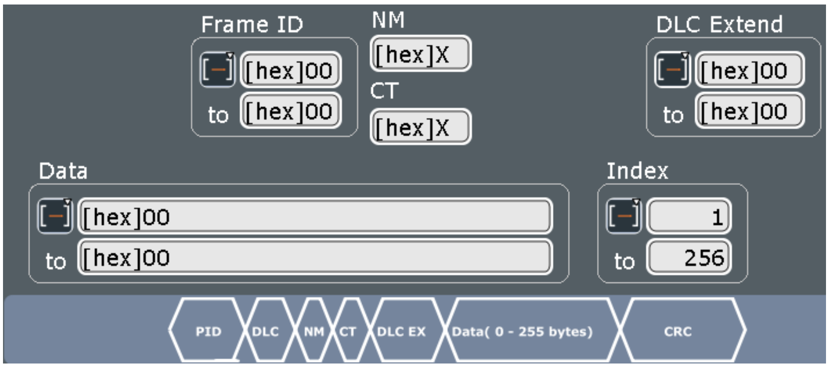


Long ← Trigger type
Triggers on a long frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 856
- Set an NM value expected in the frame.
  See "NM" on page 858.

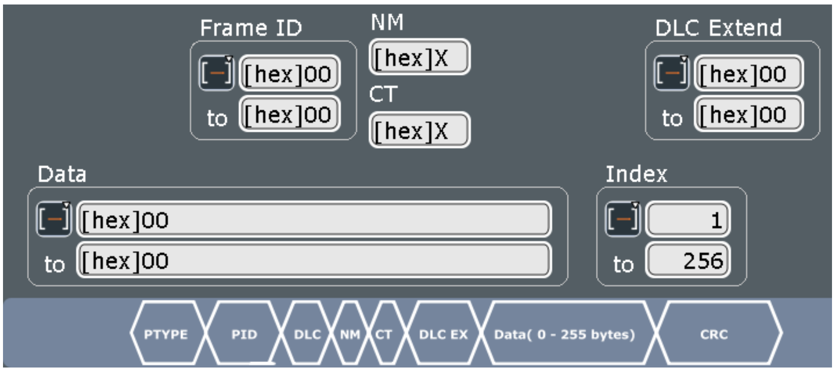
- Set a CT value expected in the frame.
  See: "CT" on page 858.
- Set a DLC extend pattern or DLC extend pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC Extend" on page 857.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 857.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 858.



Long Poll ← Trigger type
Triggers on a long poll frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 856
- Set an NM value expected in the frame.
  See "NM" on page 858.
- Set a CT value expected in the frame.
  See: "CT" on page 858.
- Set a DLC extend pattern or DLC extend pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC Extend" on page 857.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 857.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 858.



PID ← Trigger type
Triggers on a Protected ID (PID) field. A PID field consists of a frame identifier and a parity bit.

Optionally, you can define the frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.

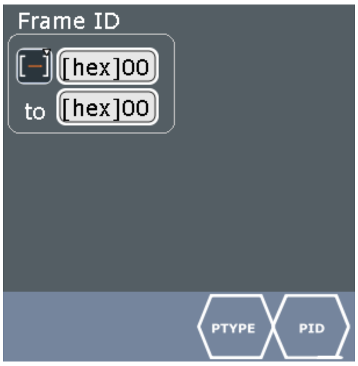
See:"Frame ID" on page 856.

PTYPE ← Trigger type
Triggers on a Protected TYPE (PTYPE) field, a special PID field with a frame ID of 00 (hex) and a parity of 1. It is used only in the polling method.

PTYPE + PID ← Trigger type
Triggers on a PTYPE field followed by a PID field.

Optionally, you can define a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.

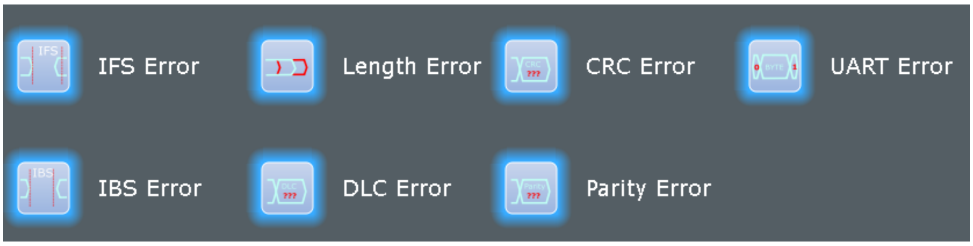
See:"Frame ID" on page 856.



Error ← Trigger type
Triggers if one or more of the following errors occur:

- "IFS Error"
  The inter-frame space (IFS) is a bit field after a frame that has a logical value 1. It is used to separate two consecutive frames. IFS errors occur when the length of the IFS field differs from the one defined in the protocol.

- "IBS Error"
  The inter-byte-space (IBS) is the interval between each byte within a frame. IBS errors occur when the length of the IBS field differs from the one defined in the protocol.
- "Length Error"
  Length error occurs when there are not enough bits/words to build a frame.
- "DLC Error"
  Data length code error occurs when the value of the DLC field is different from the data byte field value. In case of a DLC error, it is still possible to build a frame, but there are too many/too few data words.
- "CRC Error"
  The transmitting node calculates the cyclic redundancy check (CRC) value of a frame and stores it into the CRC field of the frame. CRC error occurs when this CRC field value differs from the value calculated by the receiving node.
- "Parity Error"
  Triggers on a parity error indicating a transmission error.
- "UART Error"
  For the CXPIstandard, each 8-bit byte is transported as a 10-bit UART word, framed by a start bit (1) and stop bit (0). An UART error occurs if the identified symbol does not follow the described pattern, if it is too short or too long.



Frame ID
The frame identifier is a 7-bit field. Its setup consists of the condition and one or two frame patterns.

"Condition"	Defines the operator to set a specific frame identifier ("Equal" or "Not equal") or a frame identifier range.
"Frame ID min / Frame ID"	Defines the bit pattern of the frame identifier. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"Frame ID max"	The second frame identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:FID:CONDition` on page 1845
`TRIGger<m>:CXPI:FID:MAX` on page 1846
`TRIGger<m>:CXPI:FID:MIN` on page 1846

DLC
Sets the data length code, which indicates the length of the data byte.

The DLC setup consists of the condition and one or two DLC patterns.

"Condition"	Defines the operator to set a specific DLC ("Equal" or "Not equal") or a DLC range.
"DLC min / DLC "	Defines the bit pattern of the DLC. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"DLC max"	The second DLC pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DLC:CONDition` on page 1843
`TRIGger<m>:CXPI:DLC:MAX` on page 1843
`TRIGger<m>:CXPI:DLC:MIN` on page 1843

DLC Extend

Sets the extension data length code, which indicates the length of the data byte of a long frame.

The DLC extend setup consists of the condition and one or two DLC patterns.

"Condition"	Defines the operator to set a specific DLC extend ("Equal" or "Not equal") or a DLC extend range.
"DLC Extend min / DLC Extend"	Defines the bit pattern of the DLC extend. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"DLC Extend max"	The second DLC extend pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DEXTension:CONDition` on page 1842
`TRIGger<m>:CXPI:DEXTension:MAX` on page 1842
`TRIGger<m>:CXPI:DEXTension:MIN` on page 1843

Data

The data setup consists of the condition and one or two identifier pattern.

"Condition"	Defines the operator to set a specific data ("Equal" or "Not equal") or a data range.
"Data min / Data"	Defines the bit pattern of the data. In binary format, use the following characters: 1; 0; or X (do not care). The bit pattern editor helps you to enter the pattern in any format, see Chapter 12.1.5, "Bit Pattern Editor", on page 481.
"Data max"	The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DATA:DCONdition` on page 1840
`TRIGger<m>:CXPI:DATA:DMAX` on page 1841
`TRIGger<m>:CXPI:DATA:DMIN` on page 1841

Index

The data index setup consists of the condition and one or two index patterns.

"Condition"	Defines the operator to set a specific data index ("Equal" or "Not equal") or data index range.
"Index min / Index"	Defines the bit pattern of the data index.
"Index max"	The second data index is required to specify a range with conditions "In range" and "Out of range".

Remote command:
TRIGger<m>:CXPI:DATA:ICONdition on page 1841
TRIGger<m>:CXPI:DATA:IMAX on page 1841
TRIGger<m>:CXPI:DATA:IMIN on page 1842

NM

Sets the value of the network management (NM) field, 2 bits indicating if the frame is in wakeup or sleep mode.

Remote command:
SEARch:TRIGger:CXPI:NM on page 1858

CT

Sets the value of the counter (CT), 2 bits indicating the continuity of the frame.

Remote command:
TRIGger<m>:CXPI:CT on page 1840

12.19.3.2 Triggering on CXPI

Prerequisite: A bus is configured for the CXPI signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [PROTOCOL] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to I²C.
5. Tap "Trigger Type CXPI" and select the trigger type to be used for CXPI protocol analysis.
6. Depending on the selected trigger type, more setup conditions can be specified.

For information on how to proceed with the configuration settings, see Chapter 12.19.3.1, "CXPI Trigger Settings", on page 851.

12.19.4 Search on Decoded CXPI Data

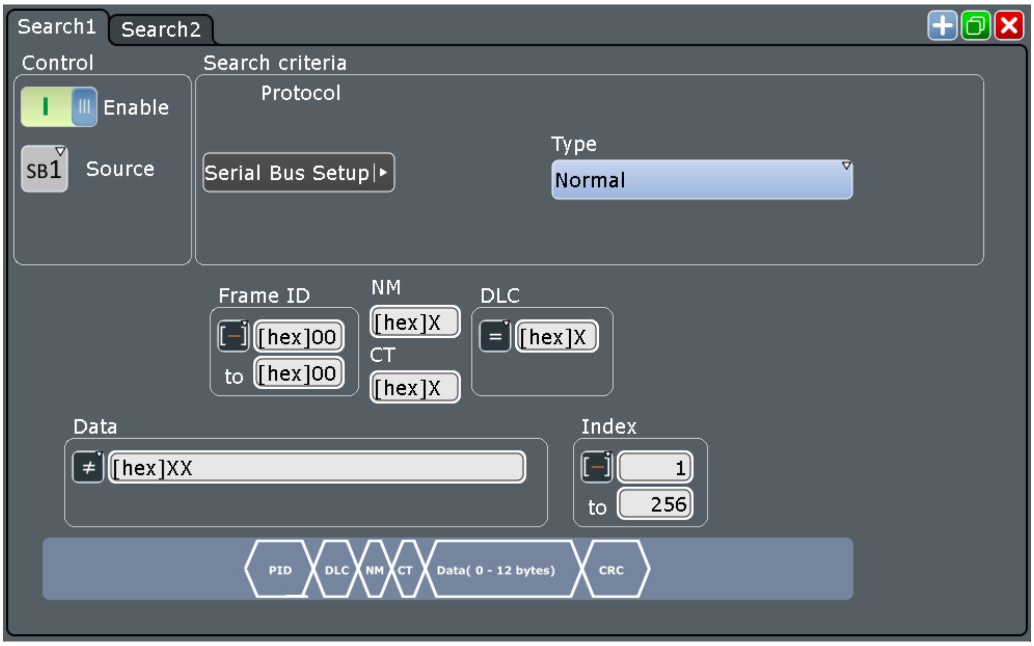
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 10, "Search Functions", on page 406.

12.19.4.1 Search Settings

Access: [SEARCH] > "Setup" tab



Search type
Selects the condition type for the CXPI search.

"Frame Start" Searches for a frame start.

"Normal" Searches for a normal frame.
You can refine the search condition, see:

- "Frame ID" on page 860
- "NM" on page 862
- "CT" on page 862
- "DLC" on page 861
- "Data" on page 861
- "Index" on page 861

"Normal Poll" Searches for a normal poll frame.
You can refine the search condition, see:

- "Frame ID" on page 860
- "NM" on page 862
- "CT" on page 862
- "DLC" on page 861
- "Data" on page 861
- "Index" on page 861

"Sleep" Searches for a sleep frame.
You can refine the search condition, see:

- "NM" on page 862
- "CT" on page 862
- "Data" on page 861
- "Index" on page 861

"Long" Searches for a long frame.
You can refine the search condition, see:

- "Frame ID" on page 860
- "NM" on page 862
- "CT" on page 862
- "DLC Extend" on page 861
- "Data" on page 861
- "Index" on page 861

"Long Poll" Searches for a long poll frame.
You can refine the search condition, see:

- "Frame ID" on page 860
- "NM" on page 862
- "CT" on page 862
- "DLC Extend" on page 861
- "Data" on page 861
- "Index" on page 861

"PID" Searches for a PID field. You can refine the frame identifier, see "Frame ID" on page 860.

"PTYPE" Searches for a PTYPE field

"PTYPE + PID" Searches for a PTYPE field followed by a PID field. You can refine the frame identifier, see "Frame ID" on page 860.

Remote command:
`SEARch:TRIGger:CXPI:TYPE` on page 1859

Frame ID

Searches for a frame identifier pattern or a frame identifier range. The setup consists of the condition and one or two frame patterns.

The frame identifier setup settings are the same as in the CXPI trigger setup, see "Frame ID" on page 856.

Remote command:
SEARch:TRIGger:CXPI:FID:CONDition on page 1857
SEARch:TRIGger:CXPI:FID:MAX on page 1858
SEARch:TRIGger:CXPI:FID:MIN on page 1858

DLC

Searches for a DLC pattern or a DLC range. The setup consists of the condition and one or two DLC patterns.

The DLC setup settings are the same as in the CXPI trigger setup, see "DLC" on page 856.

Remote command:
SEARch:TRIGger:CXPI:DLC:CONDition on page 1855
SEARch:TRIGger:CXPI:DLC:MAX on page 1855
SEARch:TRIGger:CXPI:DLC:MIN on page 1855

DLC Extend

Searches for a DLC extend pattern or a DLC extend range. The setup consists of the condition and one or two DLC extend patterns.

The DLC extend setup settings are the same as in the CXPI trigger setup, see "DLC Extend" on page 857.

Remote command:
SEARch:TRIGger:CXPI:DEXTension:CONDition on page 1854
SEARch:TRIGger:CXPI:DEXTension:MAX on page 1854
SEARch:TRIGger:CXPI:DEXTension:MIN on page 1855

Data

Searches for a data pattern or a data range. The setup consists of the condition and one or two data patterns.

The data setup settings are the same as in the CXPI trigger setup, see "Data" on page 857.

Remote command:
SEARch:TRIGger:CXPI:DATA:DCONdition on page 1852
SEARch:TRIGger:CXPI:DATA:DMAX on page 1852
SEARch:TRIGger:CXPI:DATA:DMIN on page 1853

Index

Searches for a data index pattern or a data index range. The setup consists of the condition and one or two index patterns.

The index setup settings are the same as in the CXPI trigger setup, see "Index" on page 858.

Remote command:
SEARch:TRIGger:CXPI:DATA:ICONdition on page 1853
SEARch:TRIGger:CXPI:DATA:IMAX on page 1853
SEARch:TRIGger:CXPI:DATA:IMIN on page 1854

NM
Searches for the value of the network management (NM), 2 bits indicating if the frame is in wakeup or sleep mode.

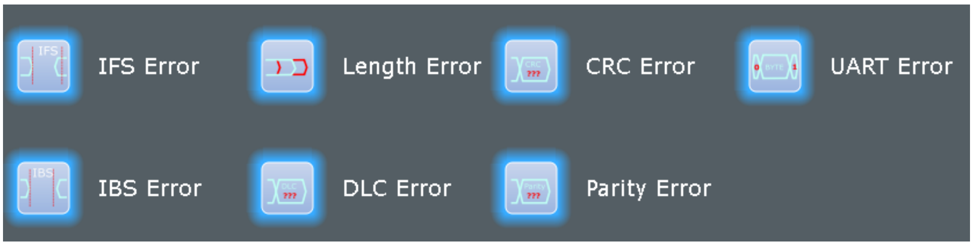
Remote command:
TRIGger<m>:CXPI:NM on page 1846

CT
Searches for the value of the counter (CT), 2 bits indicating the continuity of the frame.

Remote command:
SEARch:TRIGger:CXPI:CT on page 1852

Error
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the CXPI trigger setup, see "Error" on page 855.



12.19.4.2 Search Results

The search results are listed in the search result table and marked in the waveform by blue lines.

The "Show search zoom windows" function allows you to analyze the search results in more detail. Search zoom and result table are synchronized; if you select a row in the result table, this result is shown in the search zoom.

For an introduction to search results, see:
- Chapter 10.1.2, "Search Results", on page 407
- Chapter 10.4, "Result Presentation", on page 424

The columns in the search result table are the same as in the decoding table, see Chapter 12.19.5, "CXPI Decode Results", on page 863.

Remote commands:
- SEARch:RESult:CXPI:FCOunt? on page 1860
- SEARch:RESult:CXPI:FRAMe<m>:STATe? on page 1861
- SEARch:RESult:CXPI:FRAMe<m>:STARt? on page 1861
- SEARch:RESult:CXPI:FRAMe<m>:STOP? on page 1861
- SEARch:RESult:CXPI:FRAMe<m>:TYPE? on page 1862
- SEARch:RESult:CXPI:FRAMe<m>:DATA? on page 1860
- SEARch:RESult:CXPI:FRAMe<m>:DLCV? on page 1861

- SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE? on page 1862
- SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue? on page 1863
- SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus? on page 1862

12.19.4.3 Searching CXPI Data

Prerequisites: A CXPI bus is configured, see Chapter 12.19.2, "CXPI Configuration", on page 847, and "Decode" is enabled.

1. Press the [SEARCH] key on the front panel.
2. Tap the "Source" button and select the serial bus that is set to CXPI.

   "Protocol" shows the CXPI icon.
3. Tap "Type" and select the search type.
   All trigger types are also available for search.
4. To refine the search settings, configure additional settings, which are available for many search types.
   For details, see Chapter 12.19.4.1, "Search Settings", on page 859.
5. Under "Control", "Enable" the search.

   The "Search Results" box opens.
6. Close the "Search" dialog box.
7. Press [RUN CONT] to start acquisition.
8. Stop acquisition, or tap "Show search zoom window".

   Now you can navigate the search results and analyze the signal.

12.19.5 CXPI Decode Results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Decode" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals". For a description of the display settings, see also Chapter 12.1.3, "Display", on page 475

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

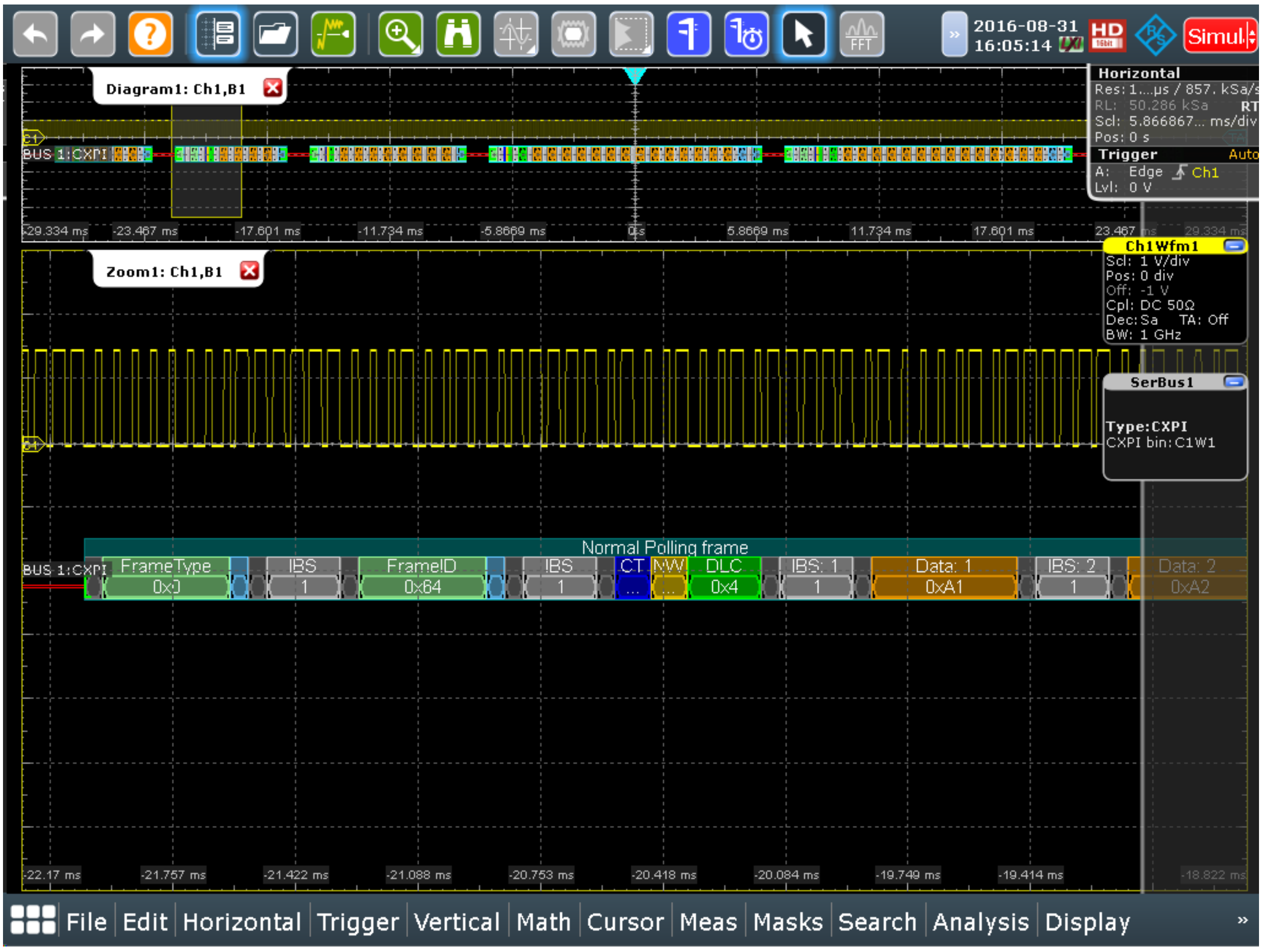


Figure 12-122: Decoded CXPI signal, normal polling frame

green brackets [...]	= start and end of frame
green frame header	= frame state is ok
red frame header	= error in frame
green	= Frame type/ frame ID
green	= DLC
grey	= IBS
dark blue	= CT
yellow	= NM
orange	= Data
red	= Error occurred

Enable "Show details" in the decode table to display a more detailed analysis of the selected frame. All data bytes are listed (in hexadecimal format).

Figure 12-123: Decoded CXPI signal result tables, normal polling frame with CRC error

Table 12-30: Content of the "Decode results" table

Column	Description
State	Overall state of the frame
Start	Time of frame start
Stop	Time of frame stop
Type	Frame type: normal, normal polling, sleep, long, long polling, PID, PTYPE, PTYPE +PID, unknown, inter-frame-space
Data	Displays the first four data words in the frame. For the value of the following data words, refer to Content of the "Decode results details" table.
DLC field	Data length code, number of data bytes

Table 12-31: Content of the "Decode results details" table

Column	Description
Type	Type (name) of field
Numeric value	The numeric value of the field
Status	Status of the field: OK or error

Data format

You can select various data formats for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:

- `BUS<m>:FORMat` on page 1384

Zooming on decoded results

The zoom functions allow you to analyze the decode results in more detail. If "Zoom coupling" is enabled, the decode zoom and result table are synchronized; if you select a row in the result table, this result is shown in the decode zoom.

Remote command:

- `BUS<m>:ZCOupling` on page 1385

For an introduction to zoom settings and setup, see:

- Chapter 6.1.2, "Zoom Settings", on page 242
- Chapter 6.1.3, "Zooming for Details", on page 246

Export of decode results

1. In the protocol decode table, press "Export".

   The "Numeric Results" dialog opens. For details, see Chapter 11.2.4, "Numeric Results", on page 452.

2. Select the decode results you want to export, the file format, and the delimiter.
3. Tap "Save" or "Save as".

Remote commands

Remote commands are described in Chapter 17.17.20.3, "Decode Results", on page 1848.

13 Mixed Signal Option (MSO, R&S RTE-B1)

The Mixed Signal Option R&S RTE-B1 adds logic analyzer functions to the classical oscilloscope functions. Using the MSO option, you can analyze and debug embedded systems with mixed-signal designs that use analog signals and correlated digital signals simultaneously.

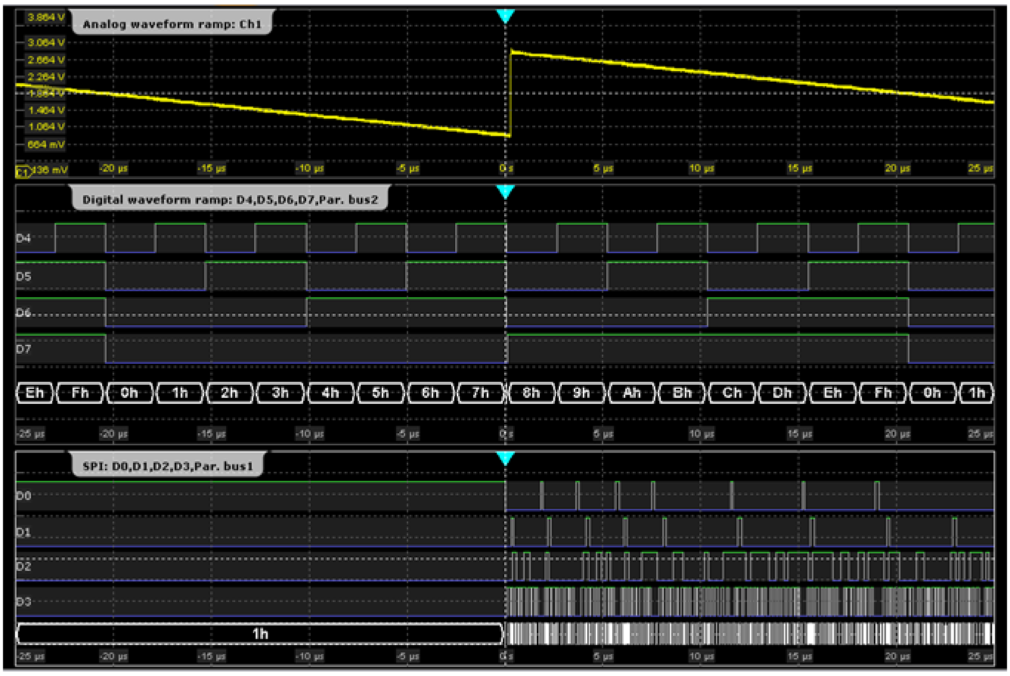
The Mixed Signal Option provides 16 digital channels grouped in two logic probes (pods) with 8 channels each. The instrument ensures that analog and digital waveforms are time-aligned and synchronized so that critical timing interactions between analog and digital signals can be displayed and tested. The automatic alignment compensates the skew between the probe connectors of the analog channels and the probe boxes of the digital channels.

13.1 Digital Channels and Parallel Buses

Each digital channel can be displayed on the screen and used as trigger source. Digital channels may be grouped and displayed as a parallel bus. Up to four parallel buses can be configured; and two bus types are supported: clocked bus and unclocked bus. The clocked bus is available only on parallel bus 1 and 2. Each digital channel can be assigned to one *active* parallel bus only, the instrument disables conflicting buses automatically.

You can display each bus and use it as trigger source, as well. For each active parallel bus, the corresponding signal icon appears on the signal bar and indicates the assigned digital channels. Individual digital channels do not have a signal icon.

If one or more parallel buses are active, the roll mode is not available.

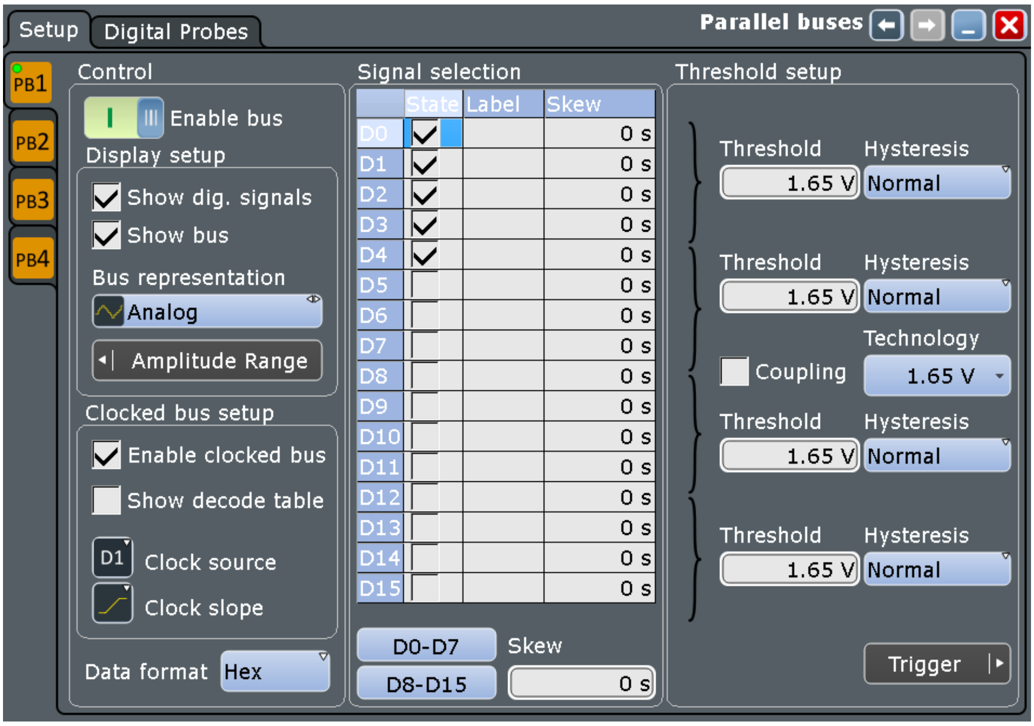


13.1.1 Parallel Buses - Configuration

Access: "Analysis" menu > "Parallel buses"

Digital channels can be displayed individually, and they can be grouped and displayed as a parallel bus. You can configure and enable up to 4 parallel buses. Each digital channel can be assigned to one *active* parallel bus only, the instrument disables conflicting buses automatically.

For clocked buses, you can display the decoded data in a result box.



If you have configured several parallel buses and you want to modify the settings, make sure that the tab of the correct bus is selected on the left side, and disable the bus before you change the settings.

Enable bus 869
Show dig. signals 870
Show bus 870
Bus representation 870
Amplitude Range 870
Clocked bus setup 871
Data format 871
Signal selection 872
└ D0-D7, D8-D15 872
└ Deskew offset 872
Threshold setup 872

Enable bus

Enables the selected parallel bus. The corresponding signal icon appears on the signal bar.

If another *active* bus already uses the same digital channel, the instrument disables the other bus and shows a message.

Remote command:

`BUS<m>:PARallel:STATe` on page 1867

Show dig. signals
If enabled, the selected digital channels are shown in the diagram. Each channel is displayed as a logic signal.

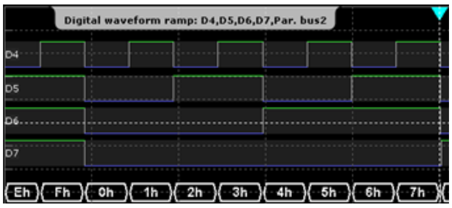
Remote command:
`BUS<m>:PARallel:DISPlay:SHDI` on page 1871

Show bus
If enabled, the resulting bus signal and bus values are displayed in the diagram. Select the presentation type for the bus signal with Bus representation.

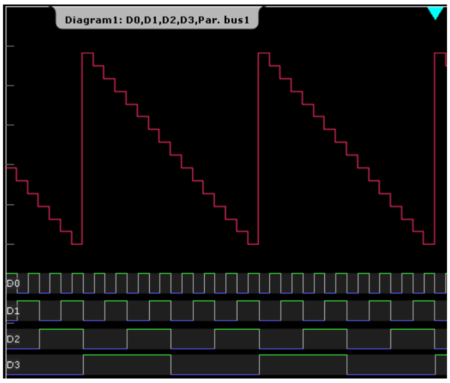
Remote command:
`BUS<m>:PARallel:DISPlay:SHBU` on page 1871

Bus representation
Defines how the parallel bus is displayed:

"Comb" Displays the decoded bus signal with bus values. When at least one digital channel changes its value, the bus value changes too.

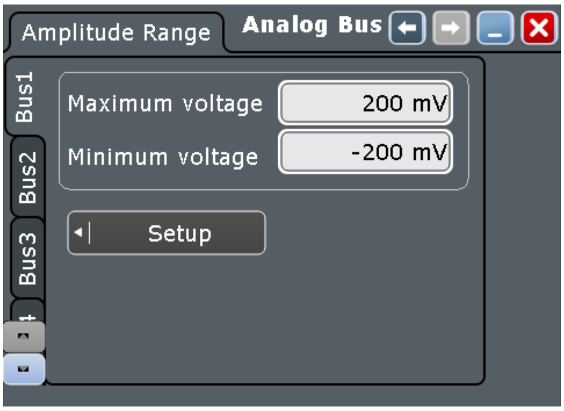


"Analog" Displays the bus values as signal amplitudes, similar to an analog waveform. Thus, a quasi-analog waveform is created.



Remote command:
`BUS<m>:PARallel:DISPlay:BTYP` on page 1871

Amplitude Range
If the bus representation is "Analog", the amplitude range defines the voltage range for the display of the analog bus. The highest bus value corresponds to the "Maximum voltage", and the lowest bus value to the "Minimum voltage".



See also: "Bus representation" on page 870

Clocked bus setup

If a bus is a clocked bus, one of the digital channels serves as clock of the bus.

For an unclocked bus, the logical state of the bus is determined for each sample. For a clocked bus, the logical state is determined only at the specified clock edges.

The settings are only available for "Bus1" and "Bus2".

"Enable clocked bus"	Enable this option, if the bus is a clocked bus.
"Show decode table"	The decode table is only available for clocked buses to check the data words. If enabled, a results box opens with decoded values of the bus signal and its time. Each clock edge corresponds to one row in the table.
"Clock source"	Selects the digital channel used as clock.
"Clock slope"	Selects the slope of the clock signal at which all digital channels of the bus are analyzed.

Remote command:
BUS<m>:PARallel:CLON on page 1872
BUS<m>:PARallel:CLOCk on page 1872
BUS<m>:PARallel:CLSLope on page 1872
BUS<m>:PARallel:DECTable<n>:SHOW on page 1873
BUS<m>:PARallel:DECTable<n>:COUNt? on page 1873
BUS<m>:PARallel:DECTable<n>:DATA? on page 1873

Data format

Sets the data format of bus values, which are displayed in the decode table and on the comb bus display. Available formats are: Hex, ASCII, Octal, Binary, Signed, and Unsigned.

Signed and Unsigned are integer data types with maximum 16-bit length. Unsigned is used for positive integers. Signed is used for positive and negative integers.

If the target file format is BIN, you can save only signed and unsigned binary data. The data format "Signed" writes signed data; all other formats are saved as unsigned binary data.

Remote command:
BUS<m>:PARallel:DATA:FORMat on page 1870

Signal selection
In the table, you select and configure the digital channels that are used in the selected bus.

"State"	Enables a digital channel, and assigns it to the bus.
"Label"	You can enter a name for each digital channel. The name is displayed in the diagram.
"Deskew"	Sets an individual delay for each digital channel to time-align it with other digital channels. The deskew value compensates delays that are known from the circuit specifics or caused by the different length of cables. The skew between the probe boxes of the digital channels and the probe connectors of the analog channels is automatically aligned by the instrument. You can also set a value that is applied to all digital channels, see "Deskew offset" on page 872.

Remote command:
BUS<m>:PARallel:BIT<n>[:STATe] on page 1867 (all buses)
DIGital<m>:DISPlay on page 1864 (Bus1)
BUS<m>:PARallel:BIT<n>:LABel on page 1870 (all buses)
DIGital<m>:LABel on page 1865 (Bus1)
BUS<m>:PARallel:BIT<n>:DESKew on page 1869 (all buses)
DIGital<m>:DESKew on page 1866 (Bus1)

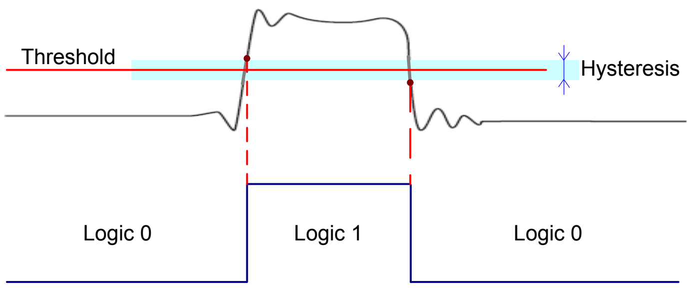
D0-D7, D8-D15 ← Signal selection
The buttons select or deselect all digital channels of a pod at once.

Deskew offset ← Signal selection
Sets a general delay for all digital channels. The resulting deskew of a digital channel is the sum of the general "Deskew offset" and the individual "Deskew".

Remote command:
BUS<m>:PARallel:DESoffset on page 1870

Threshold setup
Sets the logical threshold. For each sample, the instrument compares the input voltage with the threshold value. If the input voltage is above the threshold, the signal state "1" is stored. Otherwise, the signal state "0" is stored if the input voltage is below the threshold.

To avoid the change of signal states due to noise, a hysteresis is considered.



By default, same threshold and hysteresis value are used for all digital channels and all parallel buses: "Coupling" is enabled.

You can also set different thresholds for the individual channel groups: Disable "Coupling" and set the threshold value for each group. As long as the buses are disabled, you can set different thresholds for each bus. Active buses use the same threshold and hysteresis values, the settings of the last activated bus take effect.

The range of threshold levels and the minimum voltage swing is given in the data sheet.

"Threshold" Enter the value directly in the field.

"Technology" Selects the threshold voltage for various types of integrated circuits from a list and applies it to all digital channels. The value is set to "Manual" if a user-defined threshold was entered directly.

"Coupling" Sets the threshold and the hysteresis for all digital channels and all buses to the same value.

"Hysteresis" Defines the size of the hysteresis. Three values are available:
- Normal: the instrument sets a small value suitable for the signal and its settings. Use this setting for clean signals.
- Maximum: the instrument sets the maximum value that is possible and useful for the signal and its settings. Use this setting for noisy signals.
- Robust: sets different hysteresis values for falling and rising edges to avoid an undefined state of the trigger system. Use this setting for very noisy signals.

Remote command:
`BUS<m>:PARallel:TECHnology` on page 1868 (all buses)
`DIGital<m>:TECHnology` on page 1864 (bus1)
`BUS<m>:PARallel:THReshold<n>` on page 1867 (all buses)
`DIGital<m>:THReshold` on page 1864 (bus1)
`BUS<m>:PARallel:THCoupling` on page 1868 (all buses)
`DIGital<m>:THCoupling` on page 1865 (bus1)
`BUS<m>:PARallel:HYSTeresis<n>` on page 1869 (all buses)
`DIGital<m>:HYSTeresis` on page 1865 (bus1)

13.1.2 Parallel Buses - Digital Probes

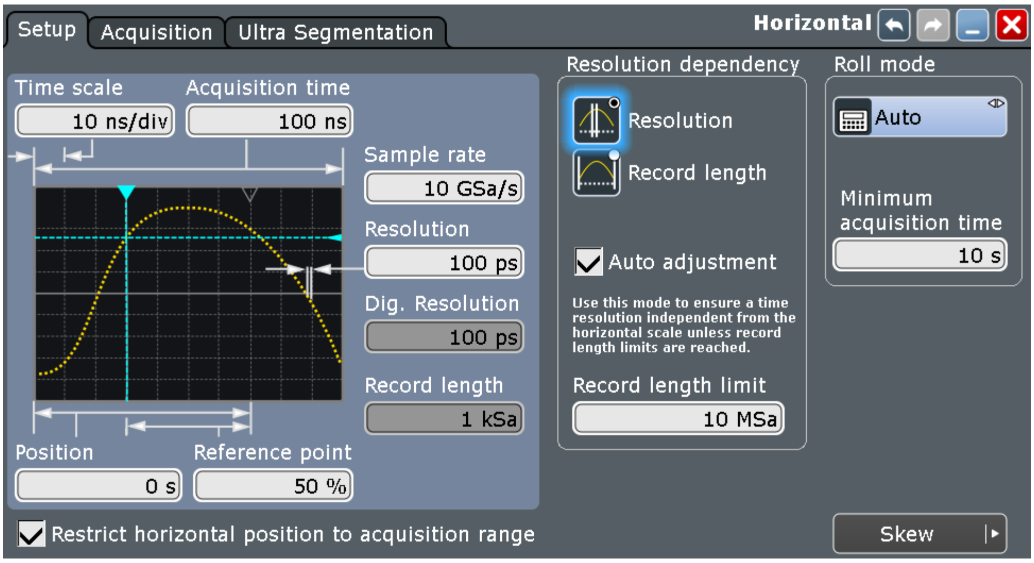
Access: "Analysis" menu > "Parallel buses" > "Digital Probes" tab

Logic probes provided by R&S are recognized by the instrument. The fields show the characteristics of each recognized probe (pod) for information. "Write EEPROM" and "Flash it" are service functions.

13.1.3 Digital Resolution

Access: [RES REC LEN] key

If an MSO option is installed and at least one digital channel is active, additional information appears on the "Setup" tab of the "Horizontal" dialog box.



Dig. resolution
Shows the current digital resolution of the digital channels. The maximum digital record length is always 200 MSa per digital channel. This number is independent of additionally installed memory.

Remote command:
ACQuire:DRESolution? on page 1873

13.1.4 Using Digital Probes

NOTICE

Ensuring accurate measurement results

The mixed-signal-option (MSO, R&S RTE-B1) with connected probe leads is considered as a test probe, according to EN 61326-2-1, clause 5.2.4.101. Therefore, the measurements are sensitive to electromagnetic interference. Consider additional shielding methods to avoid interference.

Consider the following guidelines for good probing practices:

- The ground lead from each digital channel group (D15–D8 and D7–D0) should be attached to the ground of the device under test if any channel within the group is being used for data capture. The ground lead improves signal fidelity to the oscilloscope, ensuring accurate measurements.
- For high-speed timing measurements (rise time < 3 ns), each digital channel probe should use its own ground.

1. Connect the digital probe cable to any of the MSO connectors on the rear panel of the instrument as shown on the Documentation Card delivered with the digital probe.
2. Connect the ground lead on each set of channels (each pod) with a probe grabber.
3. Connect a grabber to one of the probes leads.
4. Connect the grabber to a node in the circuit you want to test.
5. For high-speed signals, connect a ground lead to the probe lead. Connect the ground lead to ground in the device under test.
6. Repeat these steps until you have connected all points of interest.

13.1.5 Configuring Digital Channels and Parallel Buses

The configuration of a parallel bus includes the selection and setup of the digital channels, the configuration of the bus display, and, if necessary, the clock configuration.

For a detailed description of the settings, see Chapter 13.1.1, "Parallel Buses - Configuration", on page 868.

1. On the "Analysis" menu, tap "Parallel buses".
2. In the "State" column of the "Signal selection" table, enable the digital channels to be displayed and included in the bus.
   To enable or disable all channels of a pod at once, tap "D0-D7" or "D8-D15".

   Enabling one or more channels also enables the display of the signals - "Show dig. signals", and enables the parallel bus. If another active bus already uses the same digital channel, the instrument disables this bus and shows a message.

The digital signals are shown in the diagram, and the signal icon of the parallel bus appears on the signal bar. Using this bus icon, you can minimize, arrange, and switch off the bus together with its channels in the same way as you do with any waveform.

3. Optionally, you can enter a "Label" for each digital channel, and a "Deskew" value to time-align the channel.
4. Set the logical thresholds as described in Chapter 13.1.6, "Setting the Logical Thresholds", on page 876.
5. If the bus has a clock signal, enable "Bus clocked" and select the "Clock source" and "Clock slope".

   Now the configuration of the parallel bus is completed.

13.1.6 Setting the Logical Thresholds

For a detailed description of the settings, see "Threshold setup" on page 872. Threshold settings are the same for all *active* parallel buses.

1. On the "Analysis" menu, tap "Parallel buses".
2. To set the thresholds, use one of the following ways:
   - Use the same value for all digital channels and all parallel buses: Enable "Coupling" and set one threshold value, either select a predefined "Technology" value or enter a user-defined value.
   - Set different thresholds for the individual channel groups: Disable "Coupling" and set the threshold value for each group. As long as the buses are disabled, you can set different thresholds for each bus. Active buses use the same threshold and hysteresis values.
3. Set the "Hysteresis" for each threshold to avoid the change of signal states due to noise.

13.2 Display

You can adjust the display of the parallel bus signals and the individual digital channels to optimize the analysis of bus data:

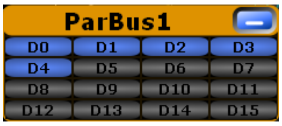
- Show the digital channels which are assigned to the bus, drag them to the optimal position, and scale them
- Show the decoded bus signal in different ways:
  – comb display with numeric bus values
  – analog display with bus values as amplitudes (quasi-analog waveform)

  You can also drag the bus waveforms on the display and scale them.
- Show the result box of the decoded clocked bus signal

Each parallel bus is shown in a separate diagram, and the diagrams can be minimized and arranged as usual.

The signal icon indicates the activities on the digital channels even if they are not displayed in the diagram, of if the acquisition has been stopped:

- Blue: channel is low
- Green: channel is high
- Gray: channel state is changing



The display update rate of the oscilloscope is adapted to the visual perception of human eyes, and it is slower than the acquisition rate. All analog and digital waveforms that are acquired during one display update cycle are overlapped and displayed at once. Thus you can see the cumulative occurrence of binary states and edge transitions on the screen at once. Bus signals are not overlapped.

The trigger point is always visible on the display, it cannot be moved outside ("Restrict horizontal position to acquisition range" is enabled automatically).

If digital channels are active, the trigger point is always visible on the display, it cannot be moved outside.

To access and analyze one or more specific acquisitions, you can use the History Viewer in the common way.

Furthermore, you can zoom in digital signals and bus signal in the same way as in analog waveforms.

See also:

- Chapter 6.4, "History", on page 273
- Chapter 6.1, "Zoom", on page 240

13.2.1 Parallel Bus - Decode Table

Decoding is available for clocked parallel buses.

The decode table shows the decoded data words of the bus signal and the corresponding time. Each clock edge corresponds to one row in the table. Below the table, you can select the data format of the bus values.

The results can be saved to a `.csv` or `.html` file, see Chapter 11.2.4, "Numeric Results", on page 452.

13.2.2 Adjusting the Display of Digital Channels and Parallel Buses

The display of digital channels and parallel buses is flexible, you can adjust it to your needs by combining the following settings:

1. Enable "Show bus" if you want to display the bus signal in the diagram. Under "Bus representation", select if you want to display the decoded bus signal with bus values ("Comb"), or show the bus values as amplitudes, similar to an analog waveform ("Analog").
2. Check the signal icon of the bus to monitor the activities on the digital channels even if they are not displayed in the diagram:
   - Blue: channel is low
   - Green: channel is high
   - Gray: channel state is changing
3. In the diagram, you can change the display order of the digital channels by dragging the individual channels to the required position.
4. To adjust the line height and vertical position of all digital channels at once, tap one of the digital channels and turn the vertical [SCALE] and [POSITION / OFFSET] rotary knobs. In the same way, you can move and scale the bus signal.
5. If the bus signal is displayed as quasi-analog waveform, you can double-tap the waveform to open the "Parallel buses" dialog box.
6. To switch off the display of the digital channels, disable "Show signals".

13.3 Trigger

For digital trigger sources are all trigger types useful that require only one trigger level as trigger condition. This level is the logical threshold. Possible trigger sources are the individual digital channels, parallel bus signals, or any logical combination of digital channels. The following trigger types are available:

Table 13-1: Trigger types and digital trigger sources

Trigger type	Trigger source is		
	Digital channel	**Logic combination of digital channels**	**Parallel bus**
Edge	X	X	
Width	X	X	
Timeout	X	X	
Data2Clock	X		X
State		X	X
Pattern (with holdoff)		X	X
Serial Pattern	X	X	

For details, see: Chapter 13.3.1, "Trigger Settings for Digital Signals and Parallel Buses", on page 879.

Additionally, you can define trigger holdoff conditions. See also Chapter 5.4, "Holdoff", on page 226.

13.3.1 Trigger Settings for Digital Signals and Parallel Buses

Depending on the selected source, the instrument provides the appropriate trigger types and the corresponding trigger settings.

The settings in the "Setup" tab are:

- Basic Trigger Settings....879
- Edge....880
- Width....881
- Timeout....882
- Data2Clock....883
- State....884
- Pattern....885
- Serial Pattern....888

13.3.1.1 Basic Trigger Settings

The basic trigger settings for MSO are the trigger source and the trigger type. They are selected in the upper part of the "Trigger" dialog box.

Make sure that the trigger sequence is set to "A only".

Also, you can define trigger holdoff conditions. See also Chapter 5.4, "Holdoff", on page 226.

Source
If the Mixed Signal Option is installed, the variety of trigger sources of the A-event setup is enhanced with specific digital trigger sources. You can select as trigger source:

- One of the digital channels "D0" ... "D15"

- A logic combination of digital channels: "Logic"

- One of the parallel buses "Par. bus1" ... "Par. bus4"

Remote command:
`TRIGger<m>:SOURce[:SELect]` on page 1128

Type

Depending on the selected source, the instrument provides the appropriate trigger types and the corresponding trigger settings. For mixed signal analysis, the following trigger types are available:

- Edge, see page 880
- Width, see page 881
- Timeout, see page 882
- Data2Clock, see page 883
- State, see page 884
- Pattern, see page 885
- Serial Pattern, see page 888

Remote command:
TRIGger<m>:PARallel:TYPE on page 1876

13.3.1.2 Edge

Using the edge trigger, you can also trigger on a single digital channel (a logic bit), and a logical combination of digital channels.

Depending on the selected trigger source, different trigger settings are available. The trigger level is already set - in MSO the logical threshold is used as trigger level.

Figure 13-1: Edge trigger settings for trigger source = logical combination of digital channels (Logic)

Slope

Defines the edge - the state transition - of the signal.

"Rising"	Means a 0 to 1 transition of the state.
"Falling"	Means a 1 to 0 transition of the state.
"Either"	Triggers on any activity on the selected trigger source.

Remote command:
TRIGger<m>:PARallel:EDGE:SLOPe on page 1877

Logical expression

Defines a logical combination of several digital channels as trigger condition if "Logic" is set for "Source".

If the "Slope" is rising, the trigger occurs when the logical expression comes true. If the "Slope" is falling, the trigger occurs when the logical expression comes false.

Remote command:
TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine] on page 1876

13.3.1.3 Width

The width trigger detects positive and/or negative pulses of a pulse width (duration) inside or outside of a defined time limit. It can trigger on a single digital channel or a logical combination of digital channels.

The instrument triggers at the end of the detected pulse.

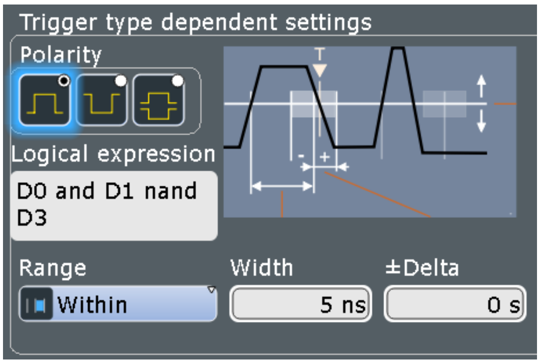


Figure 13-2: Width trigger settings for trigger source = logical combination of digital channels

Polarity
Sets the polarity of a pulse to "Positive", "Negative", or "Both".

When triggering on a positive pulse, the trigger event occurs on the high to low transition of the pulse if the timing condition is true. When triggering on a negative pulse, the trigger event occurs on the low to high transition of the pulse if the timing condition is true.

Remote command:
`TRIGger<m>:PARallel:WIDTh:POLarity` on page 1877

Logical expression
Defines a logical combination of several digital channels as trigger condition if "Logic" is set for "Source". As long as the digital signals match the logical expression (true), the pulse is positive. Otherwise, the pulse is negative.

Remote command:
`TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]` on page 1876

Range
Selects how the range of a pulse width is defined:

"Within"	Triggers on pulses inside a given time range. The time limit is defined by *Width ± Delta*.
"Outside"	Triggers on pulses shorter or longer than a given time range. The time limit definition is the same as for "Within" range.
"Shorter"	Triggers on pulses shorter than the given "Width".
"Longer"	Triggers on pulses longer than the given "Width".

Remote command:
`TRIGger<m>:PARallel:WIDTh:RANGe` on page 1877

Width
For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits "±Delta".

Remote command:
`TRIGger<m>:PARallel:WIDTh:WIDTh` on page 1878

±Delta
Defines a range around the given width value.

The combination "Range" = Within and "±Delta" = 0 triggers on pulses with a pulse width that equals "Width".

The combination "Range" = Outside and "±Delta" = 0 means to trigger on pulse widths ≠ "Width".

Remote command:
`TRIGger<m>:PARallel:WIDTh:DELTa` on page 1878

13.3.1.4 Timeout

The timeout trigger event checks if the trigger source signal stays above or below the threshold voltage for a specified time lapse. In other words, the event occurs if the state condition remains unchanged for the specified time.

You can use the timeout trigger on a single digital channel, or a logical combination of digital channels.

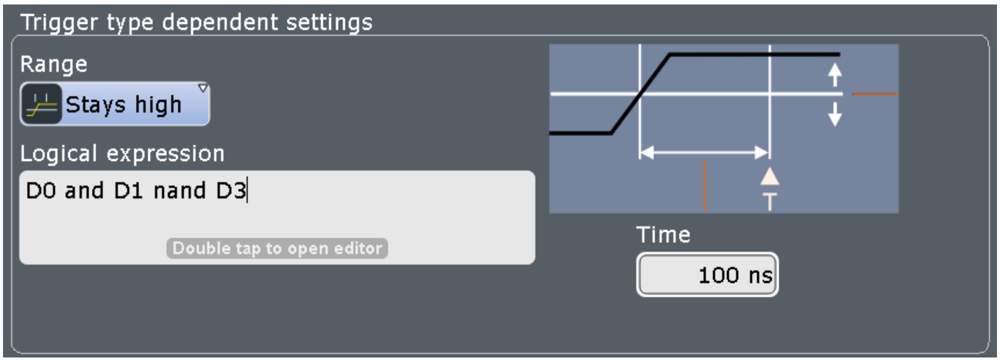


Figure 13-3: Timeout trigger settings for trigger source = logical combination of digital channels

Range
Sets the state condition:

"Stays high"	The level of a digital channel stays above the threshold, or the logical expression for "Logic" trigger source is true.
"Stays low"	The level of a digital channel stays below the threshold, or the logical expression for "Logic" trigger source is false.
"High or low"	The signal state remains unchanged.

Remote command:
TRIGger<m>:PARallel:TIMeout:RANGe on page 1878

Time
Defines the time limit for the timeout at which the instrument triggers.

Remote command:
TRIGger<m>:PARallel:TIMeout:TIME on page 1879

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine] on page 1876

13.3.1.5 Data2Clock

The Data2Clock trigger event occurs when the state of the trigger source signal changes inside a given time before the clock edge (setup time) or after the clock edge (hold time). This trigger type is also known as setup/hold trigger. The trigger event occurs at the clock edge for which the setup and/or hold time was violated.

With Data2Clock trigger, you can trigger on a single digital channel, or a parallel bus to check several digital channels simultaneously. The clock signal is connected to one of the digital channels.

If you configure this trigger type for a parallel bus, the bus configuration is adjusted by the instrument if necessary. The bus is defined as clocked bus, and the clock source of the trigger is set as clock source of the bus.

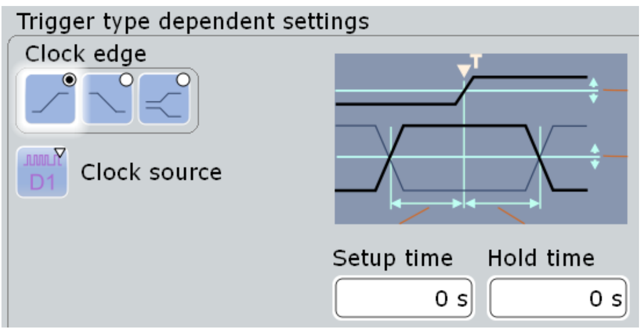


Figure 13-4: Data2clock trigger settings

Clock source
Selects the digital channel of the clock signal.

Remote command:
TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue] on page 1876
TRIGger<m>:PARallel:STATe:CSOurce:VALue on page 1876
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue] on page 1876

Clock edge
Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time reference point for the setup and hold time measurement.

Remote command:
TRIGger<m>:PARallel:DATatoclock:CSOurce:EDGE on page 1879

Setup time
Sets the minimum time *before* the clock edge while data is stable and not change its state.

The setup time can be negative. In this case, the setup interval starts after the clock edge, and the hold time starts after the setup time has expired. Thus, the hold time is always positive. If you change the negative setup time, the hold time is adjusted by the instrument.

Remote command:
TRIGger<m>:PARallel:DATatoclock:STIMe on page 1879

Hold time
Sets the minimum time *after* the clock edge while data is stable and not change its state.

The hold time can be negative. In this case, the hold time ends before the clock edge, and the setup interval ends when the hold interval starts. Thus, the setup time is always positive. If you change the negative hold time, the setup time is adjusted by the instrument.

Remote command:
TRIGger<m>:PARallel:DATatoclock:HTIMe on page 1880

13.3.1.6 State

The state trigger detects the logical state of several logically combined digital channels at a given clock edge. The trigger source is a logical combination of digital channels or a parallel bus. The trigger occurs at the clock edge at which the state condition is true.

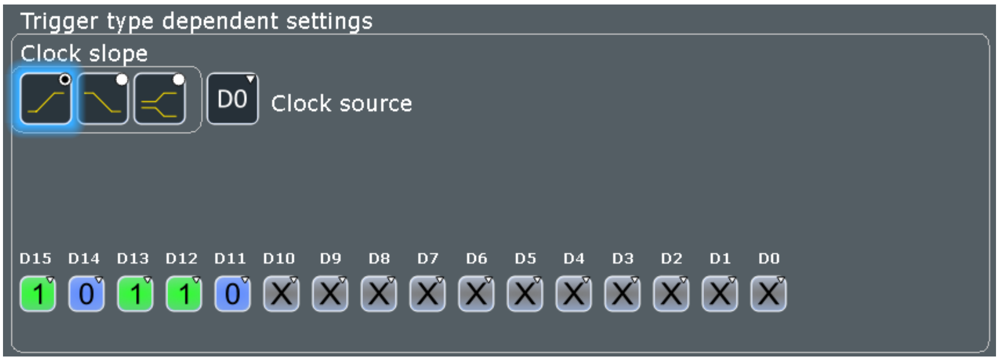


Figure 13-5: State trigger settings for trigger source = parallel bus

Clock source
Selects the digital channel of the clock signal.

Remote command:
TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue] on page 1876
TRIGger<m>:PARallel:STATe:CSOurce:VALue on page 1876
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue] on page 1876

Clock edge
Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time at which the logical states and the bus value are analyzed.

Remote command:
TRIGger<m>:PARallel:STATe:CSOurce:EDGE on page 1880

Channel states
For each digital channel that is used in the bus, set the required state: 1, 0, or X (don't care).

Remote command:
TRIGger<m>:PARallel:STATe:BIT<0..15> on page 1880

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine] on page 1876

13.3.1.7 Pattern

The pattern trigger identifies a logical state of several logically combined digital channels (pattern) and a time limitation (holdoff). The pattern definition is defined by the logical expression, if "Logic" is used for trigger source. For a parallel bus trigger source, the pattern is defined by setting the state of each digital channel.

The timing starts when the pattern comes true. The decision level is the logical threshold.

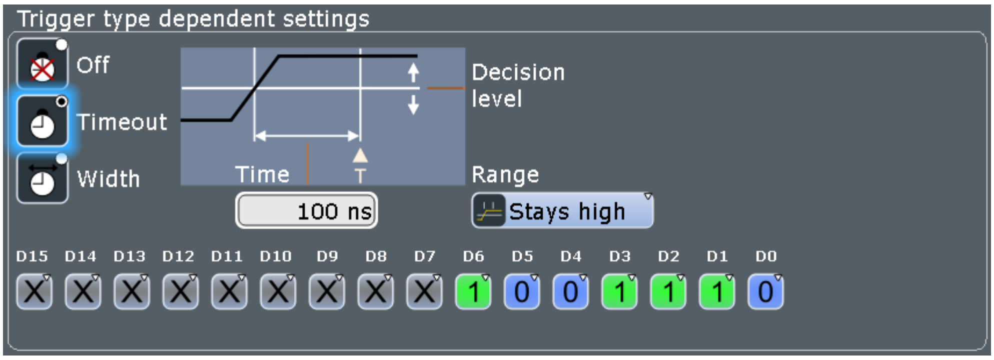


Figure 13-6: Pattern trigger settings for trigger source = parallel bus and timeout

Channel states

For each digital channel that is used in the bus, set the required state: 1, 0, or X (do not care).

Remote command:
`TRIGger<m>:PARallel:PATTern:BIT<0..15>` on page 1881

Logical expression

Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
`TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 1876
`TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 1876
`TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]` on page 1876
`TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 1876

Timing mode: Off, Timeout, Width

Sets the mode of the timing condition.

"Off" No timing condition, only the logical pattern condition is relevant.

"Timeout" Defines a minimum time qualification to avoid triggering on unstable or transitional conditions. Even in best-designed systems, there are slight delays between the signal when digital signals change states. This means that there are always transitional state conditions when signals are switching.
See "Timeout settings" on page 887 for a description of the settings. The trigger event occurs when the pattern stays unchanged for the specified time.

"Width" Sets a pulse width as timing condition, see "Width settings" on page 887. The pulse starts when the pattern comes true, and the trigger event occurs when the pattern comes false during the specified time limit.
Using this mode, you can, for example, trigger exclusively on unstable conditions - if the pattern is present for less than a specified time.

Remote command:
TRIGger<m>:PARallel:PATTern:MODE on page 1881

Timeout settings
The timeout settings "Range" and "Time" appear if the timing mode is set to "Timeout".

Range ← Timeout settings
Sets the state condition:

"Stays high"	The pattern stays true for the specified time.
"Stays low"	The pattern stays false for the specified time.
"High or low"	The pattern remains unchanged for the specified time.

Remote command:
TRIGger<m>:PARallel:PATTern:TIMeout:MODE on page 1881

Time ← Timeout settings
Defines the time limit for the timeout at which the instrument triggers.

Remote command:
TRIGger<m>:PARallel:PATTern:TIMeout[:TIME] on page 1882

Width settings
The width settings "Range", "Width" and "±Delta" appear if the timing mode is set to "Width".

Range ← Width settings
Selects how the range of a pulse width is defined:

"Within"	Triggers when the pattern comes false inside a given time range. The time limit is defined by *Width ± Delta*.
"Outside"	Triggers when the pattern comes false before or after the given time range. The time limit definition is the same as for "Within" range.
"Shorter"	Triggers when the pattern comes false before the given "Width" has expired.
"Longer"	Triggers when the pattern comes false after the given "Width" has expired.

Remote command:
TRIGger<m>:PARallel:PATTern:WIDTh:RANGe on page 1882

Width ← Width settings
For the ranges "Shorter" and "Longer", the width defines the maximum and minimum time limit, respectively.

For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits "±Delta".

Remote command:
TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh] on page 1882

±Delta ← Width settings
Defines a range around the given width value.

The combination "Range" = Within and "±Delta" = 0 triggers on pulses with a pulse width that equals "Width".

The combination "Range" = Outside and "±Delta" = 0 means to trigger on pulse widths ≠ "Width".

Remote command:
TRIGger<m>:PARallel:PATTern:WIDTh:DELTa on page 1883

13.3.1.8 Serial Pattern

The serial pattern trigger identifies a serial bit string trigger on a single digital channel, or for a logical combination of digital channels. The trigger requires a clocked bus; the bits are read at the specified clock edge. The trigger event occurs at the last clock edge of the serial bit string.

This trigger type allows you to trigger on specific address or data transmissions in serial input and output signals.

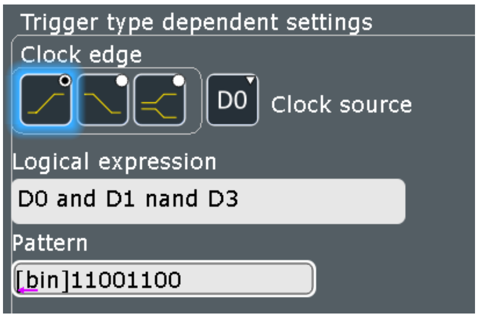


Figure 13-7: Serial pattern trigger settings for trigger source = logical combination of digital channels

Clock edge
Sets the edge of the clock signal. The bit value is determined at the crossing of the clock edge and the logical threshold.

Remote command:
TRIGger<m>:PARallel:SPATtern:CSOurce:EDGE on page 1883

Clock source
Selects the digital channel of the clock signal.

Remote command:
TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue] on page 1876
TRIGger<m>:PARallel:STATe:CSOurce:VALue on page 1876
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue] on page 1876

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine] on page 1876

TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine] on page 1876
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine] on page 1876

Pattern
Defines the serial bit string on which to trigger. Touch and hold the "Pattern" field to open the "Bit Pattern Editor" where you can enter the pattern in various formats. The pattern has to be defined exactly, X (do not care) is not supported in binary format.

See also: Chapter 12.1.5, "Bit Pattern Editor", on page 481

Remote command:
TRIGger<m>:PARallel:SPATtern:PATTern on page 1883

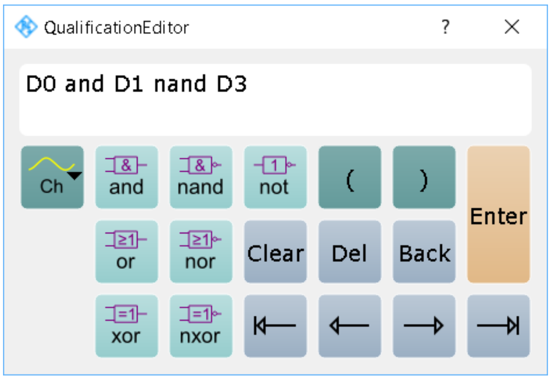
13.3.2 Triggering on Digital Signals and Parallel Buses

For a detailed description of the settings, see Chapter 13.3.1, "Trigger Settings for Digital Signals and Parallel Buses", on page 879.

1. Press the [TRIGGER] key and select the "Setup" tab.
2. Select the trigger "Source":
   - One of the digital channels "D0" ... "D15"
     

   - A logic combination of digital channels: "Logic"
     
   - One of the parallel buses "Par. bus1" ... "Par. bus4"
     

3. Select the trigger "Type".
4. Under "Trigger type dependent settings", configure the trigger.
5. For trigger source "Logic", enter the logical expression of the digital channel combination. Tap and hold the "Logical expression" field until the "Qualification Editor" opens. It provides all logic operators that can be used in the expression.



13.4 Measurements on Digital Channels

For measurements on digital channels, the number of measurement categories and types is reduced to applicable measurement.

These are:

- Period
- Frequency
- Positive and negative pulse
- Pulse count
- Delay
- Phase
- Positive and negative duty cycle
- Burst width
- Edge count

Except for delay measurement, the measurements have the same settings and results for analog and digital sources.

Delay measurement on digital channels is reduced to measure the time between two subsequent rising or two subsequent falling edges.

See also Chapter 7.2.5, "Amplitude/Time Measurements", on page 310.

13.5 Data Export

The data of digital channels and parallel buses can be saved to file in the same way as analog waveform data. One digital channel or bus per file can be saved.

The data format of the stored values is defined with "Analysis" > "Parallel Bus" > "Setup" tab >"Data format". If the data is written to XML or CSV files, the selected format is used. If the target file format is BIN, you can save signed and unsigned binary data. The data format "Signed" writes signed data; all other formats are saved as unsigned binary data.

The xport of raw data is not available.

Export of a digital channel

If the data of digital channels is stored in BIN format, 1 bit is written for each sample. 8 data samples are written in 1 byte (data word). Thus, the file size is

File size = Number of samples / 8

For example, 100 MSa are written into a 12.5 MByte BIN file. After reading the file, you have to extract the samples from the data words.

If saved to BIN file, the digital channel can be imported as reference waveform.

Export of a parallel bus

A parallel bus can be exported to file if "Enable bus" and "Show bus" are both activated.

All data formats can be saved to XML, CSV, and BIN files. If you save binary format to XML or CSV, you can see the values of each line for each sample.

In BIN files, 4 Bytes are written for each sample.

Importing parallel buses from BIN files is only possible if the bus was saved with quasi-analog bus representation.

See also:

- Chapter 11.2.7, "Saving and Loading Waveform Data", on page 457
- Chapter 11.2.2, "Waveforms - Export Settings", on page 445

Remote commands for export to file:
- `EXPort:WAVeform:SOURce` on page 1363
- `EXPort:WAVeform:NAME` on page 1364
- `EXPort:WAVeform:SAVE` on page 1365

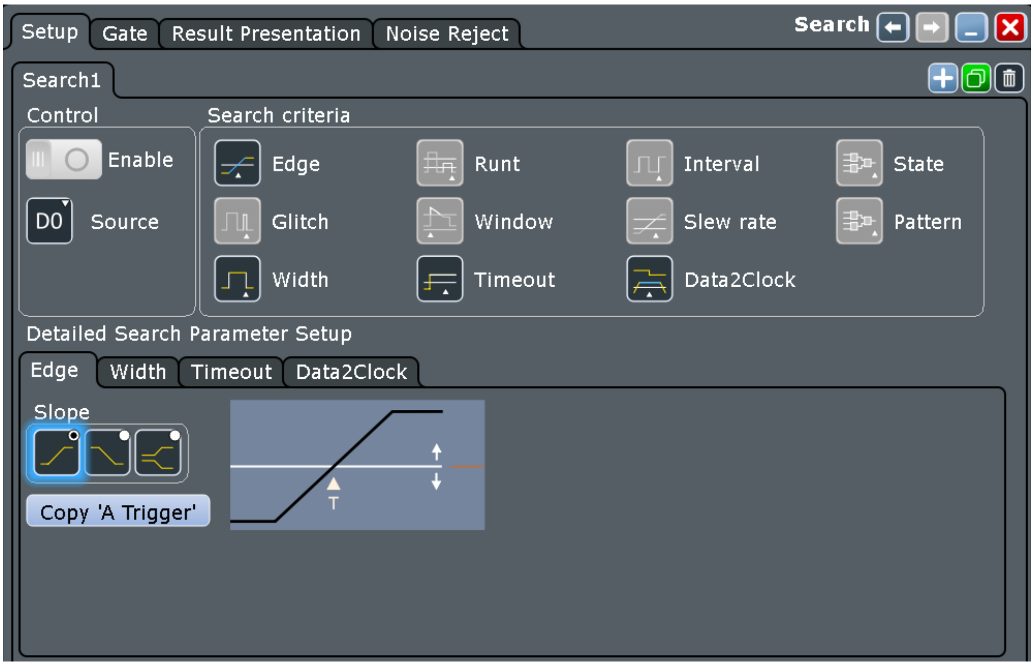
Remote commands for remote data transfer:
- `BUS<m>:PARallel:DATA:FORMat` on page 1870
- `BUS<m>:PARallel:DATA:HEADer?` on page 1885
- `BUS<m>:PARallel:DATA[:VALues]?` on page 1885
- `DIGital<m>:DATA:HEADer?` on page 1884
- `DIGital<m>:DATA[:VALues]?` on page 1884

13.6 Mathematics

A parallel bus that is displayed as quasi-analog waveform can be analyzed with FFT. To configure the FFT, use the "Advanced" mode and the formula editor.

13.7 Search

Access: [Search] > "Setup" tab.



It is also possible to search on digital channels for specified events. Search conditions use the same parameters as the trigger event definition, see Chapter 13.3.1, "Trigger Settings for Digital Signals and Parallel Buses", on page 879.

You can search for edge, width, timeout, and Data2Clock conditions.

14 Waveform Generator (Option R&S RTE-B6)

The R&S RTE includes an integrated waveform generator which can generate input signals and patterns during testing.

You can setup and output two waveform generators and a pattern generator. It is possible to couple and synchronize the settings of the different generators and enable them on a trigger signal. With each of the waveform generators, one can output simple functions, modulated sine waveform, arbitrary waveforms and sweep waveforms.

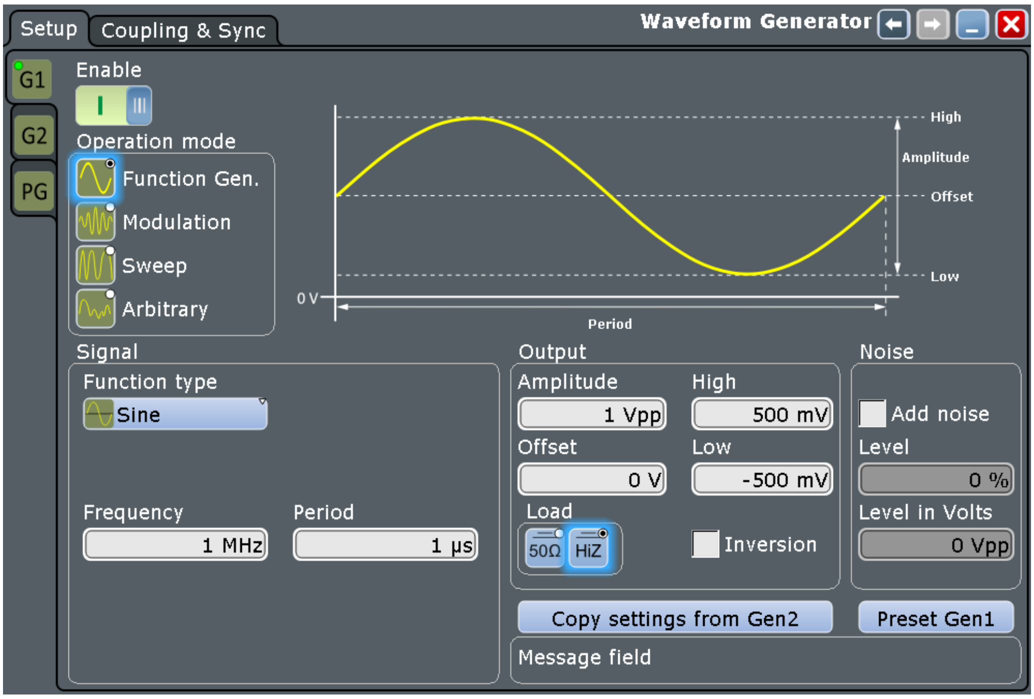
Always use double-shielded cables to prevent electromagentic interference (EMI) from impairing measurement results.

14.1 Setup of the Waveform Generator

Access: [Gen 1]/[Gen 2] > "Setup" tab

The "Setup" tab is divided into several sections:

- General settings like enabling and presetting the waveform generator
- Signal settings, depending on the selected Operation mode refer to one of the following:
  - Chapter 14.1.2, "Function Generator", on page 895
  - Chapter 14.1.3, "Modulation", on page 899
  - Chapter 14.1.4, "Sweep", on page 905
  - Chapter 14.1.5, "Arbitrary", on page 906
- "Output", including settings for defining the output see Chapter 14.1.6, "Output", on page 910
- "Noise", settings for adding noise to the waveform, see Chapter 14.1.7, "Noise", on page 911



Make sure that the tab of the correct waveform generator is selected on the left side.

The settings of the waveform generators are not affected by an instrument preset. Press "Preset Gen1/2" to preset the settings of the corresponding waveform generator.

14.1.1 General Settings

The general waveform generator settings are for enabling and presetting the generator and selecting the "Operation Mode".

Enable

Enables the waveform generator/ pattern generator and outputs the signal to the connectors.

Remote command:

WGENerator<m>[:ENABle] on page 1886

PGENerator:ENABle on page 1902

Operation mode

Selects the operation mode for the waveform generator. The "Signal" settings depend on the selected mode.

For the settings of the different operation modes, refer to:

- Chapter 14.1.2, "Function Generator", on page 895
- Chapter 14.1.3, "Modulation", on page 899
- Chapter 14.1.4, "Sweep", on page 905

- Chapter 14.1.5, "Arbitrary", on page 906

Remote command:
`WGENerator<m>:SOURce` on page 1886

Copy settings from Gen1/Gen2
Copies all settings from Gen1/Gen2 and applies them to Gen2/Gen1.

Remote command:
`WGENerator<m>:ACOPy` on page 1886

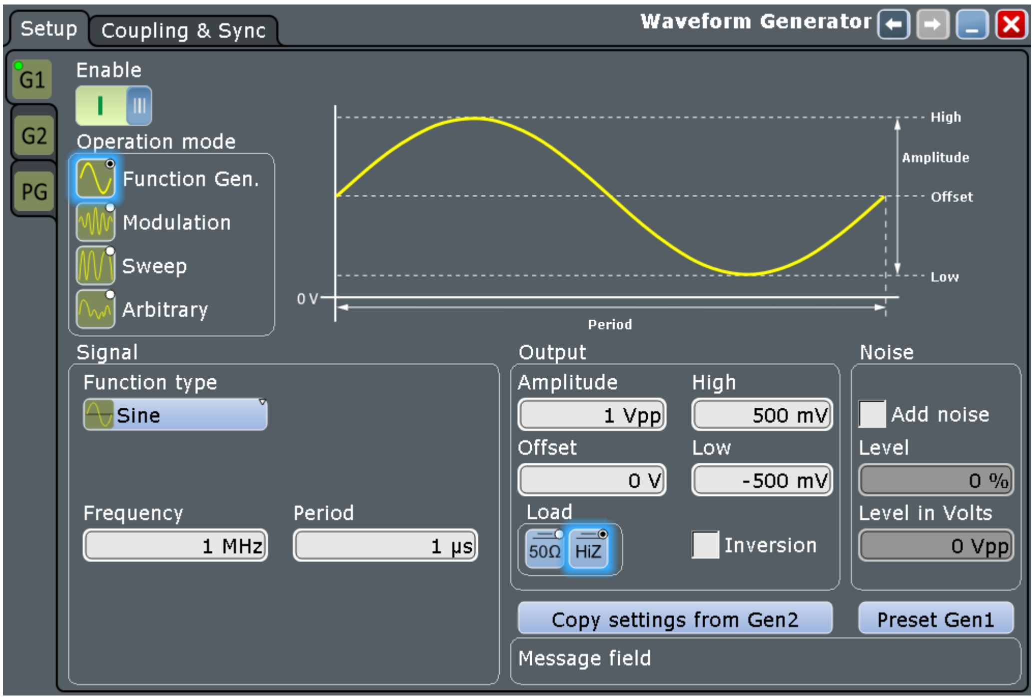
Preset Gen1/Gen2/Patt Gen
Sets the parameters of the generator to their default values. The settings of the generators are not affected by an instrument preset.They are also not stored in the user-defined preset.

Remote command:
`WGENerator<m>:PRESet` on page 1886
`PGENerator:PRESet` on page 1903

Message
Displays relevant messages concerning the coupling and sync settings.

14.1.2 Function Generator

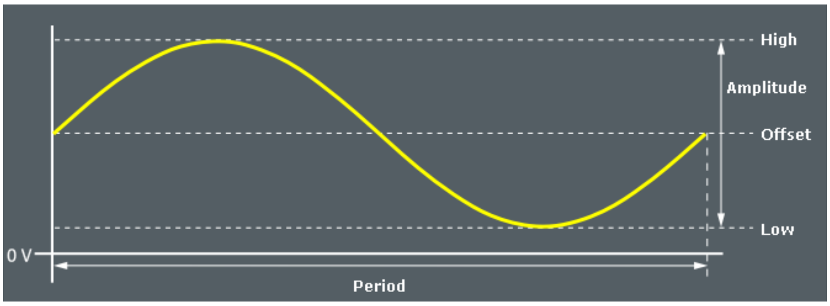
The R&S RTE function generator can generate input signals during testing. These signals can be used, for example, when testing circuits.



Function type
Selects the type of waveform to be generated for the function generator.

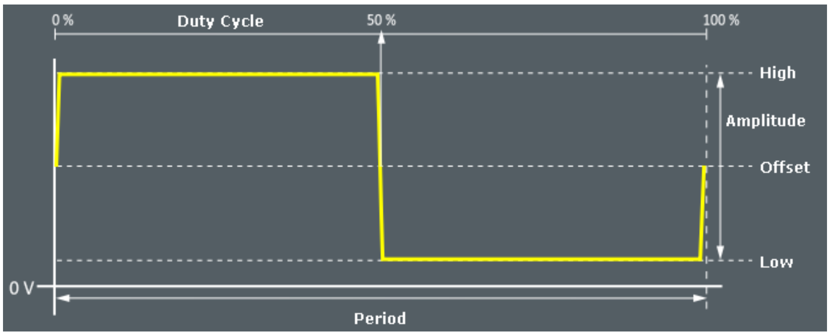
"Sine"

Generates a sine wave. You can set its Frequency and Period.



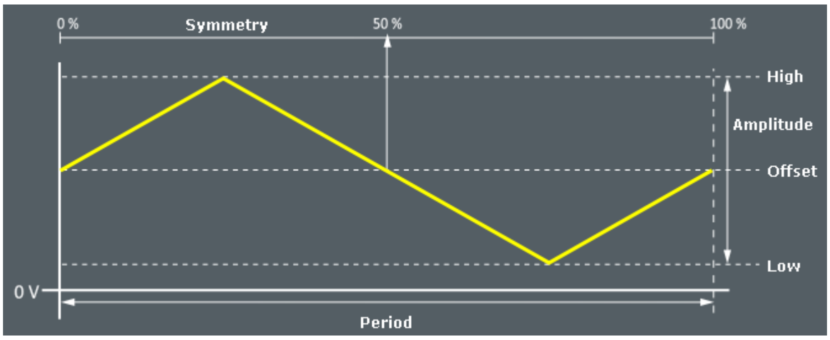
"Square"

Generates a square wave. You can set its Frequency, Period and Duty cycle.



"Ramp"

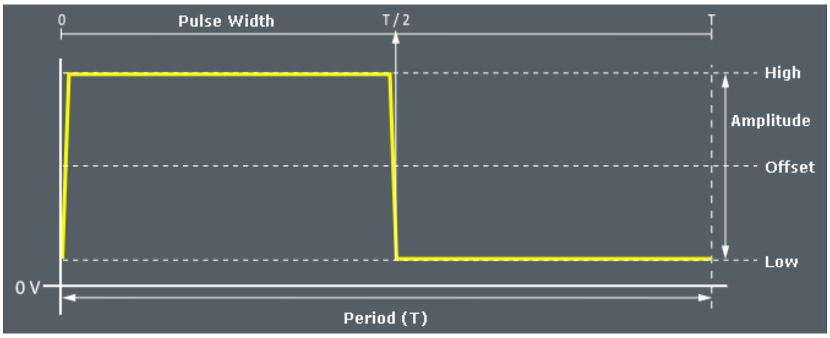
Generates a ramp signal. You can set its Frequency, Period and Symmetry.



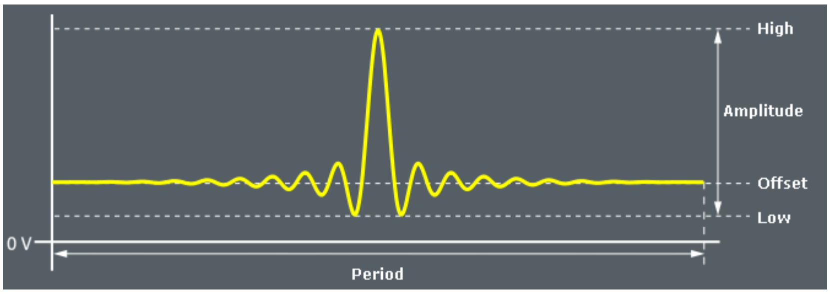
"DC"

Generates a direct current (DC) signal. You can set the DC Level.

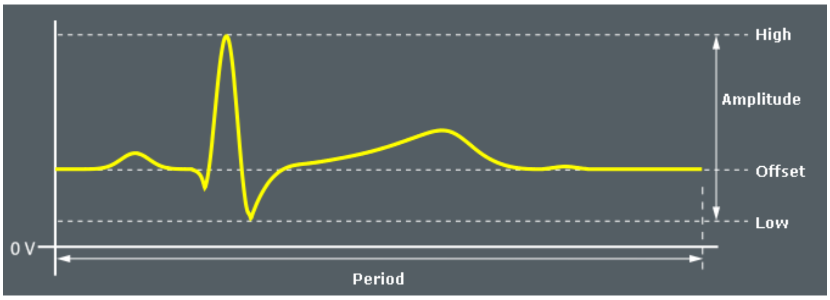
"Pulse" Generates a pulse signal. You can set the Frequency, Period and Pulse width.



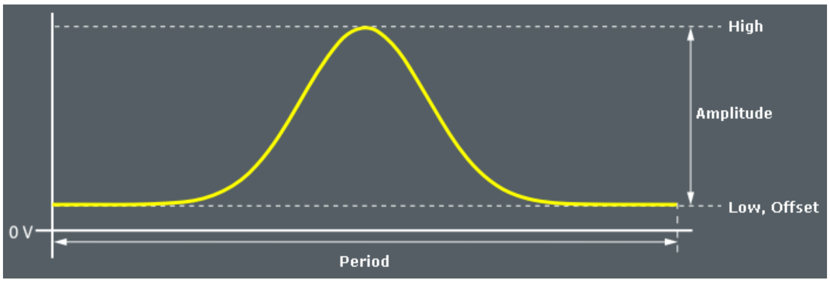
"Cardinal sine" Generates a cardinal sine wave. You can set the Frequency and Period.



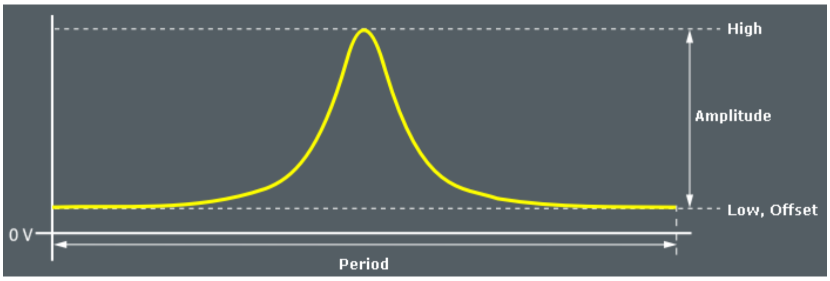
"Cardiac" Generates a cardiac signal. You can set the Frequency and Period.



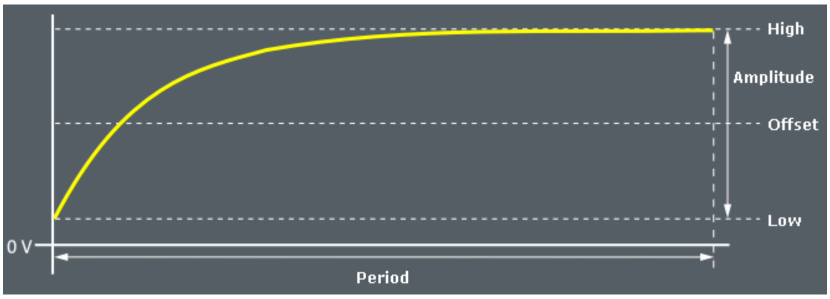
"Gauss" Generates a gauss signal . You can set the Frequency and Period.



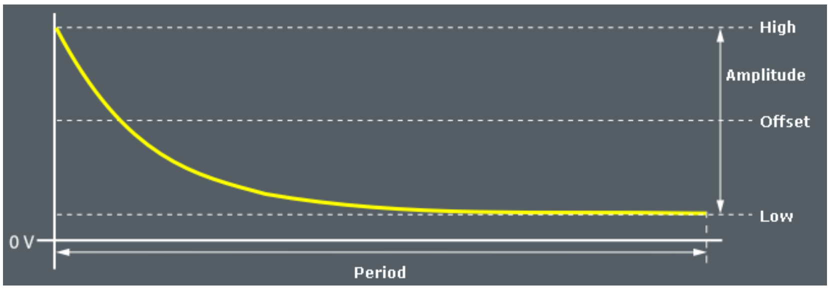
"Lorentz" Generates a Lorentz signal . You can set the Frequency and Period.



"Exp. rise" Generates an exponential rise signal. You can set the Frequency and Period.



"Exp. fall" Generates an exponential fall signal. You can set the Frequency and Period.



Remote command:
`WGENerator<m>:FUNCtion[:SELect]` on page 1887

Frequency

Sets the frequency of the waveform. The available frequency range depends on the selected "Function Type", see Frequency range of the function generator waveforms.

Table 14-1: Frequency range of the function generator waveforms

"Function type"	Min frequency	Max frequency
"Sine"	0.001 Hz	100 MHz
"Square"	0.001 Hz	30 MHz
"Ramp"	0.001 Hz	1 MHz
"DC"	-	-
"Pulse"	0.001 Hz	30 MHz
"Cardinal sine"	0.001 Hz	5 MHz

"Function type"	Min frequency	Max frequency
"Cardiac"	0.001 Hz	1 MHz
"Gauss"	0.001 Hz	25 MHz
"Lorentz"	0.001 Hz	10 MHz
"Exp.rise"	0.001 Hz	10 MHz
"Ep. fall"	0.001 Hz	10 MHz

The values of the "Frequency" and "Period" depend on each other, as:

Period = 1/ Frequency

Remote command:
`WGENerator<m>:FREQuency` on page 1887

Period
Sets the period of the waveform. The available period range depends on the selected "Function Type".

Remote command:
`WGENerator<m>:PERiod` on page 1887

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses for what percentage of the period, the signal state is high.

Remote command:
`WGENerator<m>:FUNCtion:SQUare:DCYCle` on page 1888

Symmetry
Sets the symmetry of a ramp waveform, the percentage of time the waveform is rising. By changing the symmetry of the ramp, you can create, for example, triangular waveforms.

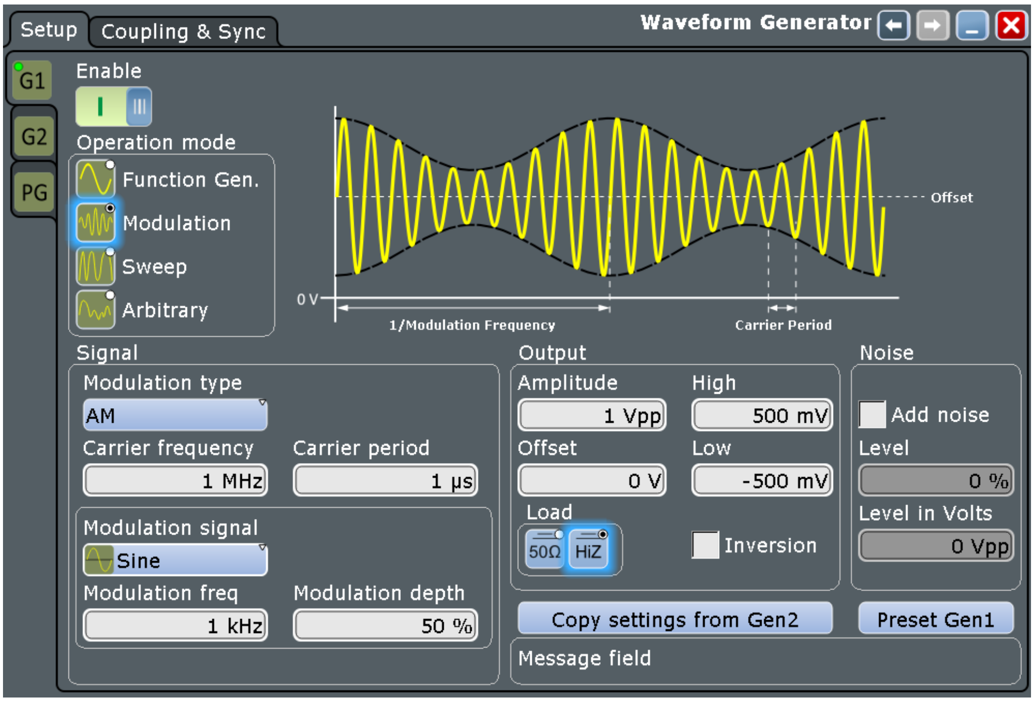
Remote command:
`WGENerator<m>:FUNCtion:RAMP[:SYMMetry]` on page 1888

Pulse width
Sets the pulse width, the pulse duration of the generated pulse waveform.

Remote command:
`WGENerator<m>:FUNCtion:PULSe[:WIDTh]` on page 1888

14.1.3 Modulation

Modulation is when properties of an original periodic waveform, the carrier signal, are varied according to a second modulating signal. The type of modulation used determines which properties are changed.



14.1.3.1 General Settings

Consists of settings for selecting the modulation type.

Modulation type

Selects the modulation type, which defines how the carrier signal is modified.

"AM" Amplitude modulation.
See: Chapter 14.1.3.2, "AM Modulation", on page 901.

"FM" Frequency modulation.
See: Chapter 14.1.3.4, "FM Modulation", on page 903.

"PWM" Pulse width modulation
See: Chapter 14.1.3.3, "PWM Modulation", on page 902

"FSK" Frequency shift keying (FSK) modulation.
See:Chapter 14.1.3.5, "FSK Modulation", on page 904.

Remote command:
`WGENerator<m>:MODulation:TYPE` on page 1889

Carrier frequency

Sets the frequency of the carrier signal.

Remote command:
`WGENerator<m>:MODulation:CARRier:FREQuency` on page 1890

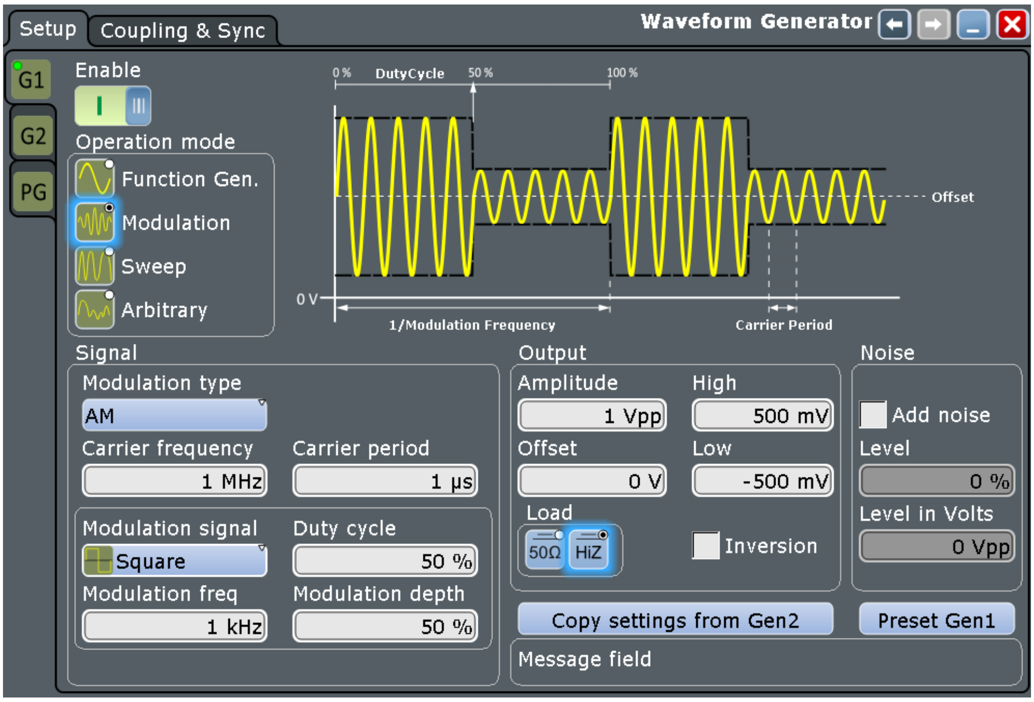
Carrier period

Sets the period of the carrier signal.

Remote command:
WGENerator<m>:MODulation:CARRier:PERiod on page 1890

14.1.3.2 AM Modulation

For amplitude modulation (AM), the amplitude of the carrier signal is varied according to the modulation signal.



Modulation signal
Selects the type of the modulation signal for the AM modulation types.

Remote command:
WGENerator<m>:MODulation:AM[:FUNCtion] on page 1891

Modulation freq
Sets the frequency of the modulation waveform.

Remote command:
WGENerator<m>:MODulation:AM:FREQuency on page 1890

Modulation depth
Sets the modulation depth, the percentage of the amplitude range that is used for the modulation.

Remote command:
WGENerator<m>:MODulation:AM:DEPTh on page 1889

Symmetry
Sets the symmetry for the ramp modulation waveform, the percentage of time that the waveform is rising.

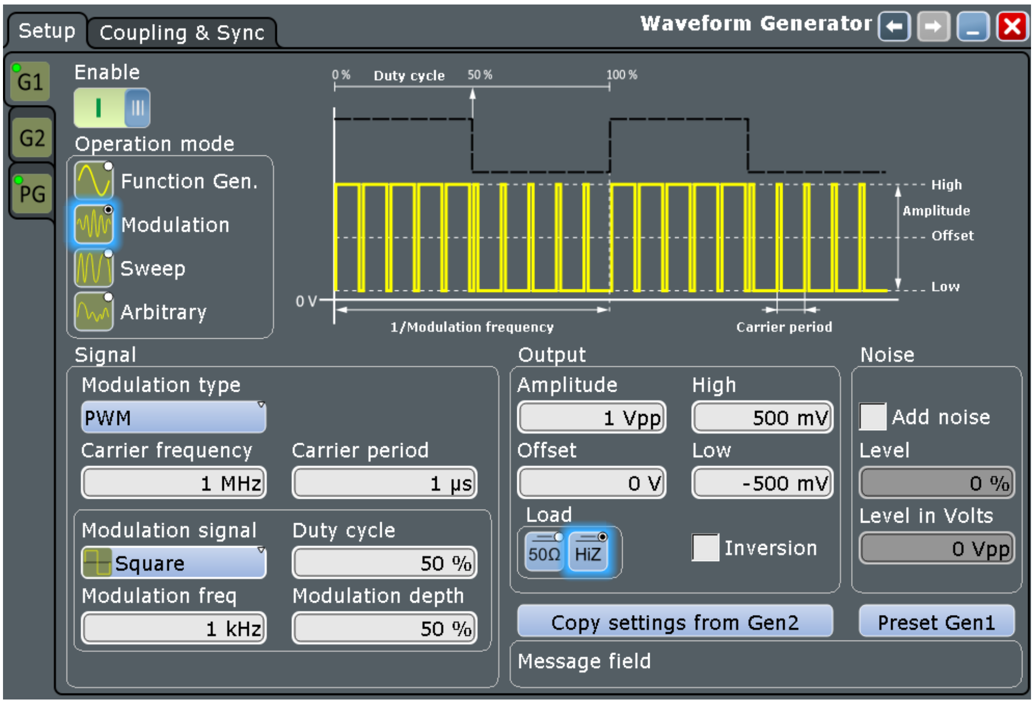
Remote command:
`WGENerator<m>:MODulation:AM:SYMMetry` on page 1890

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses for what percentage fraction of the period, the waveform is active, i.e. the signal state is high.

Remote command:
`WGENerator<m>:MODulation:AM:DCYCle` on page 1889

14.1.3.3 PWM Modulation

For pulse width modulation (PWM), the time for which the signal is in a high state is varied according to the modulation signal.



Modulation signal
Selects the type of the modulation signal for the PWM modulation types.

Remote command:
`WGENerator<m>:MODulation:PWM[:FUNCtion]` on page 1894

Modulation freq
Sets the frequency of the modulation waveform.

Remote command:
`WGENerator<m>:MODulation:PWM:FREQuency` on page 1894

Modulation depth
Sets the modulation depth, the percentage of the pulse width range that is used for the modulation.

Remote command:
WGENerator<m>:MODulation:PWM:DEPTh on page 1894

Symmetry
Sets the symmetry for the ramp modulation waveform, the percentage of time that the waveform is rising.

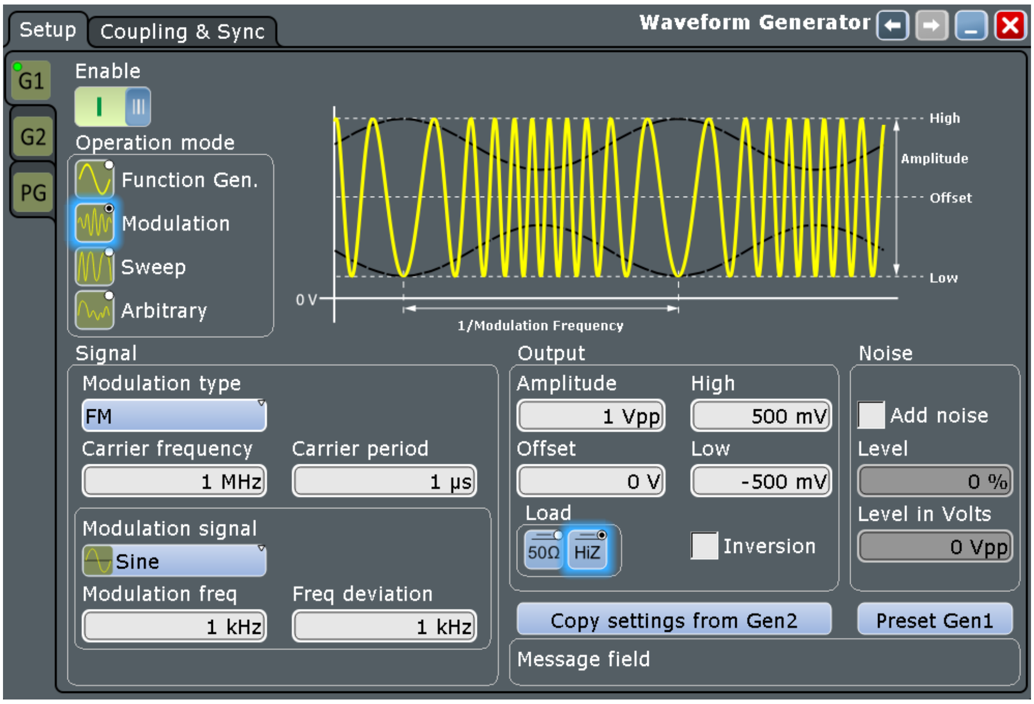
Remote command:
WGENerator<m>:MODulation:PWM:SYMMetry on page 1894

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses for what percentage fraction of the period, the waveform is active, i.e. the signal state is high.

Remote command:
WGENerator<m>:MODulation:PWM:DCYCle on page 1893

14.1.3.4 FM Modulation

For frequency modulation (FM), the frequency of the carrier signal is varied according to the modulation signal.



Modulation signal
Selects the type of the modulation signal for the FM modulation types.

Remote command:
WGENerator<m>:MODulation:FM[:FUNCtion] on page 1892

Modulation freq
Sets the frequency of the modulation waveform.

Remote command:
`WGENerator<m>:MODulation:FM:FREQuency` on page 1892

Symmetry
Sets the symmetry for the ramp modulation waveform, the percentage of time that the waveform is rising.

Remote command:
`WGENerator<m>:MODulation:FM:SYMMetry` on page 1892

Freq deviation
Sets the frequency deviation, the maximum difference between and FM modulated signal and the carrier signal.

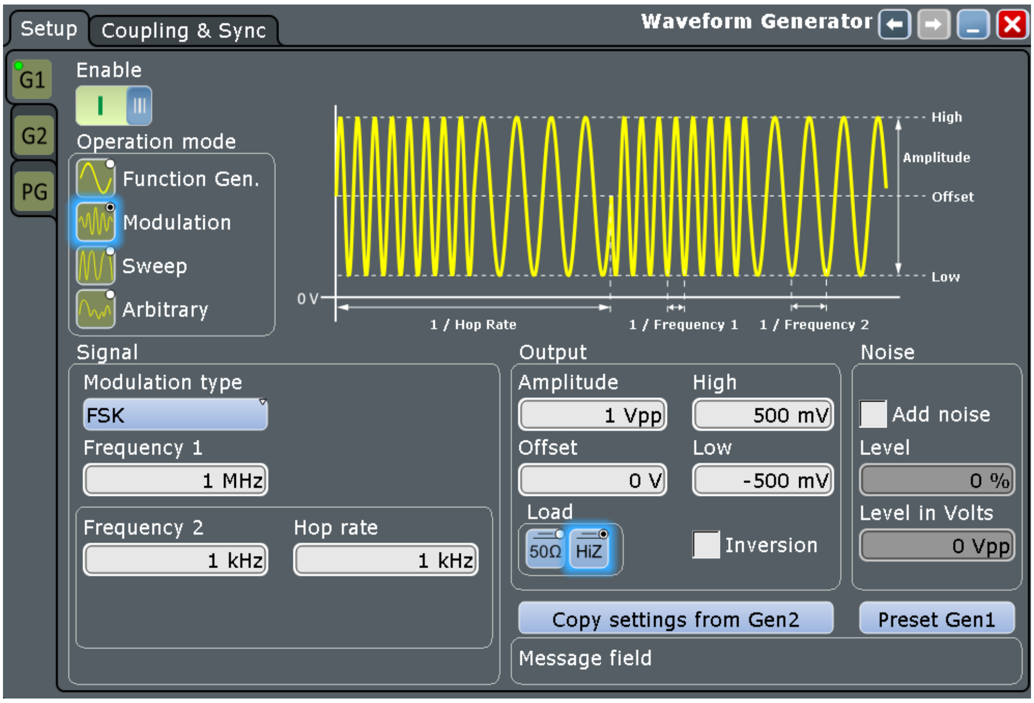
Remote command:
`WGENerator<m>:MODulation:FM:DEViation` on page 1891

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses the percentage of the period during which the waveform is active, i.e. the signal state is high.

Remote command:
`WGENerator<m>:MODulation:FM:DCYCle` on page 1891

14.1.3.5 FSK Modulation

For frequency shift keying (FSK) modulation, the signal switches between Frequency 1 and Frequency 2 at a Hop rate.



Frequency 1
Sets the frequency of the first signal in FSK modulated signal.

Remote command:
WGENerator<m>:MODulation:FSK:FONE on page 1892

Frequency 2
Sets the frequency of the second signal in FSK modulated signal.

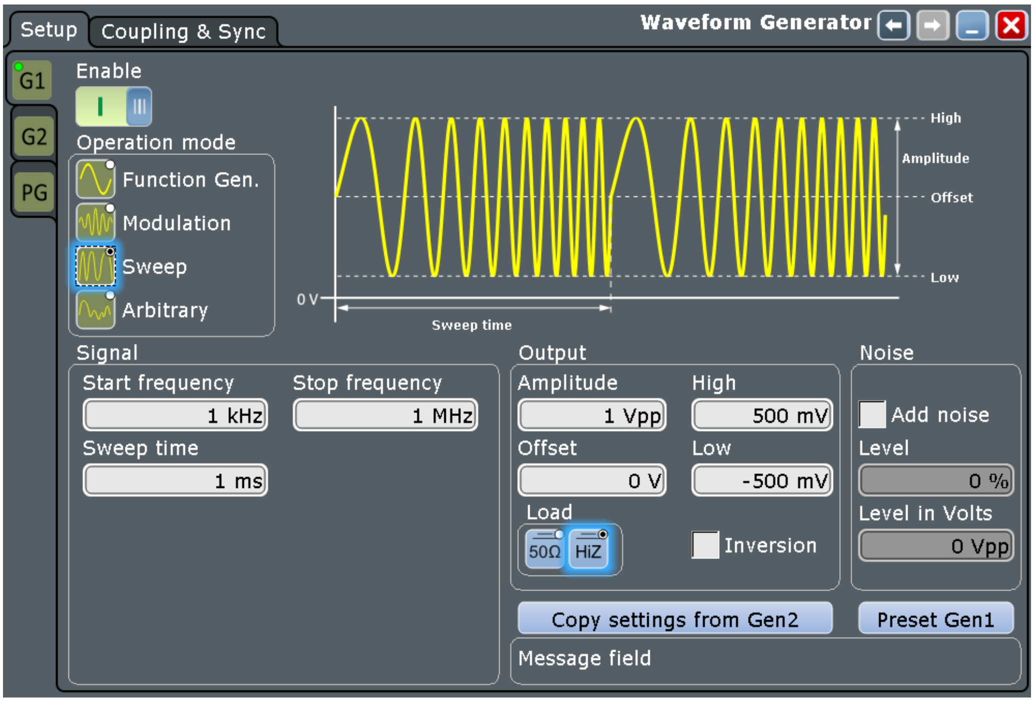
Remote command:
WGENerator<m>:MODulation:FSK:FTWO on page 1893

Hop rate
Sets the hop rate, the frequency at which signal switches between Frequency 1 and Frequency 2.

Remote command:
WGENerator<m>:MODulation:FSK[:RATE] on page 1893

14.1.4 Sweep

In the sweep mode, the R&S RTE generates a signal whose frequency gradually changes from the Start frequency to the Stop freuqency for a certain Sweep time.



Start frequency
Sets the start frequency of the sweep signal.

Remote command:
WGENerator<m>:SWEep:FSTart on page 1895

Stop freuqency
Sets the stop frequency of the sweep signal.

Remote command:
`WGENerator<m>:SWEep[:FEND]` on page 1895

Sweep time
Sets the duration of the sweep.

Remote command:
`WGENerator<m>:SWEep:TIME` on page 1895

14.1.5 Arbitrary

The arbitrary waveform generator allows you to output a user-defined waveform for testing your devices. You can output a waveform from a file or from the current R&S RTE reference curve format. Files in `*.csv` and `*.bin` formats are supported. This files must follow a defined structure. You can load `*.csv` files in an R&S Wave Gen format (see Content and format of the R&S wave gen *.csv files), Tektronix AFG format or Keysight WaveGen format.

Content and format of the R&S wave gen *.csv files

The R&S waveform generator format can contain the following values:

- Rate
- Time value
- Voltage value

If all the values are defined, the file format is as follows:

```
Rate = 5000000           //Sample rate of the arbitrary waveform.
0.000000E+000,-5.995    //Time value 1, Voltage value 1
1.237011E-005,-6.0      //Time value 2, Voltage value 2
.......
```

In this case, the rate is reflected in the "Sample Rate" field of the user interface. The total number of Time/Voltage values is reflected in the "Samples" of the user interface. Anything written after `//` is ignored as a comment.

You can define only some of the values. According to what you define, the file format looks different and is handled differently:

- With specified *Rate*:
  Time values are ignored. You can specify just rate and voltage values as below:

```
Rate = 5000000              //Sample rate of the arbitrary waveform
-5.995                      //Voltage value 1
-6.0                        //Voltage value 2
.......
```

- Without specified *Rate* and without specified *Time* values:

A sample rate of 1Mbps is used to calculate the waveform. You can change the "Sample rate" in the user interface. The voltage values are then played with this sample rate.

```
-5.995                //Voltage value 1
-6.0                   //Voltage value 2
.......
```

- Without specified *Rate* and wit specified *Time*:
  The timing information of the first 2 time values is used to calculate the sample rate.

Example:

Consider the following file:

```
0.000000E+000,-5.995    //Time value 1, Voltage value 1
1.237011E-005,-6.0      //Time value 2, Voltage value 2
```

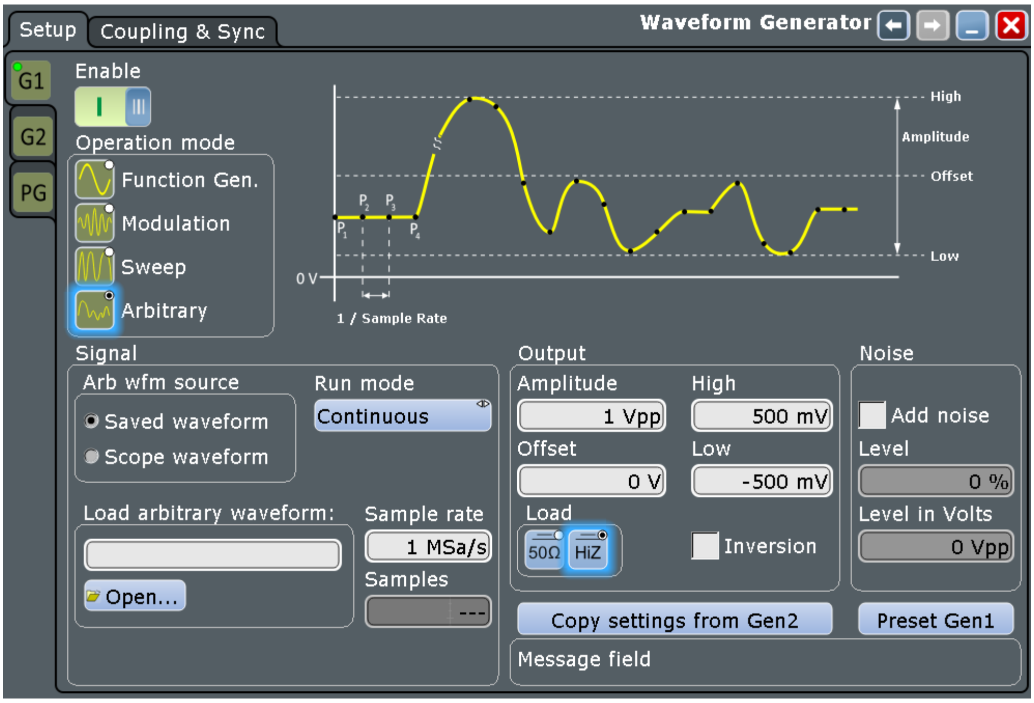
The sample rate is:

Sample rate = 1/ Time between first two samples =1 / 1.237011E-005 = 80.840KSampe/sec

Content and format of the R&S arbitrary generator *.bin files

The file stream should contain the following information in the given order:

- Sample rate [double format]
- Number of samples [double format]
- Samples [double format] * number of samples



14.1.5.1 General Settings

Arb wfm source

Selects the arbitrary waveform source. You can load an existing file or load the current oscilloscope waveform.

Remote command:

`WGENerator<m>:ARBGen[:SOURce]` on page 1898

Running mode

Selects the duration for which the signal of the arbitrary generator will be output after the trigger event. You can choose between a "Continuous" and "Single period" duration.

Remote command:

`WGENerator<m>:ARBGen:RUNMode` on page 1897

Sample rate

Sets the sample rate for the arbitrary waveform.

Remote command:

`WGENerator<m>:ARBGen:SRATe` on page 1898

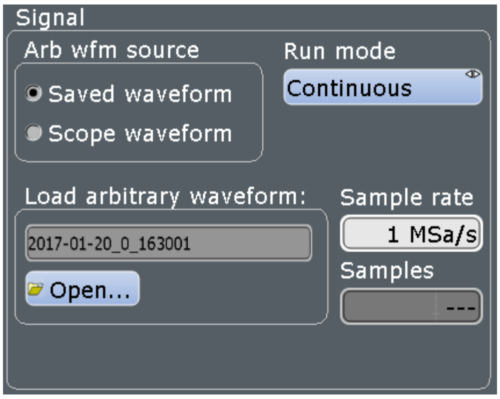
Samples

Displays the number of samples in the loaded waveform.

Remote command:

`WGENerator<m>:ARBGen:SAMPles?` on page 1898

14.1.5.2 Saved Waveform



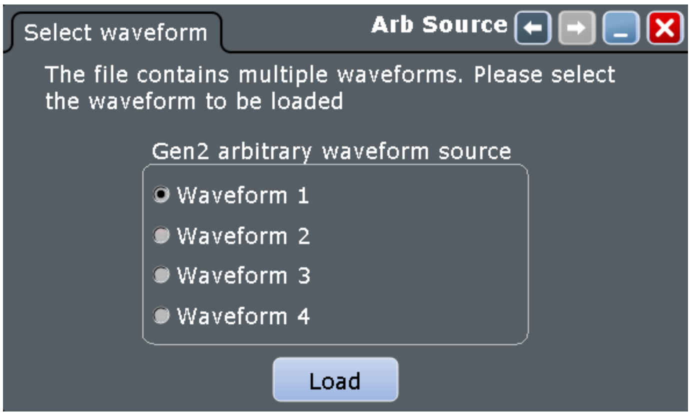
For a saved waveform, the following settings are available:

Load arbitrary waveform
Opens a file selection dialog box and loads the selected file. Supported are `.bin` and `.csv` extension files.

Remote command:
`WGENerator<m>:ARBGen:NAME` on page 1897
`WGENerator<m>:ARBGen:OPEN` on page 1897

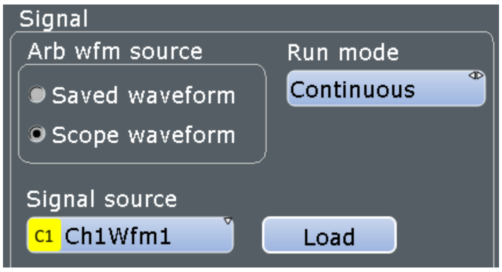
Select waveform
When a multichannel file is loaded into the arbitrary waveform generator, a dialog appears to select which waveform from the file is loaded.

Select the waveform and press "Load" to load it into the arbitrary waveform generator.



Remote command:
`WGENerator<m>:ARBGen:MULTichannel:NAME` on page 1896
`WGENerator<m>:ARBGen:MULTichannel:IMPort` on page 1896
`WGENerator<m>:ARBGen:MULTichannel:OPEN` on page 1897

14.1.5.3 Scope Waveform



For a scope waveform, the following settings are available:

Signal source
Selects the oscilloscope source, from which the arbitrary signal is loaded.

Remote command:
`WGENerator<m>:ARBGen:SELect` on page 1898

Load
Loads the waveform from the selected "Signal source".

Remote command:
`WGENerator<m>:ARBGen:COPY` on page 1896

14.1.6 Output

Amplitude
Sets the amplitude, peak to peak voltage, of the output waveform. It is defined as the voltage difference between the maximum ("High") and the minimum ("Low") voltage levels.

The "Amplitude" value is set for the currently selected "User Load". If the "User Load" is changed, the value of "Amplitude" is adapted to this new setting.

Remote command:
`WGENerator<m>:VOLTage[:VPP]` on page 1899

Offset
Sets a voltage offset

Remote command:
`WGENerator<m>:VOLTage:OFFSet` on page 1901

High
Sets the high signal level of the output waveform.

Remote command:
`WGENerator<m>:VOLTage:HIGH` on page 1900

Low
Sets the low signal level of the output waveform.

Remote command:
WGENerator<m>:VOLTage:LOW on page 1900

Inversion
Inverts the waveform at the offset level.

Remote command:
WGENerator<m>:VOLTage:INVersion on page 1900

User Load
Select the user load, the load of the DUT at its connection. You can select either a "50Ω" or a "HiZ" (high input impedance) load.

Remote command:
WGENerator<m>:OUTPut[:LOAD] on page 1899

DC Level
Sets the voltage DC level for the generated DC signal, for "Operation mode" >"Function Gen." and "Function type"> "DC".

Remote command:
WGENerator<m>:VOLTage:DCLevel on page 1900

14.1.7 Noise

Add Noise
Enables the adding of noise to the waveform.

Remote command:
WGENerator<m>:MODulation:NOISe on page 1902

Level
Sets the level of the noise in percentage of the set "Amplitude" output of the signal.

Remote command:
WGENerator<m>:MODulation:NLPCent on page 1901

Level in Volts
Displays the level of the noise in volts.

Remote command:
WGENerator<m>:MODulation:NLABsolute? on page 1901

Level
For "Function type">"DC" only.

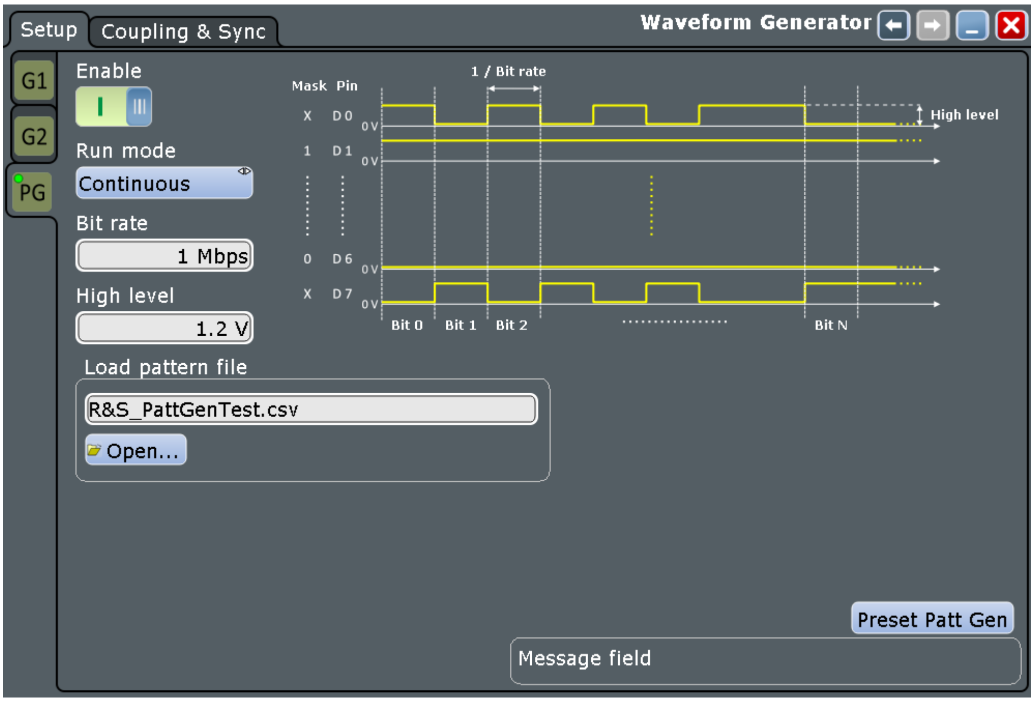
Sets the level for the DC signal.

Remote command:
WGENerator<m>:MODulation:NDCLevel on page 1901

14.2 Setup of the Pattern Generator

Access: [Gen 1]/[Gen 2] > "PG" tab

The pattern generator outputs parallel patterns.



NOTICE

Using pattern generator accessories

The pattern generator with connected Patt Gen Cable and Patt Gen Board (1329.7054.02) is considered as a test probe, EN 61326-2-1, clause 5.2.4.101, note 1. Therefore normal operation may display increased emissions above the limits as specified in EN 55011 and/or reduced interference resistance as required in EN 61326-1, table 1, basic requirements.

If the cable and the board are connected, other surrounding electronic devices may be disturbed. Furthermore, signals at the analog generator outputs Gen1 and Gen2 may be distorted by surrounding devices.

The settings of the pattern generator are not affected by an instrument preset. Press "Preset Patt Gen" to preset the settings of the pattern generator.

Content and format of the pattern generator files

The pattern generator supports `.bin` or `.csv` file formats.

Content and format of the R&S*.csv files

The file's header have the following structure:

- Bit rate [double]: the number of transmitted bits per second. The value is reflected in the user interface.
- High level [double]: the value is reflected in the user interface.
- Mask [string of 8 characters made up of "X"/"1"/"0"] : defines how the output of the pattern generator looks like. The following values are defined:
  - *1'* means that the pin output is always at high level
  - *0'* means that the pin output is always at low level (close to 0V)
  - *X* means that the pin output varies according to the given pattern
- Data sample format [HEX, BIN, OCT, DEC]: indicates how the samples are going to be interpreted. Each sample is represented as 8bit value (corresponding to the 8bit pattern generator) considering the selected format.

Example: Sample format HEX

```
Format= HEX        // Defines the format of the pattern values [HEX, DEC, BIN, OCT]
0F -> Data Sample 1
21 -> Data Sample 2
........
```

The samples are mapped on the 8 pins of the pattern generator as follows:

```
=> Pattern Samples are:
D7    D6    D5    D4   D3    D2   D1    D0
0     0     0     0    1     1    1     1
0     0     1     0    0     0    0     1
```

Example: .csv pattern generator file

```
R&S Pattern Generator File
Rate= 1000000        // Bit Rate [double]
HLevel= 1.5          // High Voltage Level [double]
Mask= X111000X       // Masks the Pins to be used in the Pattern Generator
                     // [0 => always LOW, 1 => always HIGH, X/x => used in the Pattern]
Format= DEC          // Defines the format of the pattern values [HEX, DEC, BIN, OCT]0
1
2
3
....
200
```

Content and format of the R&S pattern generator *.bin files

For the content of the fields, refer to "Content and format of the R&S*.csv files" on page 914.

The file stream should contain the following information in the given order:

- Bit rate [double]
- High level [double]
- Mask [string of 8 characters made up of "X"/"1"/"0"]
- Number of samples [UINT32]
- Data samples [UINT8] * number of samples

Enable
Enables the waveform generator/ pattern generator and outputs the signal to the connectors.

Remote command:
WGENerator<m>[:ENABle] on page 1886
PGENerator:ENABle on page 1902

Run mode
Selects the duration for which the signal of the generator will be output after the trigger event. You can choose between a "Continuous" and "Single period" duration.

Remote command:
PGENerator:RUNMode on page 1903

Bit rate
Sets the number of transmitted bits per second for the pattern generator.

Remote command:
PGENerator:BITRate on page 1902

High level
Sets the high level of the signal.

Remote command:
PGENerator:HLEVel on page 1903

Load pattern file
Opens a dialog for selecting an existing pattern file. It is possible to load `.bin` or `.csv` files, see "Content and format of the pattern generator files" on page 914.

Remote command:
PGENerator:FILE:OPEN on page 1902
PGENerator:FILE[:NAME] on page 1903

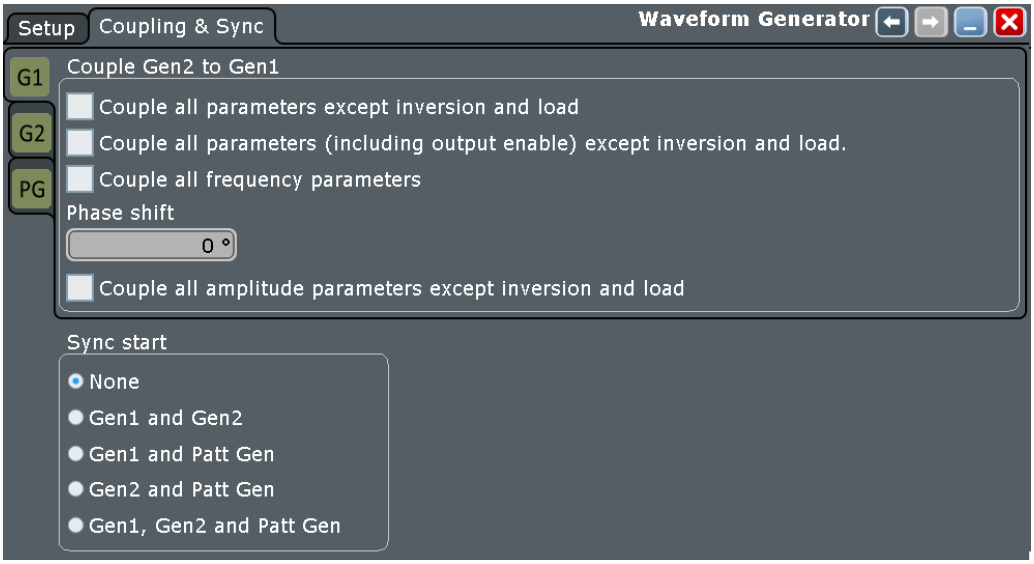
Preset Gen1/Gen2/Patt Gen
Sets the parameters of the generator to their default values. The settings of the generators are not affected by an instrument preset.They are also not stored in the user-defined preset.

Remote command:
WGENerator<m>:PRESet on page 1886
PGENerator:PRESet on page 1903

14.3 Coupling and Sync Settings

Access: [Gen 1]/[Gen 2] > "Coupling & Sync" tab.

In the R&S RTE, you can couple certain settings of the available waveform generators. If one of the available coupling options is enabled for one of the generators, then you cannot change the coupled parameters at the "Setup" tab of the other generator. The values for all coupled parameters are automatically taken from the currently selected generator.



Couple Gen 2 to Gen1/ Couple Gen 1 to Gen 2
Enables the coupling of the selected parameters of "Gen1" to "Gen2"/"Gen2" to "Gen1".

"Couple all parameters except inversion and load"
All signal, output and noise parameters of the generators are coupled, except for "Load" and "Inversion".

"Couple all parameters (including output enable) except inversion and load"
All signal, output and noise parameters of the generators are coupled, including the output enabling, but excluding "Load" and "Inversion". The setting is useful for the generation of differential signals

"Couple all frequency parameters"
All frequency parameters of the generators are coupled:

- For "Operation Mode" > "Function Gen.": "Frequency" and "Period"
- For "Operation Mode" > "Modulation": "Carrier frequency", "Carrier period", "Modulation freq", "Freq deviation", "Frequency 1", "Frequency 2" and "Hop rate"
- For "Operation Mode" > "Sweep": "Start frequency", "Stop frequency" and "Sweep time".
- For "Operation Mode" > "Arbitrary": "Arb wfm source" and "Signal source".

You can still change the other settings of the generators independently.

"Phase shift" Sets the phase shift between the waveform of "Gen1" and "Gen2" when the frequency parameters of the two waveforms are coupled.

"Couple all amplitude parameters except Inversion and Load"
All amplitude parameters of the generators are coupled:

- For all "Operation Mode": the output settings except of "Load" and "Inversion", "Amplitude", "High", "Offset" and "Low".
- For "Operation Mode" > "Function Gen.": "DC Level"
- For "Operation Mode" > "Modulation": "Modulation depth"

Remote command:
`WGENerator<m>:COUPling:ALL` on page 1903
`WGENerator<m>:COUPling:OUTPut` on page 1904
`WGENerator<m>:COUPling:AMPLitude` on page 1904
`WGENerator<m>:COUPling:PHASeshift` on page 1904
`WGENerator<m>:COUPling[:FREQuency]` on page 1904

Sync
Selects, which signals generated from the waveform generator are synchronized.

Selecting one of the sync options indicates that the first samples of those signals are generated at the same time, irrespective of if the generators are on or off. Selecting one of the coupling options automatically syncs the signals generated by the two waveform generators.

Remote command:
`GENerator:SYNC[:COMBination]` on page 1905

14.4 Configuring the Waveform Generator

This chapter explains step-by-step how to configure the waveform generator.

- Configuring a Function Waveform....918
- Configuring a Modulation Waveform....918
- Configuring a Sweep Waveform....919
- Configuring an Arbitrary Waveform....920
- Configuring a Pattern Generator Waveform....920

14.4.1 Configuring a Function Waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Function Gen." button.
4. Select the "Function type" that you want to generate, e.g. "Sine".
5. Depending on the selected "Function type", configure the settings of the waveform like "Frequency" and "Period".
6. If necessary, change the "Output" settings or add "Noise" to the waveform.
7. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

14.4.2 Configuring a Modulation Waveform

Generating an AM modulated waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "AM".
5. Set the "Carrier frequency" and the "Carrier period".
6. Tap "Modulation signal" and select the required waveform.
7. Depending on the selected "Modulation signal", configure the settings of the waveform like "Modulation freq" and "Modulation depth".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an PWM modulated waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "PWM".
5. Set the "Carrier frequency" and the "Carrier period".
6. Tap "Modulation signal" and select the required waveform.

7. Depending on the selected "Modulation signal", configure the settings of the waveform like "Modulation freq" and "Modulation depth".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an FM modulated waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "FM".
5. Set the "Carrier frequency" and the "Carrier period".
6. Tap "Modulation signal" and select the required waveform.
7. Depending on the selected "Modulation signal", configure the settings of the waveform like "Modulation freq" and "Freq deviation".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an FSK modulated waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "FSK".
5. Set the "Frequency 1", "Frequency 2" and the "Hop rate".
6. If necessary, change the "Output" settings or add "Noise" to the waveform.
7. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

14.4.3 Configuring a Sweep Waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Sweep" button.
4. Set the "Start frequency", the "Stop frequency" and the "Sweep time".

5. If necessary, change the "Output" settings or add "Noise" to the waveform.
6. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

14.4.4 Configuring an Arbitrary Waveform

Generating an arbitrary waveform from a saved file

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Arbitrary" button.
4. Set the "Arb wfm source" to "Saved waveform".
5. Select the "Run mode".
6. Press "Open" and set the path to your saved arbitrary waveform.
7. Set the "Sample rate".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an arbitrary waveform from the scope waveform

1. Press the [Gen 1] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Arbitrary" button.
4. Set the "Arb wfm source" to "Scope waveform".
5. Select the "Run mode".
6. Press the "Signal source" button and select the channel source for the waveform.
7. Set the "Sample rate".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

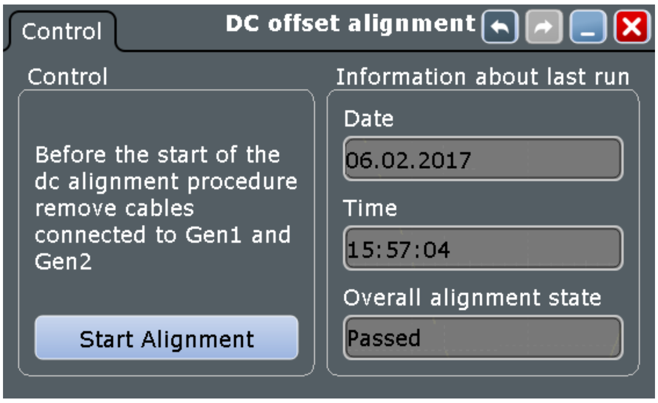
14.4.5 Configuring a Pattern Generator Waveform

1. Press the [Gen 1] key on the front panel.
2. At the left-hand side, select the vertical tab "PG".

3. Select the "Setup" tab.
4. Select the "Run mode".
5. Press "Open" and set the path to your saved pattern file.
6. Set the "Bit rate" and "High level".
7. Press "Open" and set the path to your saved arbitrary waveform.
8. Press the "Enable" button, to output the pattern at the output connector of the pattern generator.

14.5 DC Offset Alignment

Access: "Wave Gen" > "DC offset Allignment" tab



Start Alignement
Starts the alignment of the DC offset.

Remote command:
`GENerator:ALIGnment:DC[:STARt]` on page 1906

Date
Displays the date of the last performed DC offset alignment.

Remote command:
`GENerator:ALIGnment:DC:RESult:DATE?` on page 1905

Time
Displays the time of the last performed DC offset alignment.

Remote command:
`GENerator:ALIGnment:DC:RESult:TIME?` on page 1905

Overall alignment state
Displays the result of the DC offset alignment.

Remote command:
`GENerator:ALIGnment:DC:RESult[:STATe]?` on page 1906

15 Power Analysis (Option R&S RTE-K31)

With the R&S RTE and option R&S RTE-K31, you can perform power analysis measurements.

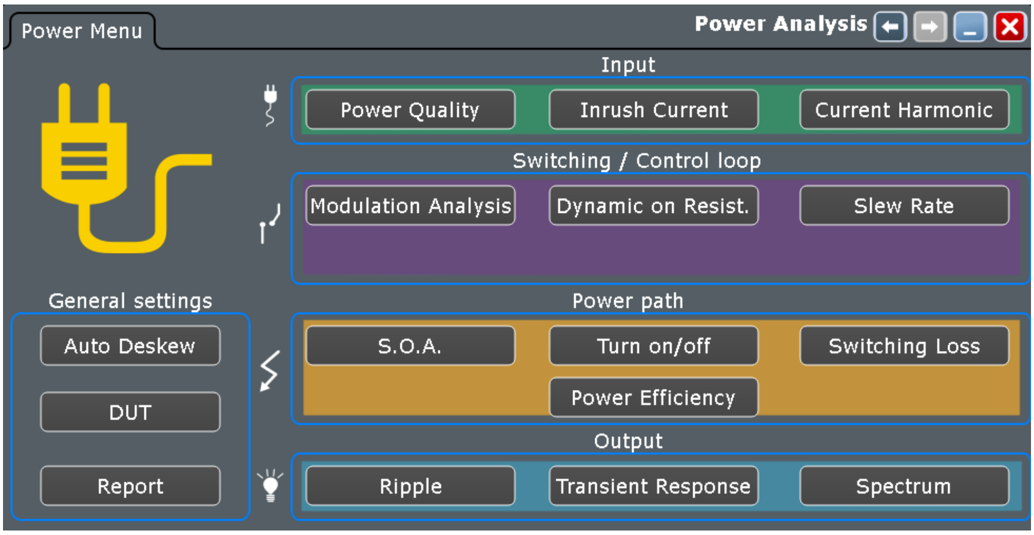
The following power measurements are available:

- Power Quality
- Inrush Current
- Current Harmonic
- Modulation Analysis
- Dynamic on Resistance
- Slew Rate
- Safe Operating Area (S.O.A.)
- Turn On/Off
- Switching Loss
- Power Efficiency
- Output Ripple
- Transient Response
- Output Spectrum

15.1 Power Measurement Selection

Access: "Analysis"> "Power"

The "Power Menu" is the entry point to all power measurements and the general setting required for them.



The tab has several areas:

- "General Settings": general settings, that can be used by all measurements, like deskewing.
- "Input": measurements for performing input line analysis. They are used to measure the characteristics of the input power and the effects the power supply exudes to the input line.
- "Switching and Control Loop": measurements for characterizing the switching properties of a device.
- "Power Path": measurements for analyzing the behavior of the devices that control the power flow through the switched-mode power supply (SMPS) circuit, including switching devices and inductors.
- "Output": measurements for characterizing the behavior and quality of the SMPS output voltage.

15.1.1 General Settings

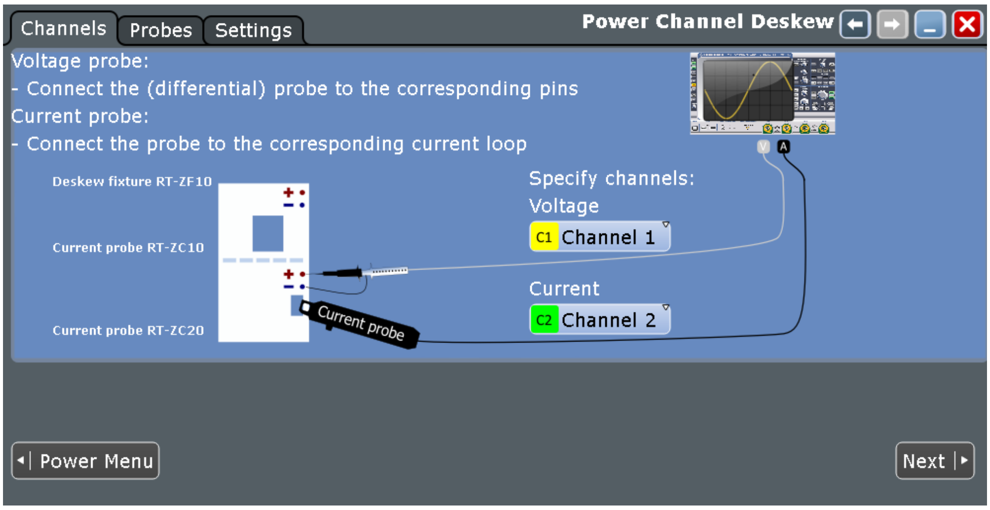
15.1.1.1 Auto Deskew

The "Auto Deskew" dialog box guides you through the auto deskew of your current and voltage probes.

Required equipment:

- R&S RT-ZF20 power deskew fixture
- Rohde & Schwarz voltage probe
- Rohde & Schwarz current probe

1. Select "Analysis" > "Power".
2. Under "General", select "Auto Deskew".
3. Connect the voltage probe and the current probe to the oscilloscope.
4. Connect the probes to the R&S RT-ZF20 power deskew fixture as shown in the "Channels" tab:



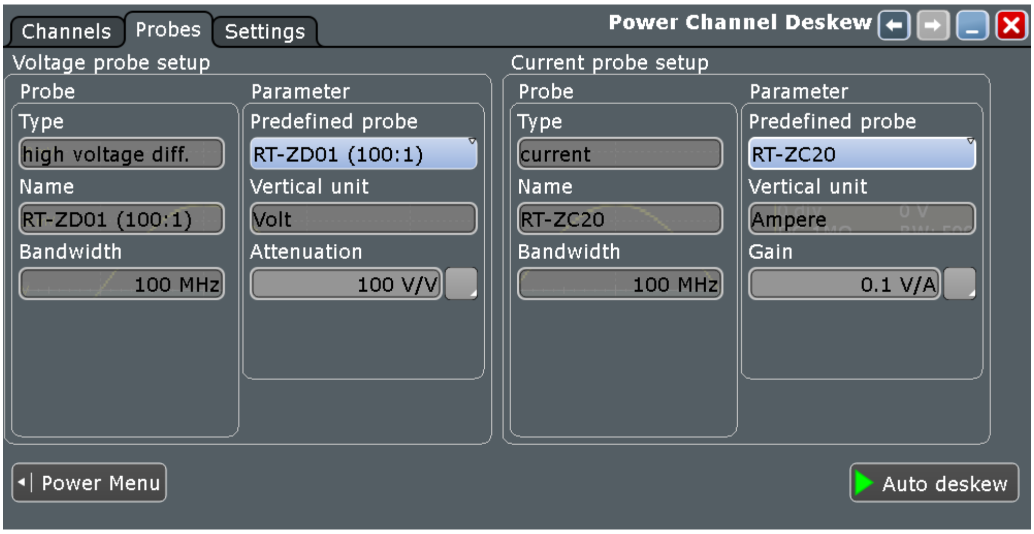
5. Select the correct channels for the "Current Source" and the "Voltage Source".
6. Tap "Next".
7. Check and complete the probe setup in the "Probes" tab.
   Current probes and high-voltage differential probes are not automatically detected by the instrument. Tap "Predefined probe" and select the correct probe type.
8. Tap the "Settings" tab.
9. Set the "Overwrite present skew setup" on page 926 and "Activate user defined preset" on page 926 options. These settings define whether the instrument uses the deskew result for user-defined preset and general skew settings.
10. Tap "Auto deskew".

    The probes are deskewed and the measurement can be started.

If no deskew fixture is available, you have to deskew your probes manually, see Chapter 4.9.1, "Skew", on page 194.

Probes

In the "Probes" tab, you check and set up your voltage and current probes.



Type, Name, Bandwidth

The fields show the characteristics of a recognized or predefined probe for information. If the instrument cannot recognize the probe, and the probe is not known, the "Type" is "None", and the other fields are empty.

Remote command:
`PROBe<m>:SETup:TYPE?` on page 1087
`PROBe<m>:SETup:NAME?` on page 1087
`PROBe<m>:SETup:BANDwidth?` on page 1087
`TRPRobe:SETup:TYPE?` on page 1087 (external trigger input)
`TRPRobe:SETup:NAME?` on page 1087 (external trigger input)
`TRPRobe:SETup:BANDwidth?` on page 1087 (external trigger input)

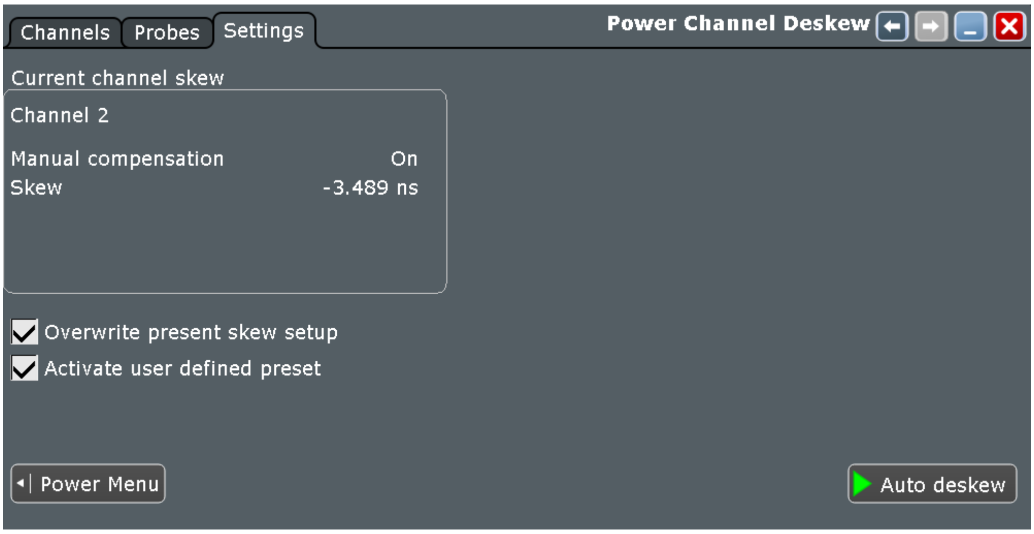
Predefined probe, Vertical unit, Attenuation

Current probes R&S RT-ZCxx, the high-voltage active probe R&S RT-ZD01 and the transmission line probe R&S RT-ZZ80 are not recognized automatically but the parameters of these probes are known to the instrument. Select the correct probe type and enter additional parameters if necessary. The corresponding "Vertical unit" and the "Attenuation" or "Gain" are set.

For an auto deskew, only Rohde & Schwarz probes are supported.

Settings

In this tab, you can define the preset behavior and how the auto deskew results are stored.



Current channel skew
Shows the skew settings of the channel connected to the current probe. Skew settings are defined in the "Horizontal > Skew" dialog box.

See also: Chapter 4.9.1, "Skew", on page 194

Overwrite present skew setup
If disabled, the instrument only stores the result of the auto deskew procedure as a separate value and does not use it. This value can be used later for power measurements. The general skew offset under "Current channel skew" remains unchanged.

If enabled, the result of the auto deskew procedure is used for all measurements on the selected channel. It is shown under "Current channel skew".

Remote command:
`POWer:DESKew:RESet` on page 1908

Auto deskew result
Available only if "Overwrite present skew setup" is disabled.

"Deskew value" Result of the auto deskew.

Remote command:
`POWer:DESKew:TIME?` on page 1908

"Apply" Writes the result of the auto deskew to the "Skew offset" of the selected channel.

Remote command:
`POWer:DESKew:CURRent` on page 1907

Activate user defined preset
If enabled, the deskew values are written to a user-defined preset file, and the user-defined preset is enabled. Thus, the probe setup and deskew values are not influenced by a manual [PRESET].

See also: Chapter 11.6, "Preset Setup", on page 469.

Remote command:
POWer:DESKew:UDPReset on page 1908

Auto Deskew
Starts an auto deskew.

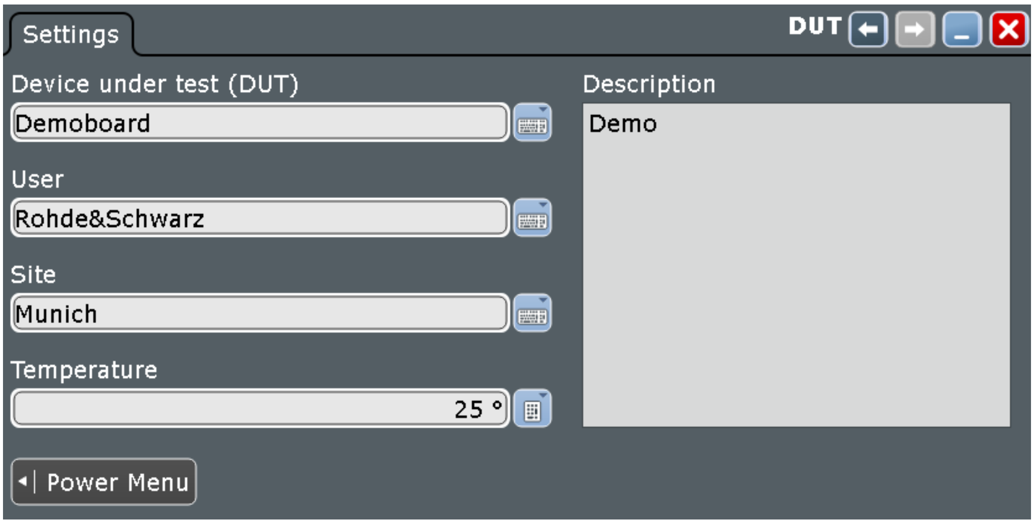
Make sure that the probes are configured correctly before you start the deskewing.

Remote command:
POWer:DESKew:EXECute on page 1908

15.1.1.2 DUT

Access: "Analysis" > "Power" > "DUT".

In this dialog, you can describe your device under test (DUT). The information set in this dialog can be used on the title page for a report generated from the "Power Analysis" measurements, see "Content" on page 931.



Device under test (DUT)
Enter a name for your DUT.

Remote command:
POWer:REPort:DUT on page 1910

User
Enter a user.

Remote command:
POWer:REPort:USER on page 1910

Site
Enter a site.

Remote command:
POWer:REPort:SITE on page 1910

Temperature
Enter the temperature.

Remote command:
`POWer:REPort:TEMPerature` on page 1910

Description
Enter a description.

Remote command:
`POWer:REPort:DESCription` on page 1910

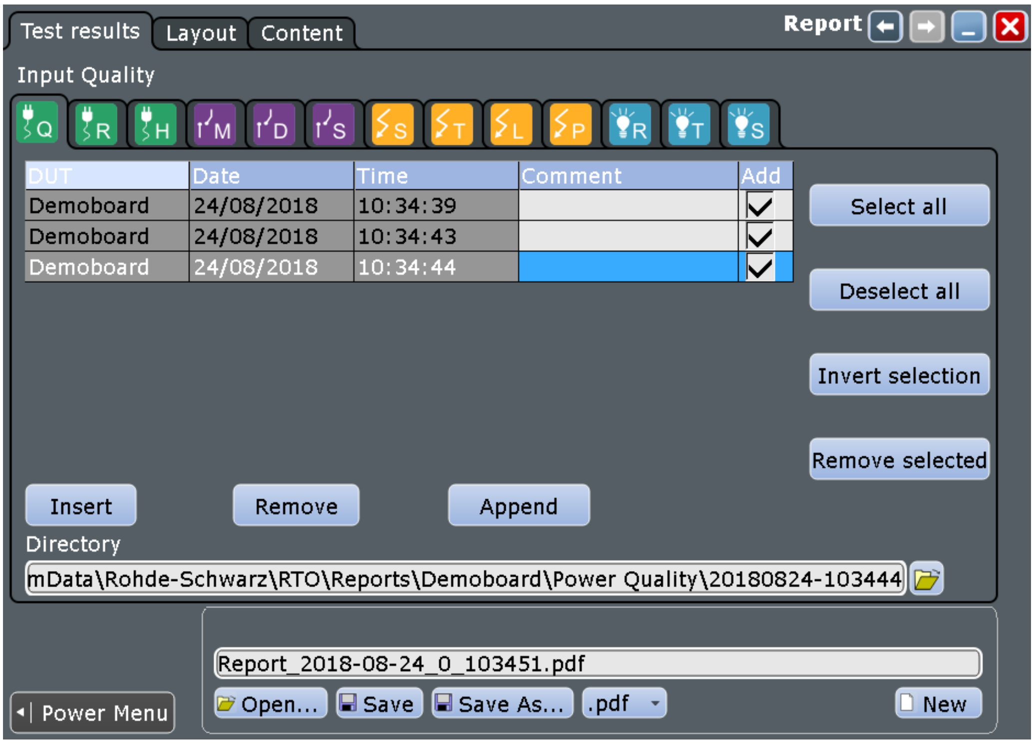
15.1.1.3 Report

Access "Analysis">"Power" > "Report".

Test Results

After executing a measurement, you can press "Add to report" and save the results. In this tab, you can manage all saved measurement results.

At the top of the tab, you can switch through the different "Power Analysis" measurements.



Report Table
Shows a list of the available measurements.

After you select a Directory, you can manage previous report results from this directory. To add a measurement report press "Insert" or "Append". To remove a measurement report press "Remove".

"DUT"	Shows the name of the DUT, see Chapter 15.1.1.2, "DUT", on page 927.
"Date"	Shows the date of the measurement.
"Time"	Shows at what time, the measurement result was added to report.
"Comment"	Enters a comment.
"Add"	Adds the selected measurement to the report.

Remote command:
`POWer:REPort:TEST:ADD` on page 1911
`POWer:REPort:TEST:COMMent` on page 1913
`POWer:REPort:TEST:COUNt` on page 1913
`POWer:REPort:TEST:INSert` on page 1911
`POWer:REPort:TEST:LSENd?` on page 1914
`POWer:REPort:TEST:REMove` on page 1911

Selection
Manages the selection of the result reports.

Select all ← Selection
Selects all result reports.

Remote command:
`POWer:REPort:TEST:SEA` on page 1912

Deselect all ← Selection
Deselects all result reports.

Remote command:
`POWer:REPort:TEST:DSEA` on page 1912

Invert Selection ← Selection
Inverts the selection of all result reports, meaning that all selected result reports are deselected and vice versa.

Remote command:
`POWer:REPort:TEST:ISE` on page 1912

Remove selected ← Selection
Removes the selected result report.

Remote command:
`POWer:REPort:TEST:RSE` on page 1912

Directory
Selects the directory, from which previous report results are inserted into the report table. You can use this directory to insert previously recorded report data into the current report.

Remote command:
`POWer:REPort:TEST:DIRectory` on page 1913

Report Path

Enter the file name to load or to save the report to, and select the file format with the format button on the right.

"Load" Loads the most recently created report with the Windows default viewer application for the pdf/rtf file format.

"Open" Opens a file selection dialog box and loads the selected file.

"Save" Saves the data to the selected file.

"Save As..." Opens the file selection dialog box and saves the data to the selected file.

".pdf/.rtf" Selects the file format.

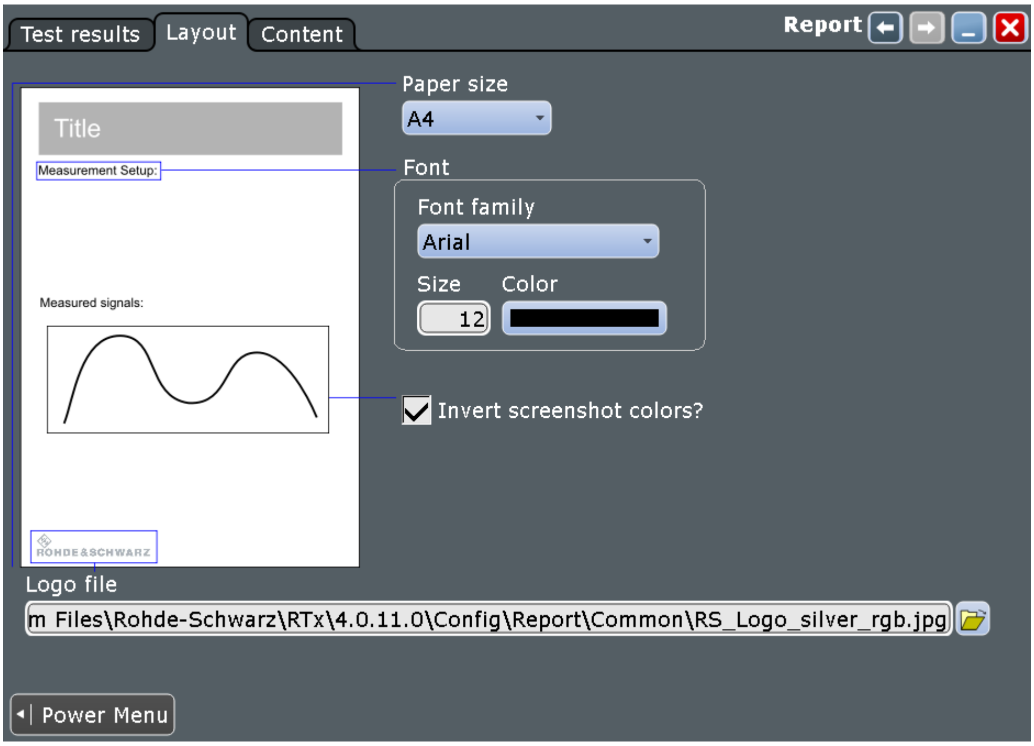
"New" Creates a new file.

"Delete" Deletes the selected file.

Remote command:
`POWer:REPort:FILE:DELete` on page 1911
`POWer:REPort:FILE:NAME` on page 1911
`POWer:REPort:FILE:NEW` on page 1911
`POWer:REPort:FILE:SAVE` on page 1911

Layout

In this tab, you can set up a layout for your report.



Paper size

Selects the paper size.

"A4" Selects A4.

"US Letter" Selects US Letter.

Remote command:
POWer:REPort:PAPersize on page 1910

Font
Sets the font for the report

Font Family ← Font
Selects the font family.

"Arial" Selects the font Arial.

"Helvetica" Selects the font Helvetica.

Remote command:
POWer:REPort:FONT:FAMI on page 1910

Size ← Font
Sets the font size.

Remote command:
POWer:REPort:FONT:SIZE on page 1910

Color ← Font
Sets the font color.

Remote command:
POWer:REPort:FONT:COLO on page 1910

Invert Screenshot Colors
Inverts the screenshot colors.

Remote command:
POWer:REPort:INVert on page 1912

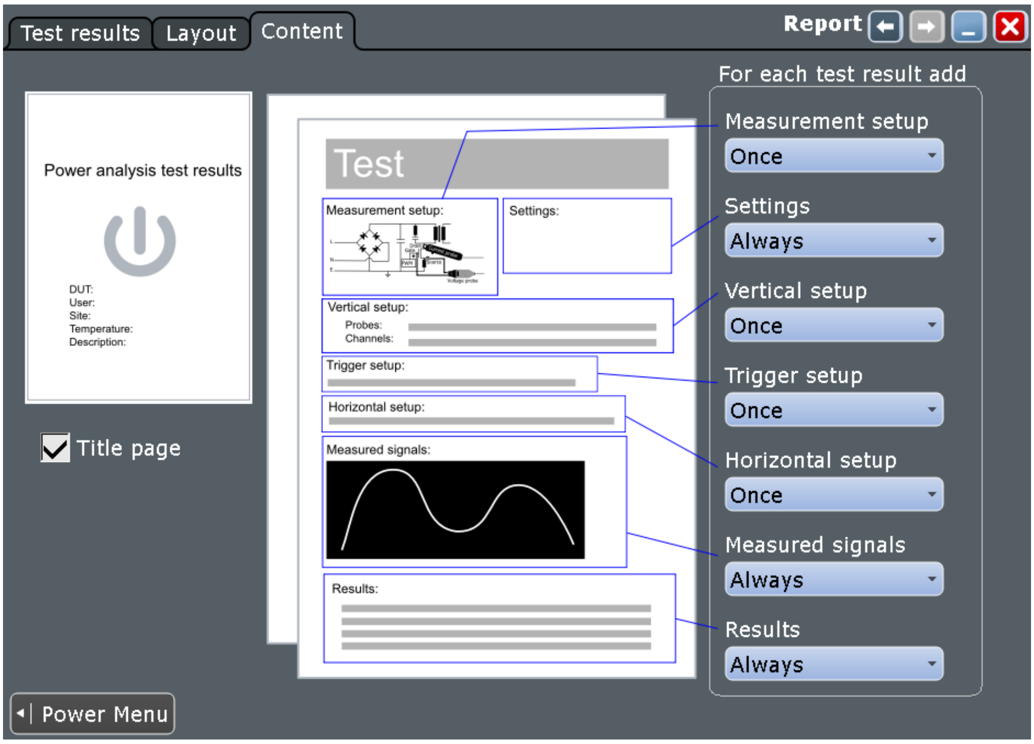
Logo File
Selects a path to a logo picture file.

Remote command:
POWer:REPort:LOGO on page 1910

Content

In this tab, you can select the contents of your report. For each content you can select how often it is included in the report:

- "Always": Shows the respective contents for each measurement.
- "Never": Does not show the respective contents in the report.
- "Once": Shows the respective contents once at the beginning of the report.



Title Page

Adds a Title page to the report. The contents can be set up in the "DUT" dialog, see Chapter 15.1.1.2, "DUT", on page 927.

Remote command:

`POWer:REPort:CONTent:TITLe` on page 1909

Measurement Setup

Adds a graphic of the measurement setup.

Remote command:

`POWer:REPort:CONTent:MSETup` on page 1909

Settings

Adds the settings of the analysis.

Remote command:

`POWer:REPort:CONTent:SETTings` on page 1909

Vertical Setup

Adds the vertical setup settings.

Remote command:

`POWer:REPort:CONTent:VSETup` on page 1909

Trigger Setup

Adds the trigger setup settings.

Remote command:

`POWer:REPort:CONTent:TSETup` on page 1909

Horizontal Setup
Adds the horizontal setup settings.

Remote command:
POWer:REPort:CONTent:HSETup on page 1909

Measured signals
Adds a diagram of the measured signal.

Remote command:
POWer:REPort:CONTent:MSIGnal on page 1909

Results
Adds the result box.

Remote command:
POWer:REPort:CONTent:RESU on page 1909

15.2 Overview of Power Measurement Setup

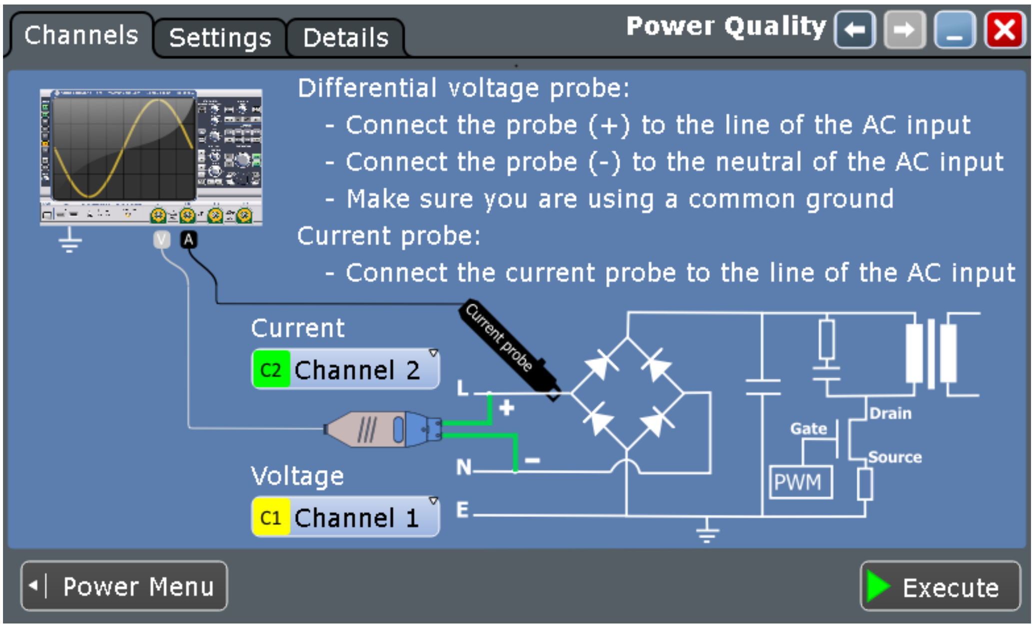
Each power analysis measurement dialog box consists of the following tabs:

- Channels
- Settings
- Details

At the bottom of each tab, you can find two buttons. Tapping "Power Menu", you can return to the power analysis measurement selection. "Execute" starts the power measurement.

15.2.1 Channels Tab

In the "Channels" tab, you find information on the experimental setup of the selected power measurement. A short description explains what probes are needed and how to connect them. The description is supported by a block diagram of the experimental setup that shows the connection points for the probes.



Depending on the selected power measurement, one or two voltage sources and current sources are required.

Current Source
Sets the channel for the current source.

Remote command:
POWer:SOURce:CURRent<1..2> on page 1907

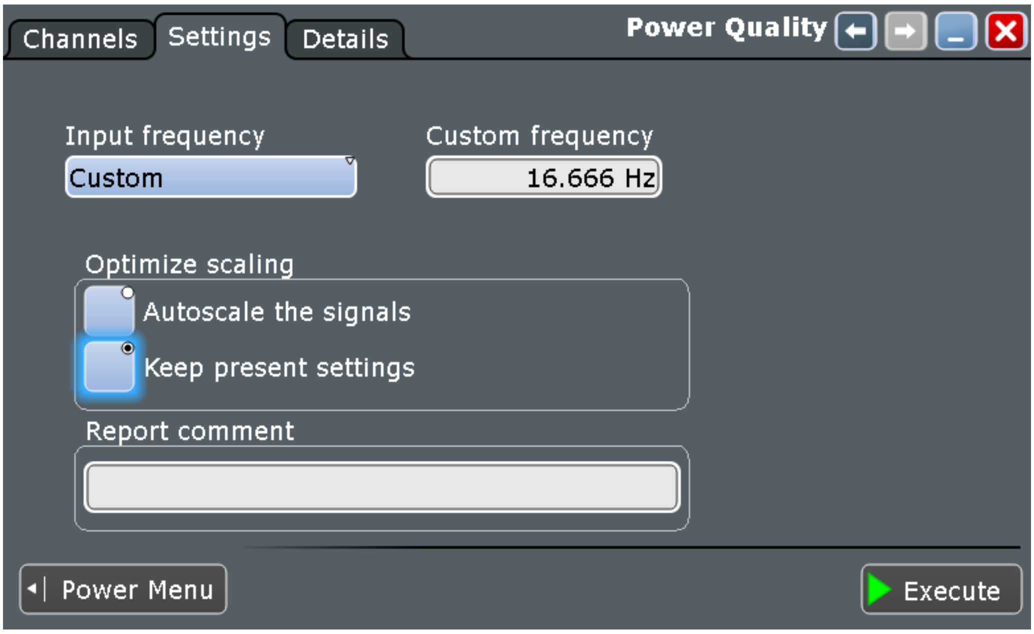
Voltage Source
Sets the channel for the voltage source input.

Remote command:
POWer:SOURce:VOLTage<1..4> on page 1907

15.2.2 Settings Tab

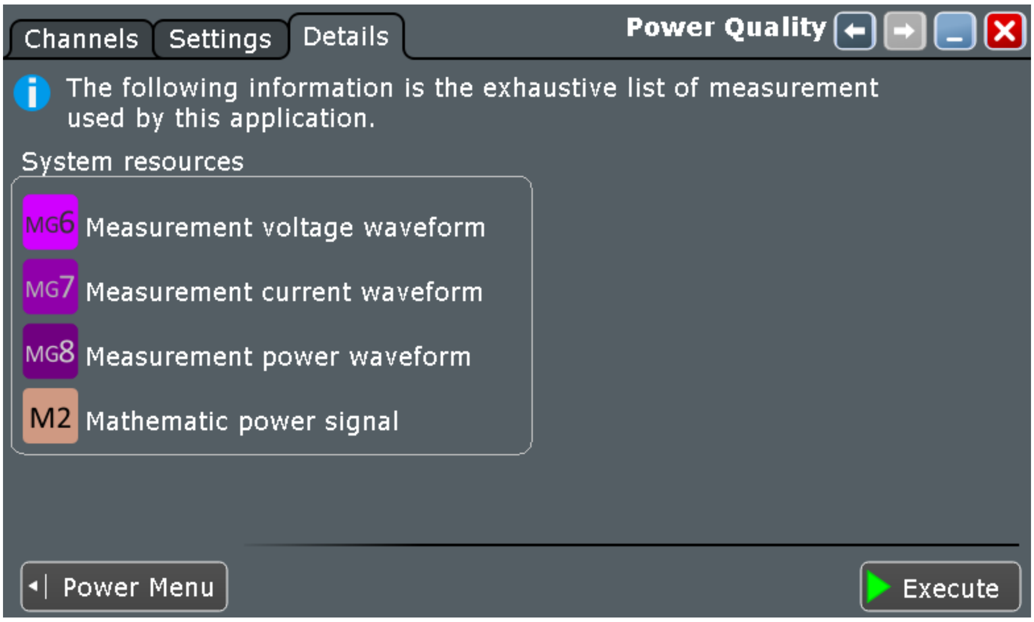
In the "Settings" tab, you configure the measurement and display settings. The settings depend on the selected power measurement.

For detailed information, see the "Settings" chapter of the relevant power measurement description.



15.2.3 Details Tab

In the "Details" tab, you find information on the measurement resources used by the selected power measurement.



The following resources may be used:

- Measurements
- Mathematic waveforms
- Cursors
- XY-diagrams

The instrument enables the required resources when the power measurement is started.

15.3 Power Quality

In an electric circuit power is a measure for the rate of flow of energy at a certain point of the circuit. The real power of a circuit, or the energy that can be used for work, is the portion of energy that is transferred in one direction over a complete cycle of the AC waveform. In AC circuits, however, inductive and capacitive elements can store energy temporarily. This portion of the power flow known as reactive power is then returned to the source without doing any work.

The "Power Quality" analysis measurements include the real power, the reactive power, the apparent power and the power factor. The crest factors and the phase angle between the current and voltage are also measured. These properties describe the power transfer in the system and allow you to characterize the power quality of the system.

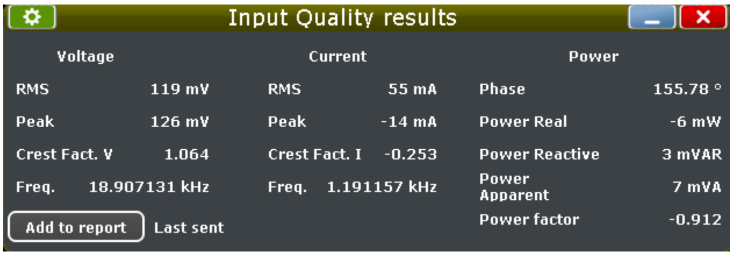
Required probes:

- Differential voltage probe
- Current probe

15.3.1 Power Quality Results

The results of "Power Quality" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the voltage waveform
  – the current waveform
  – the power waveform that is the product of the current and voltage waveforms
- The result box displays the numeric measurement results.



To measure and display the power quality, the instrument uses the following measurements and waveforms:

- "P6" "Meas 6" to measure the voltage
- "P7" "Meas 7" to measure the current
- "P8" "Meas 8" to measure the power
- "M2" "Math 2" to calculate the power

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

Voltage and current results

The voltage and current results are defined as follows:

Result	Description
RMS	Square root of the mean of the square of the current or voltage averaged over N cycles
Peak	Highest measured magnitude value of the voltage or current
Crest factor	Peak value / RMS value
Frequency	Frequency of the signal

Power results

The power in a system is described by several physical quantities: real power, reactive power, complex power, and phase angle. In Figure 15-1, you can see how these quantities are related if the voltage and the current are sinusoidal signals.

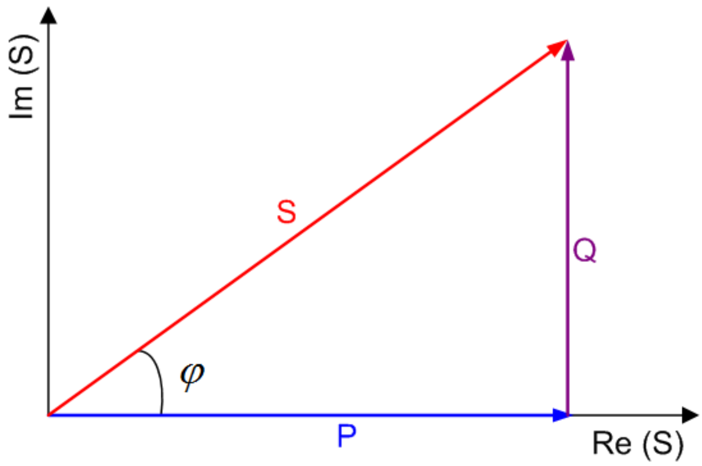


Figure 15-1: Power diagram for sinusoidal signals

P = real power [W]
Q = reactive power [VAR]
S = complex power [VA]
φ = phase angle between the current and the voltage sine waves [°]

The power results are defined as follows:

Result	Unit	Formula	Description
Power factor, P_{Factor}	-	$P_{Factor} = \|P\| / \|S\|$	Measure of the system efficiency. The value varies between -1 and 1.

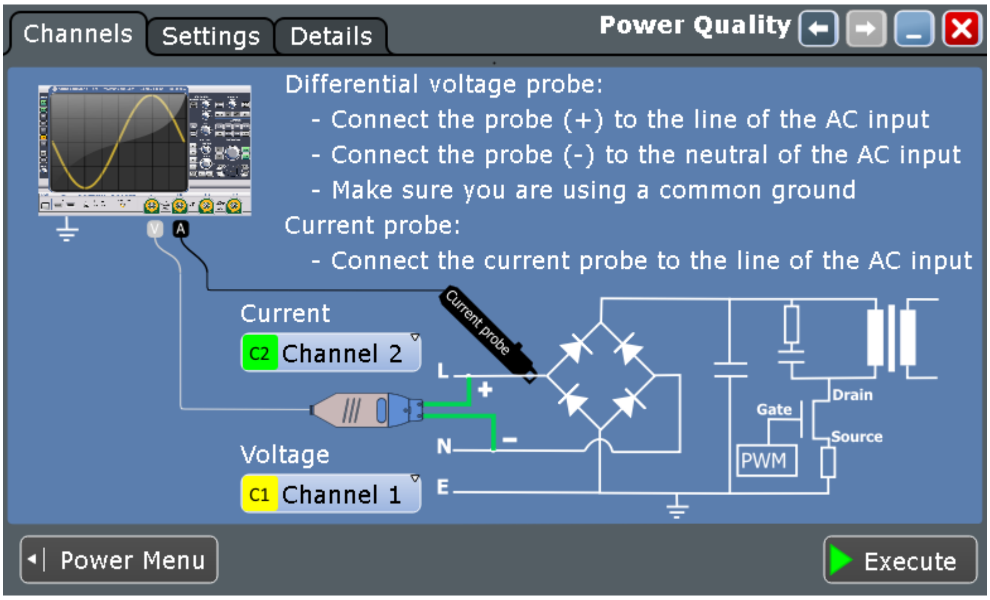
Phase, φ	°	$\varphi = acos\ (P_{Factor})$	Phase angle between the current and the voltage sine waves.
Real power, P	W	$P = V_{INSTANTENEOUS} \bullet I_{INSTANTENEOUS}$ (averaged over N cycles)	Energy of the system that can be used to do work.
Reactive power, Q	VAR (Volt-Ampere reactive)	$Q = \lvert S \rvert\ sin\varphi$	Power flow that is temporally stored in a system because of the inductive and capacitive elements.
Apparent power, |S|	VA	$\lvert S \rvert = V_{RMS} \bullet I_{RMS}$ (averaged over N cycles)	|S| is the magnitude of the vector sum of real and reactive power (the complex power S).

The following remote commands are used for handling the measurement results:
- POWer:QUALity:RESult:CURRent:CREStfactor? on page 1915
- POWer:QUALity:RESult:CURRent:FREQuency? on page 1915
- POWer:QUALity:RESult:CURRent:PEAK? on page 1915
- POWer:QUALity:RESult:CURRent:RMS? on page 1915
- POWer:QUALity:RESult:POWer:APParent? on page 1915
- POWer:QUALity:RESult:POWer:PFACtor? on page 1915
- POWer:QUALity:RESult:POWer:PHASe? on page 1915
- POWer:QUALity:RESult:POWer:REACtive? on page 1915
- POWer:QUALity:RESult:POWer:REALpower? on page 1915
- POWer:QUALity:RESult:VOLTage:CREStfactor? on page 1915
- POWer:QUALity:RESult:VOLTage:FREQuency? on page 1915
- POWer:QUALity:RESult:VOLTage:PEAK? on page 1915
- POWer:QUALity:RESult:VOLTage:RMS? on page 1915
- POWer:QUALity:REPort:ADD on page 1915

15.3.2 Configuring Power Quality

For details of the configuration settings, see Chapter 15.3.3, "Power Quality Settings", on page 939.

1. Select "Analysis" > "Power".
2. Under "Input", select "Power Quality".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



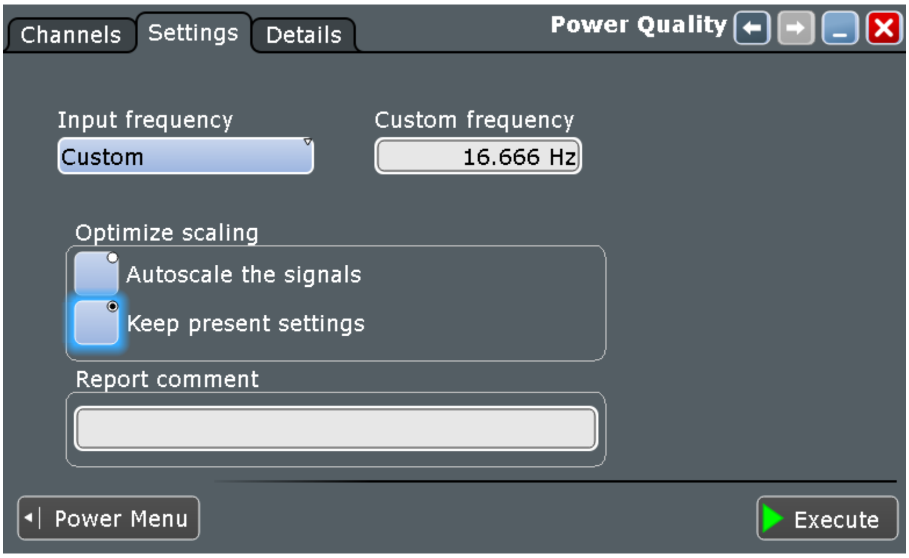
6. Select the correct channels of the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Input frequency" according to your signal.
9. Select an "Optimize Scaling" option.
10. Tap "Execute".

    On the screen you can see the measurement waveforms of the current, the voltage and the power. Also, the result box with numeric measurement results is shown. For details, see Chapter 15.3.1, "Power Quality Results", on page 936.

15.3.3 Power Quality Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the power measurement parameters and display settings.



Input frequency

Selects the input frequency of the source signal.

Remote command:

`POWer:QUALity:FREQ` on page 1914

Custom frequency

Sets the user-defined frequency if the "Input frequency" is set to "Custom".

Remote command:

`POWer:QUALity:FCUS` on page 1915

Optimize scaling

Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:

`POWer:QUALity:AUTO` on page 1914

Report comment

In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute

Starts the "Power Quality" measurement.

Remote command:

`POWer:QUALity:EXECute` on page 1914

15.4 Inrush Current

The "Inrush Current" analysis measures the peak of the input current that is drawn by the device, when the device is first turned on.

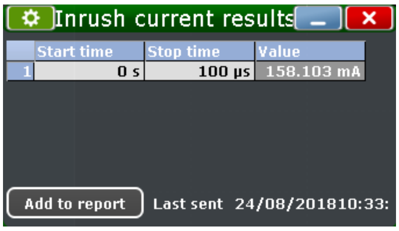
Required probes:

- Current probe

15.4.1 Inrush Current Results

The results of "Inrush Current" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the current waveform
- The result box displays the following numeric measurement results:
  – "Start time" / "Stop time" define the time period for the present gate
  – The "Value" stands for the maximum amplitude of the current for the present time period. This is the inrush current for the correspondent gate.



To measure and display the inrush current, the instrument uses the following measurements and waveforms:

- "P1" to "P5": "Meas 1" to "Meas 5" to measure the inrush current of "Gate 1" to "Gate 5"

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The following remote commands are used for handling the measurement results:

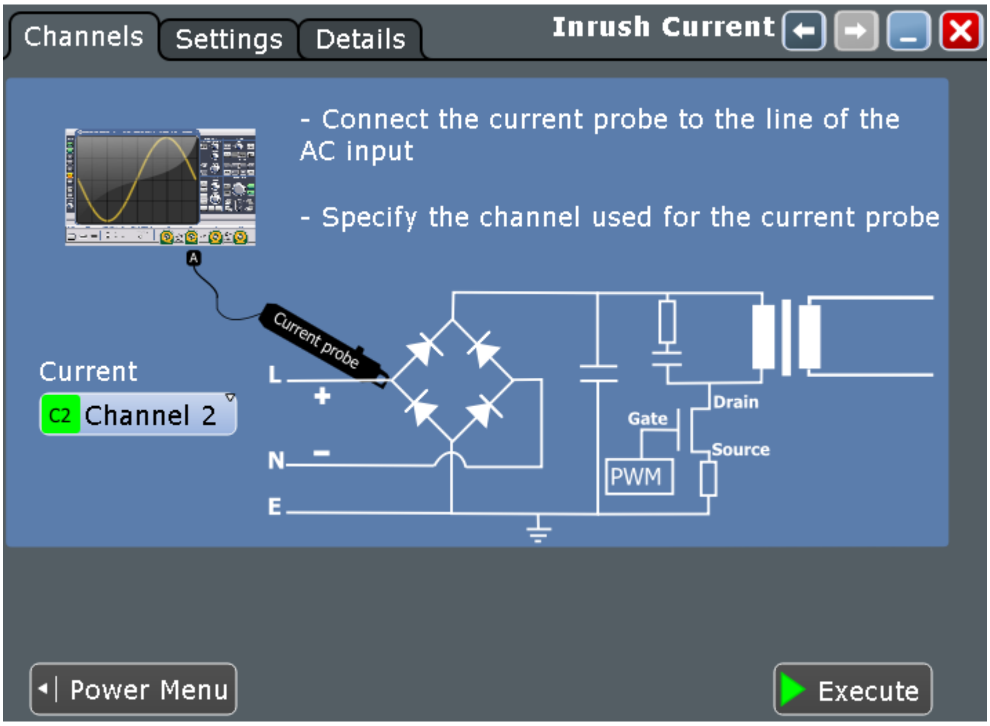
- `POWer:INRush:GATE<m>:VALue` on page 1917
- `POWer:INRush:REPort:ADD` on page 1918

15.4.2 Configuring Inrush Current

For details of the configuration settings, see Chapter 15.4, "Inrush Current", on page 941.

1. Select "Analysis" > "Power".
2. Under "Input", select "Inrush Current".
3. Connect the current probe to the oscilloscope.

4. Select the correct channel for the "Current Source".
5. Select "Vertical" > "Probe Setup" > "Channel" and set your probe parameters.
6. Connect the probes to the DUT as shown in the "Channels" tab:



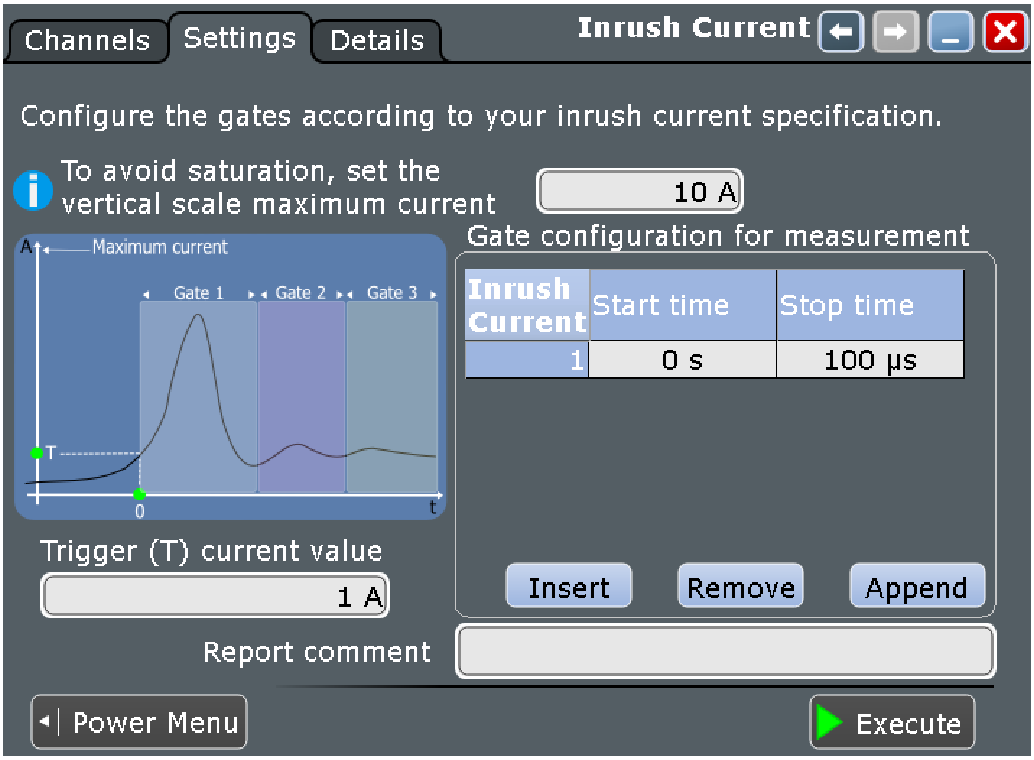
7. Select the "Settings" tab.
8. Set the "Trigger current value".
9. Set the "Maximum current" that should be displayed in the vertical scale.
10. In the "Gate configuration" table, define the different time periods. You can set up to five different gates.
11. Tap "Execute".
12. Start the DUT.

    On the screen, you can see the result box with the inrush current of each gate. For details, see Chapter 15.4.1, "Inrush Current Results", on page 941.

15.4.3 Inrush Current Settings

In the "Channels" tab, you set the current source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the inrush current measurement parameters and display settings.



Maximum current

Sets the maximum expected current for the vertical scale. Set the value according to your signal to avoid saturation.

Remote command:

POWer:INRush:MAXCurrent on page 1917

Trigger current value (T)

Sets the current value for the trigger. The measurement starts after the signal of the DUT reaches this current value.

Remote command:

POWer:INRush:TRIGger on page 1917

Gate Configuration

In this table, you can configure different gates (time periods). You can configure up to five different gates. The time periods of the defined gates may overlap.

To add a gate press "Insert" or"Append". To remove a gate press "Remove".

Remote command:

POWer:INRush:ADD on page 1916

POWer:INRush:INSert on page 1916

POWer:INRush:REMove on page 1916

Inrush current ← Gate Configuration

Shows the index of the gate.

Remote command:

POWer:INRush:COUNt? on page 1916

Start time ← Gate Configuration
Sets the start measuring time for the selected gate.

Remote command:
POWer:INRush:GATE<m>:STARt on page 1917

Stop time ← Gate Configuration
Sets the stop measuring time for the selected gate.

Remote command:
POWer:INRush:GATE<m>:STOP on page 1917

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Inrush Current " measurement.

Remote command:
POWer:INRush:EXECute on page 1917

15.5 Current Harmonic

Current harmonics appear in an electric power system due to non-linear electric loads. The harmonics can be ejected back into the AC line and disturb other equipment on the grid. To avoid this disturbance, there are often standards of compliance that consumer or industry end-products should meet.

The "Current Harmonic" analysis tests the devices according to the pre-compliance standards EN 61000-3-2, MIL-STD-1399 and RTCA DO-160.

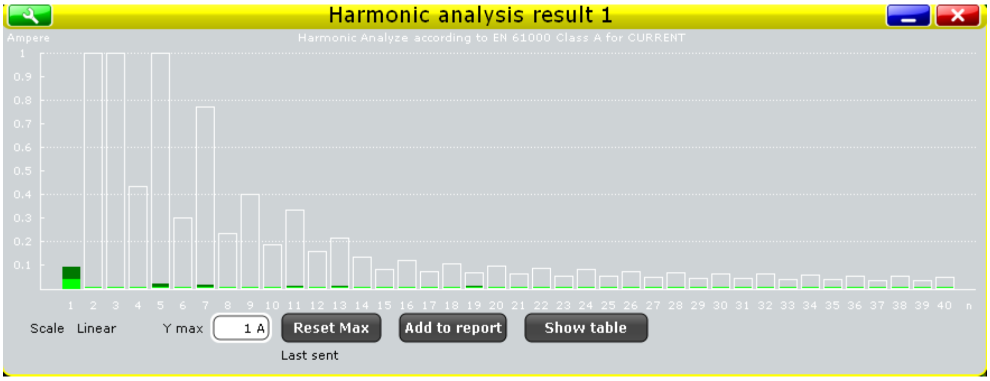
Required probes:

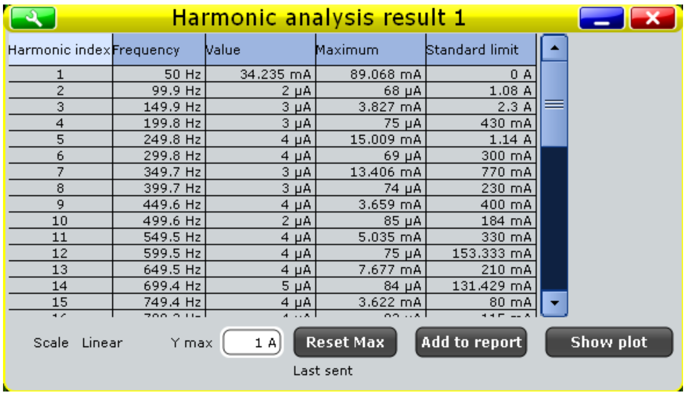
- Differential voltage probe
- Current probe

15.5.1 Current Harmonic Results

The results of "Current Harmonic" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the voltage waveform
  – the current waveform
  – the power waveform
- The result box displays a bar chart or a table with the numerical measurement results up to the 40th harmonics.
  To switch the display, tap "Show table" or "Show plot" accordingly





To measure and display the current harmonic, the instrument uses the following measurements and waveforms:

- "P6" "Meas 6" to measure the power waveform
- "P7" "Meas 7" to measure the spectrum voltage
- "P8" "Meas 8" to measure the spectrum current
- "M2" Math 2 to calculate the power
- "M3" Math 3 to calculate the FFT of the voltage
- "M4" Math 4 to calculate the FFT of the current

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The current harmonic results are defined as follows:

Result Table	Bar Chart Match	Description
Harmonic Index	Value of the X-Axis	The harmonic order
Frequency	-	The frequency value of the signal
Value	Value of the Y-Axis. Shown by a green bar	The present value of the current harmonic
Maximum	Shown by a darkened green bar	The maximum measured value

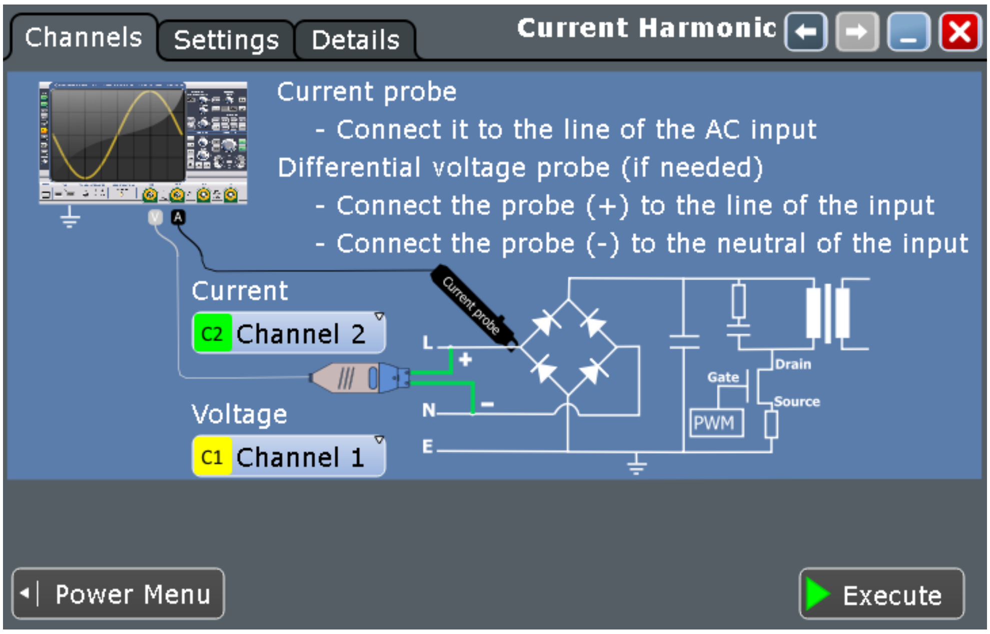
Result Table	Bar Chart Match	Description
Standard limit	Shown by a white bar	The maxim allowed value according to the selected standard
"Y max"	"Y max"	Sets the upper limit for the display of the Y scale. This value can be reset with the "Reset Max" button

The following remote commands are used for handling the measurement results:
- POWer:HARMonics:RESult<m>:FREQuency<n>:VALue? on page 1919
- POWer:HARMonics:RESult<m>:MAXValue<n>:VALue? on page 1919
- POWer:HARMonics:RESult<m>:STDinuse? on page 1919
- POWer:HARMonics:RESult<m>:STDValue<n>:VALue? on page 1919
- POWer:HARMonics:RESult<m>:VALue<n>:VALue? on page 1919
- POWer:HARMonics:REPort:ADD on page 1919

15.5.2 Configuring Current Harmonic

For details of the configuration settings, see Chapter 15.5.3, "Current Harmonic Settings", on page 947.

1. Select "Analysis">"Power".
2. Under "Power Analysis", select "Current Harmonic".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



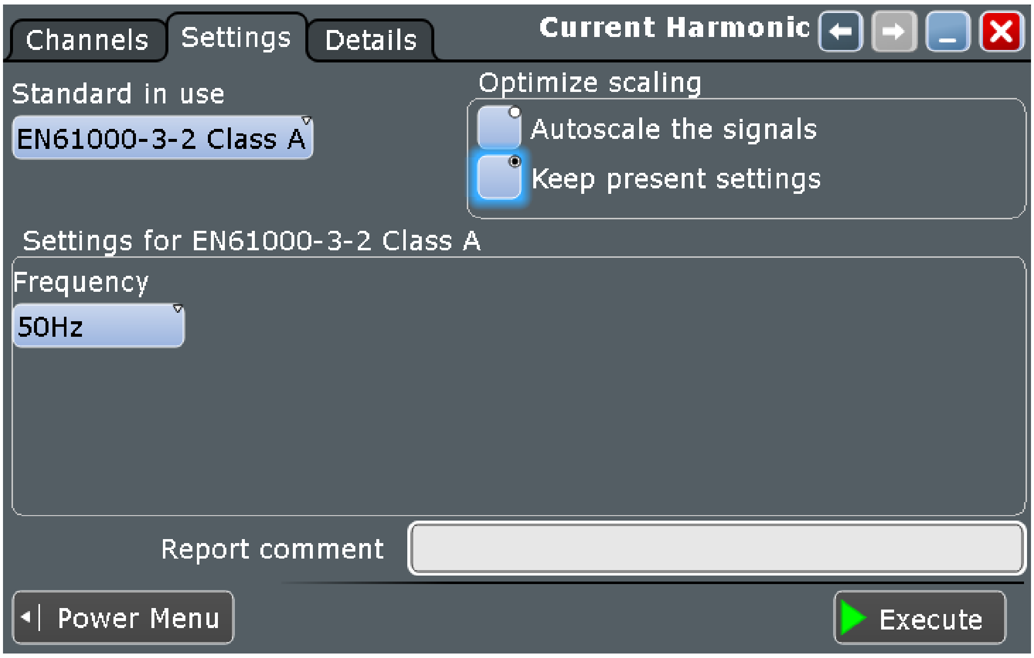
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Standard in use"
9. Set the "Frequency" according to your signal.
10. Select an"Optimize Scaling" option.
11. Tap "Execute".

    On the screen you can see the measurement of the current, the voltage and the power. Also there is a table giving information about important measurement parameters. For details, see Chapter 15.5.1, "Current Harmonic Results", on page 944.

15.5.3 Current Harmonic Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the current harmonic measurement parameters and display settings.



Standard in use

Select the standard in use. For a list of the available standards, see Table 15-1.

Table 15-1: Current Harmonic pre-compliance standards

Standard	Application
EN 61000-3-2 Class A	Balanced 3-phase equipment, household appliances (excluding equipment identified as class D), tools (excluding portable tools), dimmers for incandescent lamps, audio equipment
EN 61000-3-2 Class B	Portable tools, not professional arc welding equip-ment
EN 61000-3-2 Class C	Lighting equipment
EN 61000-3-2 Class D	PC, PC monitors, radio, or TV receivers with an input power less than or equal to 600W
MIL-STD-1399	Military shipboard user equipment
RTCA DO-160	Environmental tests of avionics hardware

Remote command:
`POWer:HARMonics:STAN` on page 1920

Frequency

Selects the frequency of the input signal.

Remote command:
`POWer:HARMonics:ENFR` on page 1918
`POWer:HARMonics:MIFR` on page 1919
`POWer:HARMonics:DOFR` on page 1918

Revised Current
Available only for "Standard" > "RTCA DO-160".

Selects how the results are evaluated. Available are the following settings:

- Evaluation with current source only
- Evaluation with voltage source and revised current law
  - Display opposite voltage harmonic result chart
  - Do not display voltage result

Remote command:
`POWer:HARMonics:EVAL` on page 1919
`POWer:HARMonics:VOLT` on page 1920

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:HARMonics:AUTO` on page 1918

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Current Harmonics" measurement.

Remote command:
`POWer:HARMonics:EXECute` on page 1919

15.6 Modulation Analysis

The "Modulation Analysis " measures the control pulse signal to a switching device.

Required probes:

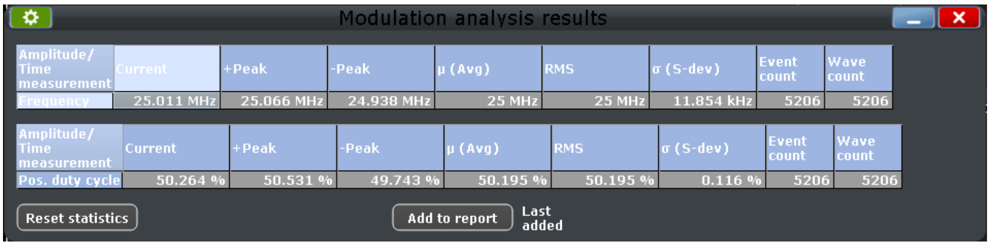
- Differential voltage probe
- Current probe

15.6.1 Modulation Analysis Results

The results of "Modulation Analysis " measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage or the current waveform
  - for "Type >Turn on" a track of the frequency and the duty cycle.

- (Optional for "Type > Continuous" measurement) Two histograms display the density distribution of the measurement results in dependence of the frequency and the positive duty cycle.
- The result box displays the numeric measurement results.



To measure and display the power quality, the instrument uses the following measurements and waveforms:

- "P7" "Meas 7" to measure the positive duty cycle
- "P8" "Meas 8" to measure the frequency

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

Table 15-2: Statistic result parameters

Label	Description
Current	Currently measured value
+Peak	Positive peak value (maximum)
-Peak	Negative peak value (minimum)
μ (Avg)	Average
RMS	Root mean square
σ (S-dev)	Standard deviation
Event count	Number of measured pulses
Wave count	Number of waveforms (acquisitions) the measurement is based on

"Modulation Analysis" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The following remote commands are used for handling the measurement results:

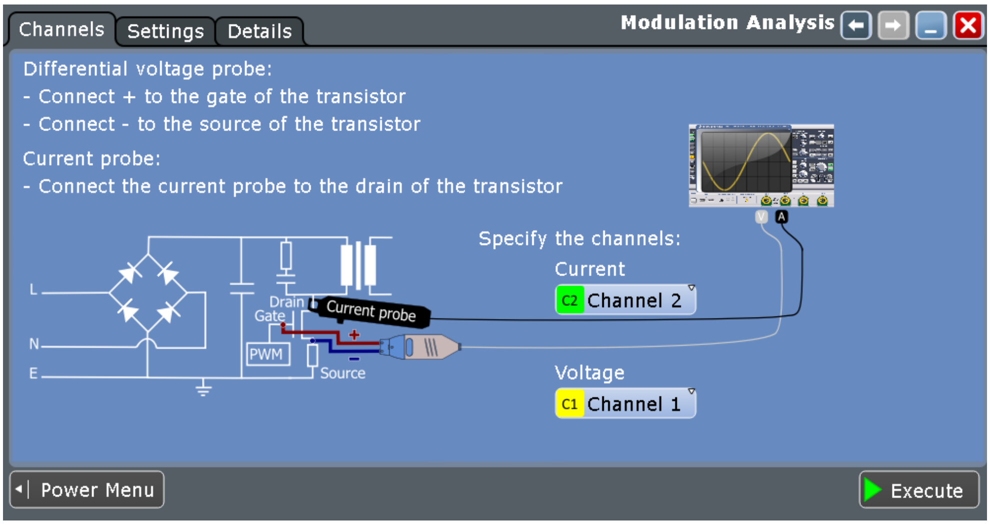
- `POWer:MODulation:RESult:ACTual?` on page 1921
- `POWer:MODulation:RESult:AVG?` on page 1921
- `POWer:MODulation:RESult:EVTCount?` on page 1921
- `POWer:MODulation:RESult:NPEak?` on page 1921
- `POWer:MODulation:RESult:PPEak?` on page 1921
- `POWer:MODulation:RESult:RMS?` on page 1921
- `POWer:MODulation:RESult:STDDev?` on page 1921
- `POWer:MODulation:RESult:WFMCount?` on page 1921

- `POWer:MODulation:REPort:ADD` on page 1921

15.6.2 Configuring Modulation Analysis

For details of the configuration settings, see Chapter 15.6.3, "Modulation Analysis Settings", on page 951.

1. Select "Analysis" > "Power".
2. Under "Switching / Control Loop", select "Modulation Analysis".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



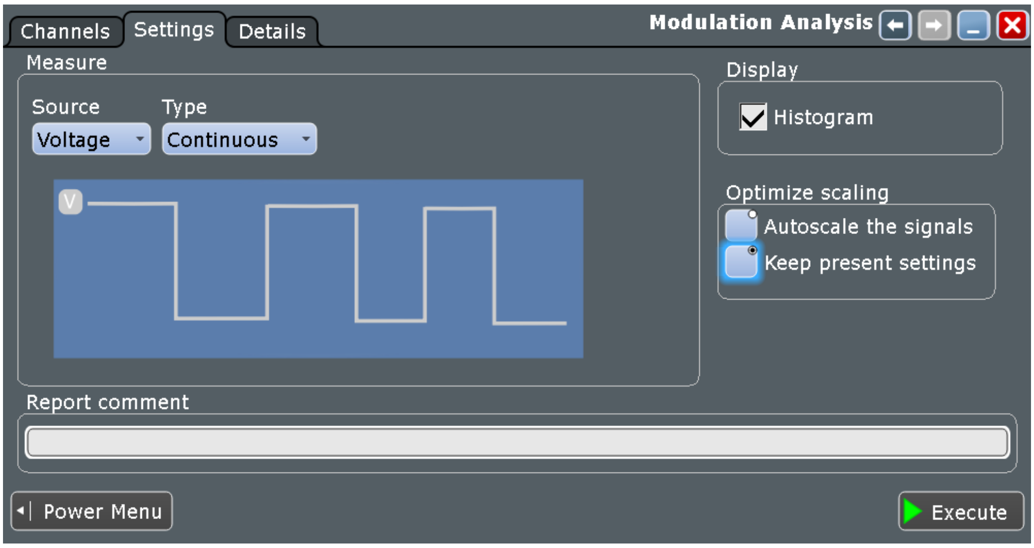
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Source" and the "Type" of measurement.
9. Select an "Optimize Scaling" option.
10. Tap "Execute".

    On the screen, you can see the measurement waveforms of the current or the voltage. Also, the result box with numeric measurement results is shown. For details, see Chapter 15.6.1, "Modulation Analysis Results", on page 949.

15.6.3 Modulation Analysis Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the modulation analysis parameters and display settings.



Source
Selects the source for the measurement.

Remote command:
`POWer:MODulation:SOURce` on page 1922

Type
Selects the type of signal flow for the measurement

"Continuous" The measurement is running continuously.

"Turn on" The measurement runs once when the DUT is turned on.

Remote command:
`POWer:MODulation:TYPE` on page 1922

Display Histogram
Available only for "Type" > "Continuous".

Enables the display of two histograms after the measurement is executed. The histograms show the density distribution of the measurement results in dependence of the frequency/ duty cycle in a graphic. Thus they illustrate the statistics of the measurements.

Remote command:
`POWer:MODulation:DHIStogram` on page 1921

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
POWer:MODulation:AUTO on page 1921

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Modulation Analysis" measurement.

Remote command:
POWer:MODulation:EXECute on page 1921

15.7 Dynamic on Resistance

The "Dynamic ON Resistance" analysis measures the resistance of a switching device, during operation. Because voltage and current may vary in time, the resistance is not constant, thus it is called dynamic ON resistance. It is defined as the ratio dV/dI.

The resistance-related voltage should be measured during a stable part of the switch node waveform, when the undershoot and ringing have decayed, after the high-to-low voltage transition.

Required probes:

- Differential voltage probe
- Current probe

15.7.1 Dynamic on Resistance Results

The results of "Dynamic on Resistance" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
- The result box displays the dynamic on resistance value.

To measure and display the dynamic on resistance, the instrument uses the following measurements and waveforms:

- "P5" "Meas 5" to measure the amplitude of the voltage
- "P6" "Meas 6" to measure the amplitude of the voltage
- "P7" "Meas 7" to measure the amplitude of the current
- "P8" "Meas 8" to measure the amplitude of the current
- "C1 " Cursor 1 to measure gate ["t_0" ,"t_1"]

- "C2 " Cursor 2 to measure gate ["t_2" ,"t_3"]

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The dynamic on resistance displayed as the result is defined as:

$$R = \frac{V(t_2) - V(t_0)}{I(t_3) - I(t_1)}$$

The points "t_0" , "t_1", "t_2" and "t_3" are defined by the cursor lines displayed in the result diagram of the measurement. You can move the cursor lines to define another area of interest.

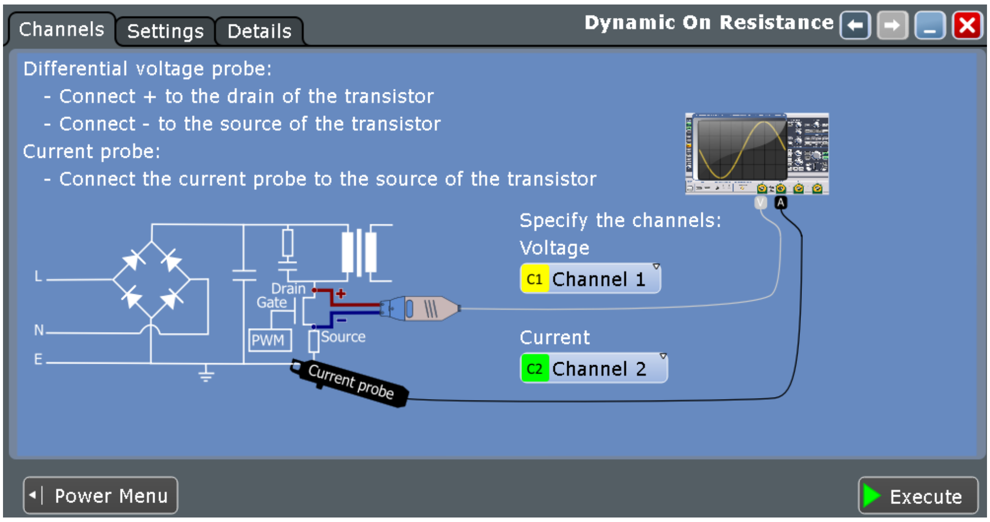
The following remote commands are used for handling the measurement results:

- `POWer:DONRes:RESult:RESistance?` on page 1923
- `POWer:DONRes:GATE<m>:STARt` on page 1923
- `POWer:DONRes:GATE<m>:STOP` on page 1923
- `POWer:DONRes:REPort:ADD` on page 1923

15.7.2 Configuring Dynamic on Resistance

For details of the configuration settings, see Chapter 15.7.3, "Dynamic on Resistance Settings", on page 955.

1. Select "Analysis" > "Power".
2. Under "Switching / Control Loop", select "Dynamic on Resistance".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



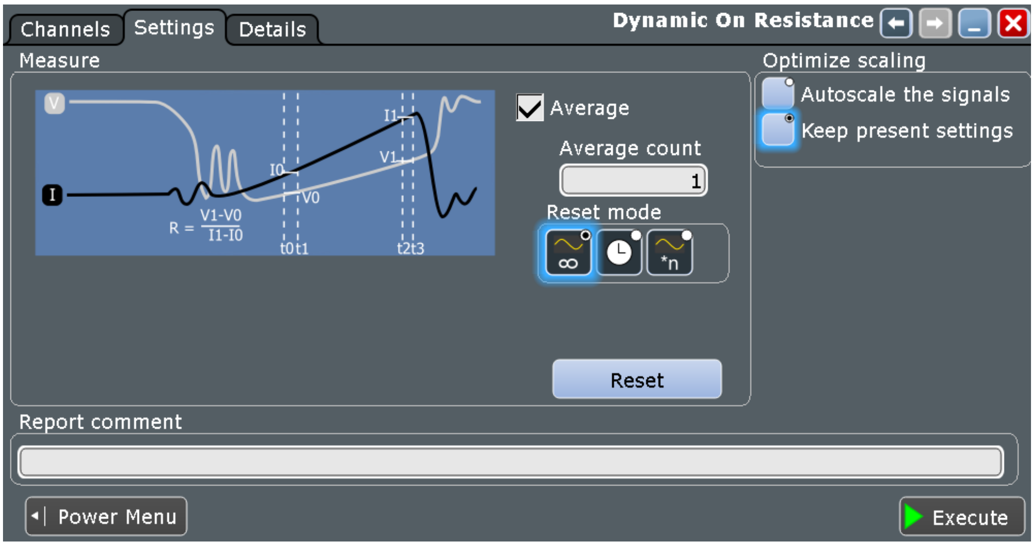
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. If "Average" is selected, enter the "Average count", that is the number of waveforms used for average calculation.
9. Set the reset condition for the average calculation:
   - If "Time" is selected, enter the "Reset time".
   - If "Waveforms" is selected, enter the "Reset count".
10. Select an "Optimize Scaling" option.
11. Tap "Execute".
12. If needed adjust the cursors manually. You can tap on a cursor and change its position with the [NAVIGATION] rotary knob.

    On the screen, you can see the measurement waveforms of the current and the voltage. Also, the result box displays the dynamic on resistance. For details, see Chapter 15.7.1, "Dynamic on Resistance Results", on page 953.

15.7.3 Dynamic on Resistance Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the power measurement parameters and display settings.



Average
Enables the "Average" method for building the resulting waveform. The average is calculated from the data of the current acquisition and several acquisitions before. The method reduces random noise and other heterodyne signals. It requires a stable, triggered and periodic signal for correct function. The number of acquisitions for average calculation is defined with "Average count", and the "Reset mode" defines the restart condition.

Remote command:
`POWer:DONRes:AVG` on page 1922

Auto reset mode / Reset mode
Defines when the envelope and average evaluation restarts.

"None" No restart, the number of acquisitions considered by the waveform arithmetics is not limited.

"Time" Restarts the envelope and average calculation after the time defined in "Reset time".

"Waveforms" Restarts the envelope and average calculation after a number of acquired waveforms defined in "Reset count".

Remote command:
`ACQuire:ARESet:MODE` on page 1924
`ACQuire:ARESet:TIME` on page 1925
`ACQuire:ARESet:COUNt` on page 1925

Reset
Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1078

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:DONRes:AUTO` on page 1922

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Dynamic on Resistance" measurement.

Remote command:
`POWer:DONRes:EXECute` on page 1923

15.8 Slew Rate

The "Slew Rate" analysis measures the rate of change of the voltage or current waveform during the switching of the switching transistor.

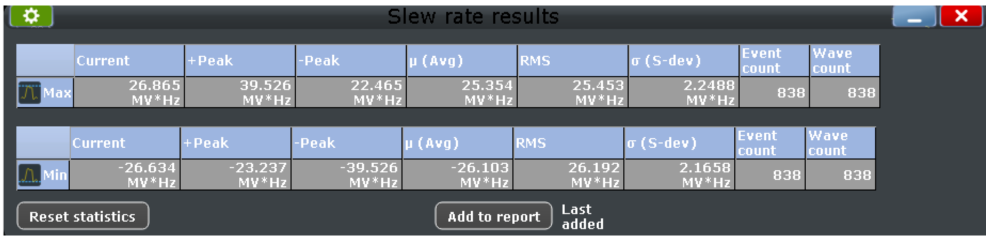
Required probes:

- Differential voltage probe
- Current probe

15.8.1 Slew Rate Results

The results of "Slew Rate" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the voltage waveform
  – the current waveform
  – a waveform of the derivative of voltage and current
- The result box displays the numeric measurement results. For a detailed description, see Table 15-2.



To measure and display the slew rate, the instrument uses the following measurements and waveforms:

- "P8" "Meas 8" to measure the amplitude of the current or voltage waveform
- "M2" Math 4 to calculate the time derivative of the current or voltage waveform
- "C1" Cursor 1 to determine the measurement area

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

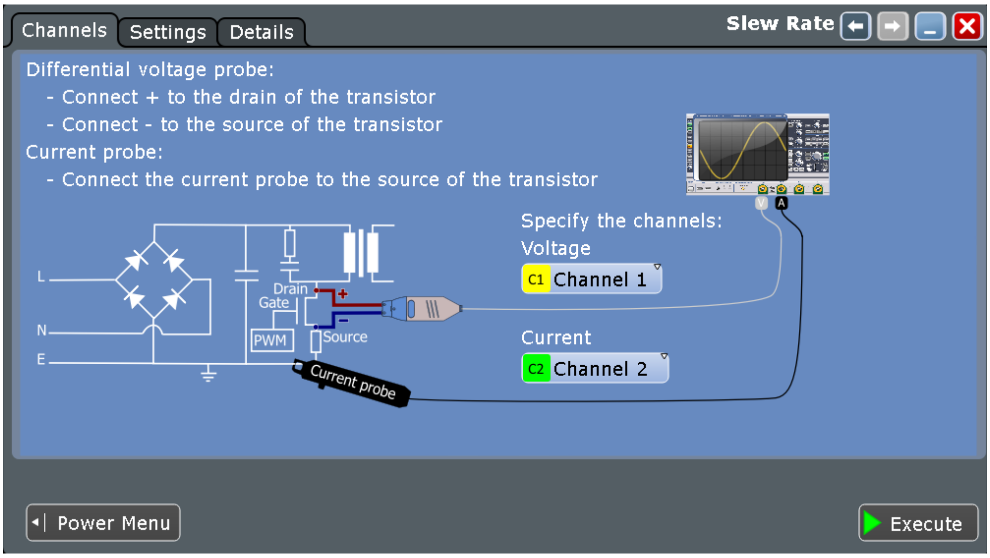
The following remote commands are used for handling the measurement results:

- `POWer:SLEWrate:RESult:ACTual?` on page 1925
- `POWer:SLEWrate:RESult:AVG?` on page 1925
- `POWer:SLEWrate:RESult:EVTCount?` on page 1925
- `POWer:SLEWrate:RESult:NPEak?` on page 1925
- `POWer:SLEWrate:RESult:PPEak?` on page 1925
- `POWer:SLEWrate:RESult:RMS?` on page 1925
- `POWer:SLEWrate:RESult:STDDev?` on page 1926
- `POWer:SLEWrate:RESult:WFMCount?` on page 1926
- `POWer:SLEWrate:REPort:ADD` on page 1925

15.8.2 Configuring Slew Rate

For details of the configuration settings, see Chapter 15.8.3, "Slew Rate Settings", on page 959.

1. Select "Analysis" > "Power".
2. Under "Switching / Control Loop", select "Slew Rate".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



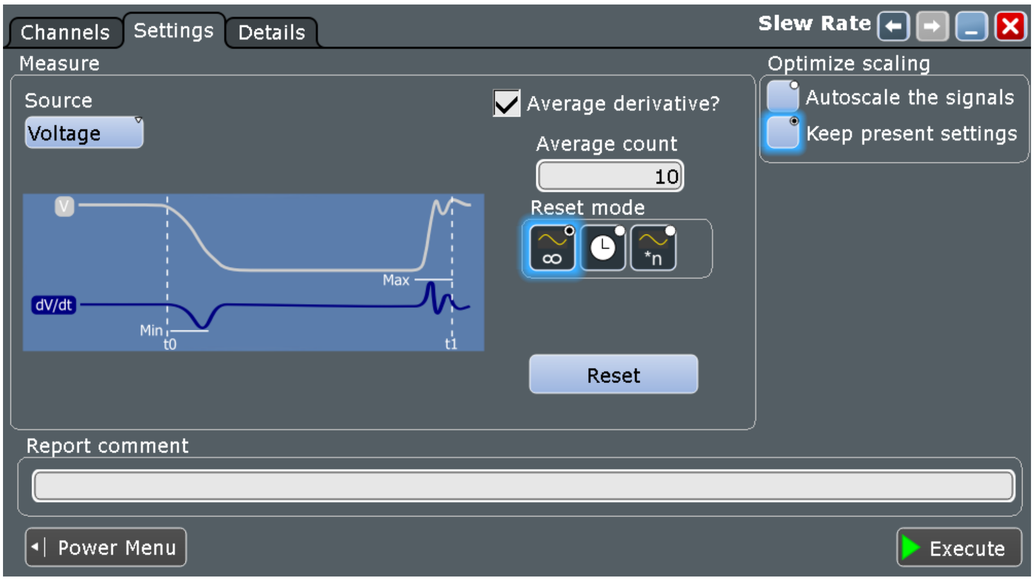
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Select the "Source".
9. If "Average" is selected, enter the "Average count", that is the number of waveforms used for average calculation.
10. Set the reset condition for the average calculation:
    - If "Time" is selected, enter the "Reset time".
    - If "Waveforms" is selected, enter the "Reset count".
11. Select an "Optimize Scaling" option.
12. Tap "Execute".

    On the screen you can see the measurement waveforms of the slew rate, the current and the voltage. The result box with numeric measurement results is shown. For details, see Chapter 15.8.1, "Slew Rate Results", on page 957.

15.8.3 Slew Rate Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the slew rate measurement parameters and display settings.



Source

Selects dV/dt or dI/dt as the source of the measurement.

Remote command:

`POWer:SLEWrate:SOURce` on page 1924

Average

Enables the "Average" method for building the resulting waveform. The average is calculated from the data of the current acquisition and several acquisitions before. The method reduces random noise and other heterodyne signals. It requires a stable, triggered and periodic signal for correct function. The number of acquisitions for average calculation is defined with "Average count", and the "Reset mode" defines the restart condition.

Remote command:

`POWer:SLEWrate:AVGDeriv` on page 1924

Auto reset mode / Reset mode

Defines when the envelope and average evaluation restarts.

"None"	No restart, the number of acquisitions considered by the waveform arithmetics is not limited.
"Time"	Restarts the envelope and average calculation after the time defined in "Reset time".
"Waveforms"	Restarts the envelope and average calculation after a number of acquired waveforms defined in "Reset count".

Remote command:

`ACQuire:ARESet:MODE` on page 1924

`ACQuire:ARESet:TIME` on page 1925

`ACQuire:ARESet:COUNt` on page 1925

Reset
Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1078

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:SLEWrate:AUTO` on page 1924

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Slew Rate" measurement.

Remote command:
`POWer:SLEWrate:EXECute` on page 1924

15.9 Safe Operating Area (S.O.A.)

The safe operating area is defined by the voltage and current conditions over which a power semiconductor device is expected to operate without self-damage. The "Safe Operating Area" analysis provides a diagram of the safe operating conditions of your device.

Required probes:

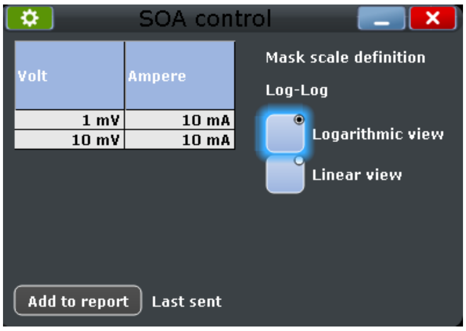
- Differential voltage probe
- Current probe

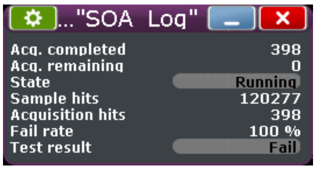
15.9.1 Safe Operating Area Results

The results of "Safe Operating Area" measurements are provided in the following ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform

- A logarithmic or linear XY diagram of the calculated voltage (x-axis) and current (y-axis) waveforms. This curve is a graphical representation of the power handling capability of the device under various conditions.
- The result box displays the numeric measurement results. Also, you can see the mask definition and change the scale in the "SOA Control" dialog. If the state of "Enable mask test" is "On" an extra result box appears, see also Table 15-3.





To measure and display the safe operating area, the instrument uses the following measurements and diagrams:

- "XY1" XY Diagram 1 to measure the logarithmic waveform
- "XY2" XY Diagram 2 to measure the linear waveform
- "M1" Math 1 to calculate the voltage signal
- "M2" Math 2 to calculate the current signal

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The results of the safe operating area mask test are described in Table 15-3.

Table 15-3: Results of the mask test

Result	Description
Acq. completed	Number of tested acquisitions
Acq. remaining	Remaining acquisitions until "Average count / Nx Single count" is reached
State	Shows if the test has been completed. The state is set to "Finished" when "Nx Single count" acquisitions are tested and the number of "Acq. remaining" is 0. As long as the number of tested acquisitions is less the "Nx Single count" number, the state is "Running". If you run the acquisition with [RUN CONT], or the number of played history acquisitions exceeds "Nx Single count", the mask testing is performed according to fail criteria settings independently of the test state. The testing is not stopped when the state is set to "Finished".

Result	Description
Sample hits	Number of samples that hit the mask
Acquisition hits	Number of acquisitions that contained at least one sample hit
Fail rate	Ratio of acquisition hits to the number of tested acquisitions
Test result	A test has failed if the number of sample hits or acquisition hits exceeds the limit of "Violation tolerance" hits

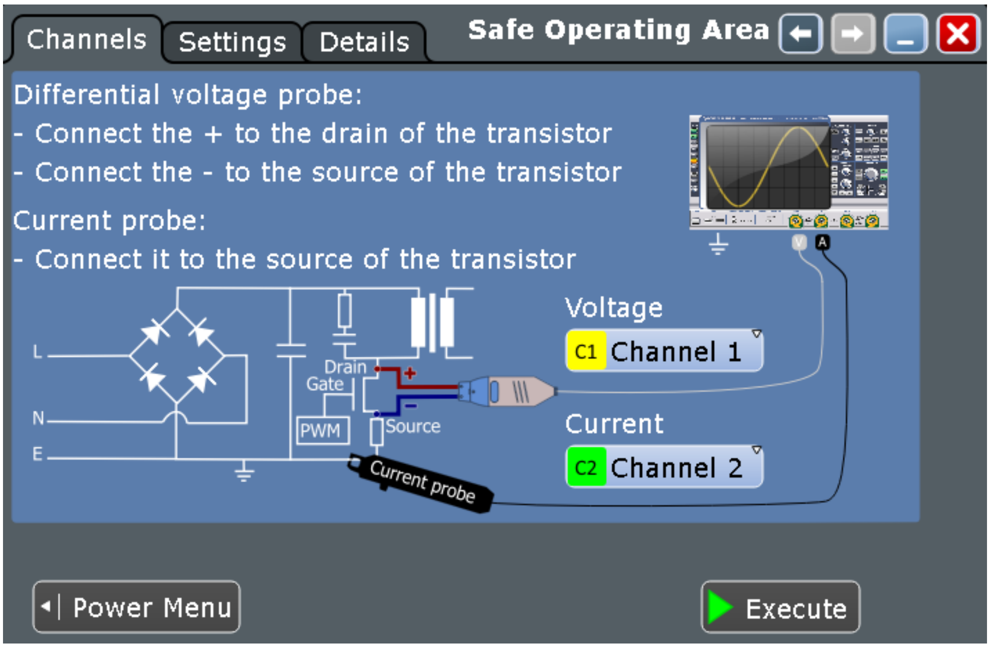
The following remote commands are used for handling the measurement results:

- POWer:SOA:SWITch on page 1928
- POWer:SOA:REPort:ADD on page 1928

15.9.2 Configuring Safe Operating Area

For details of the configuration settings, see Chapter 15.9.3, "Safe Operating Area Settings", on page 964.

1. Select "Analysis" > "Power".
2. Under "Power Path", select "Safe Operating Area".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



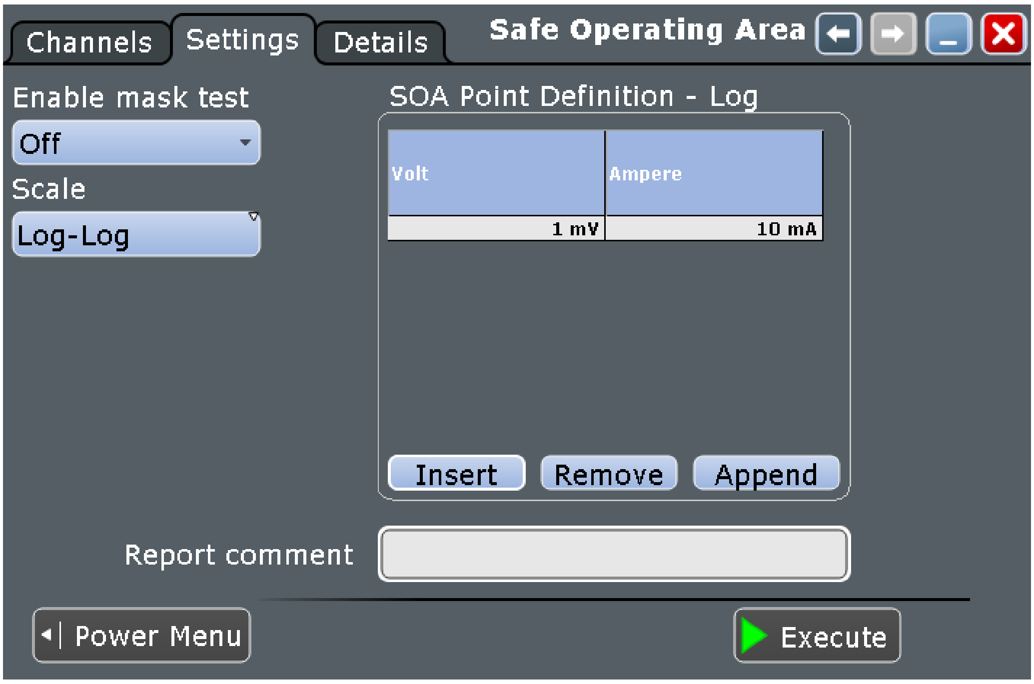
6. Select the correct channels of the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Select the state of the "Enable mask test".
9. Select the "Scale".
10. Define the SOA Points.
11. Tap "Execute".

    On the screen you can see the measurement waveforms of the Additionally, the result box with numeric measurement results is shown. For details, see Chapter 15.9.1, "Safe Operating Area Results", on page 961.

15.9.3 Safe Operating Area Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the safe operating area parameters and display settings.



Enable mask test
Enables a mask test.

Remote command:
POWer:SOA:MASK on page 1928

Scale
Selects the scale for the result diagram.

Remote command:
POWer:SOA:SCALe on page 1928

SOA point definition
In this table, you can set voltage-current points to define a mask for the safe point operating area. If "Enable mask test > On", you can check whether the signal remains within the specified limits.

To add a point press "Insert" or "Append". To remove a point press "Remove".

Remote command:
POWer:SOA:LINear:ADD on page 1926
POWer:SOA:LOGarithmic:ADD on page 1926
POWer:SOA:LINear:COUNt? on page 1927
POWer:SOA:LOGarithmic:COUNt? on page 1927
POWer:SOA:LINear:INSert on page 1927
POWer:SOA:LOGarithmic:INSert on page 1927
POWer:SOA:LINear:REMove on page 1927
POWer:SOA:LOGarithmic:REMove on page 1927

Volt ← SOA point definition
Sets the voltage value of the SOA point.

Remote command:
POWer:SOA:LINear:POINt<m>:VOLTage on page 1927

Ampere ← SOA point definition
Sets the current value of the SOA point.

Remote command:
POWer:SOA:LINear:POINt<m>:CURRent on page 1927

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Safe Operating Area" measurement.

Remote command:
POWer:SOA:EXECute on page 1926

15.10 Turn On/Off

"Turn On/Off" analysis measures the time that a power supply needs to reach a certain percentage of the steady state output level when initially turned on or turned off.

Common measuring scenarios include:

- Turn on time: measurement of the time it takes for the DC output to reach 90 % of the expected steady state level, after the power supply is initially turned on.

- Turn off time: measurement of the time it takes for the DC output to reach 10 % of the expected steady state level, after the power supply is initially turned off.

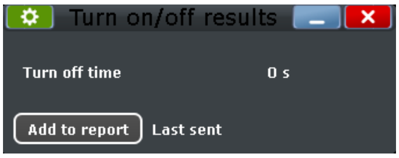
Required probes:

- Differential voltage probe
- Passive or differential voltage probe

15.10.1 Turn on/off Results

The results of "Turn on/off" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the input voltage waveform
  – the output voltage waveform
- The result box displays the "Turn on time" or the "Turn off time".



The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The"Turn on time" is measured as the time between the trigger point ("Trigger level on" value is reached) and the time the given percentage of the "Steady state level" is reached.

The"Turn off time" is measured as the time between the trigger point, delayed with the set "Time" ("Trigger level on" value is reached) and the time the given percentage of the "Steady state level" is reached, see

See also Figure 15-2 and Figure 15-3

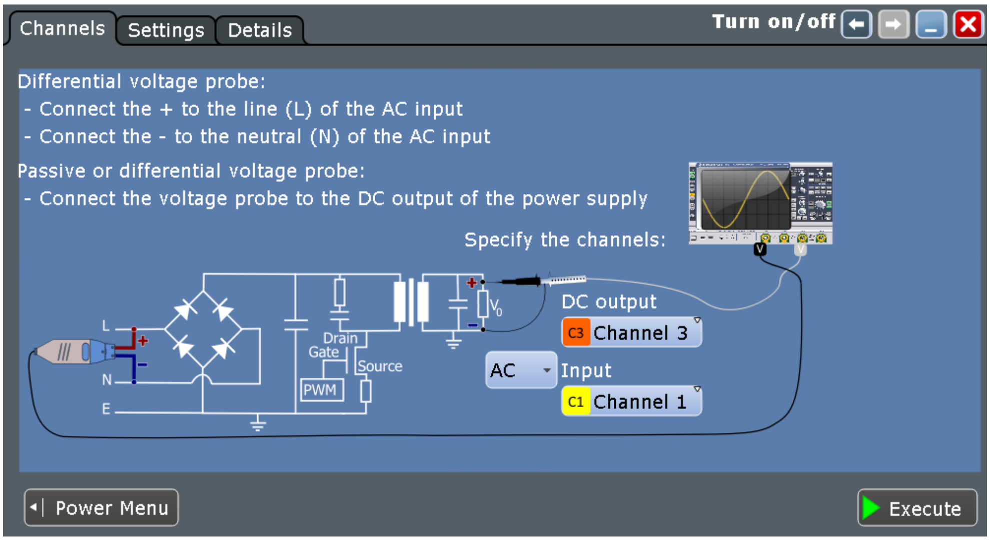
The following remote commands are used for handling the measurement results:

- `POWer:ONOFf:RESult:TOFF?` on page 1930
- `POWer:ONOFf:RESult:TON?` on page 1930
- `POWer:ONOFf:REPort:ADD` on page 1929

15.10.2 Configuring Turn On/ Off

For details of the configuration settings, see Chapter 15.10.3, "Turn On/Off Settings", on page 967.

1. Select "Analysis">"Power".
2. Under "Power Path", select "Turn on/off ".
3. Connect the probes to the DUT as shown in the "Channels" tab:



4. Select the correct channels for the "DC output" and the "AC input" or the "DC input".
5. Select the "Settings" tab.
6. Select whether you want to measure "Turn on" or "Turn off".
7. Set the "Steady state level" and the "Trigger level " according to your requirements.
8. Tap "Execute".
9. Turn on/off the DUT.

   On the screen, you can see the measurement. Also, the result box displays the turn on or the turn off time. For details, see Chapter 15.10.1, "Turn on/off Results", on page 966.

15.10.3 Turn On/Off Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the turn on and the turn off parameters.

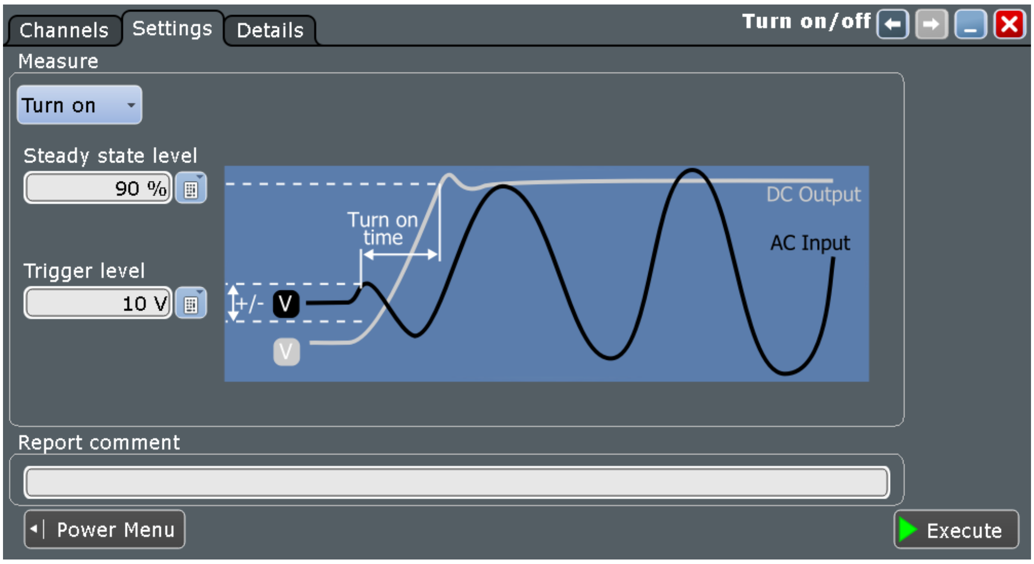


Figure 15-2: Settings turn on time

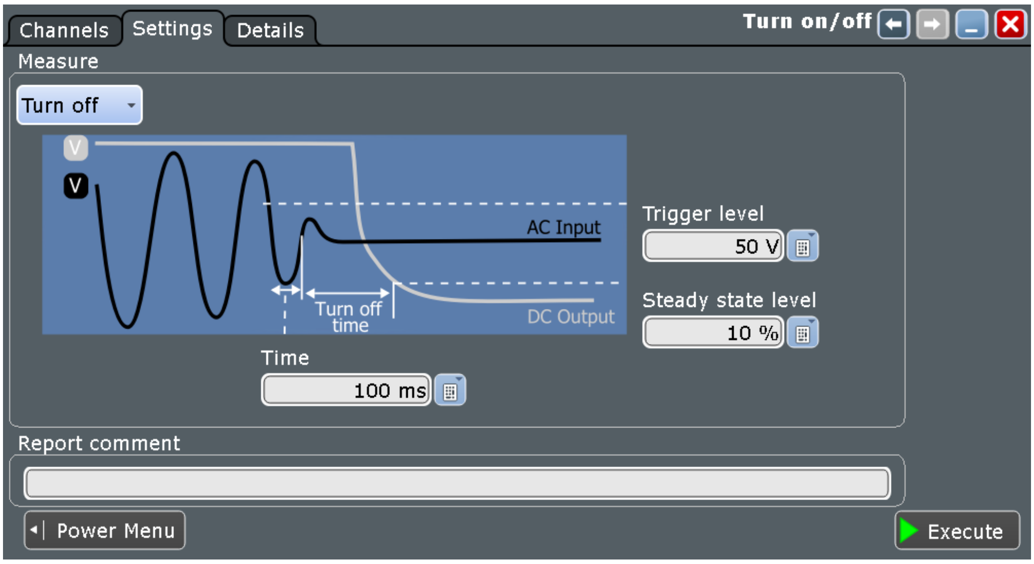


Figure 15-3: Settings turn off time

Input Type

To access this setting, select the "Channels" tab.

Selects the AC or DC input type.

Remote command:

`POWer:ONOFf:INPut` on page 1929

Measurement Type

Selects the"Turn on" or the "Turn off " measurement.

Remote command:

`POWer:ONOFf:TYPE` on page 1930

Turn on
Enables the configuration of the turn on time measurement.

Steady state level-Turn on ← Turn on
Sets the percentage of the steady state level of the DC output that has to be reached.

Remote command:
`POWer:ONOFf:DSON` on page 1929

Trigger level on ← Turn on
Triggers the beginning of the measurements at the moment the AC or DC input voltage reaches the set value.

Remote command:
`POWer:ONOFf:ATON` on page 1929
`POWer:ONOFf:DTON` on page 1929

Turn off
Enables the configuration of the turn off time measurement.

Steady state level- Turn off ← Turn off
Sets the percentage of the steady state level of the DC output that has to be reached.

Remote command:
`POWer:ONOFf:DSOFf` on page 1929

Trigger level ← Turn off
Triggers the beginning of the measurements at the moment the AC or DC input voltage reaches the set value.

Remote command:
`POWer:ONOFf:ATOFf` on page 1929
`POWer:ONOFf:DTOFf` on page 1929

Time ← Turn off
Sets the time, the start of the measurement of the turn off time is delayed with, after the trigger point.

Remote command:
`POWer:ONOFf:TIME` on page 1930

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Power Quality" measurement.

Remote command:
`POWer:ONOFf:EXECute` on page 1929

15.11 Switching Loss

The "Switching Loss" analysis measures the power and energy losses of a switching device, that occur during the switching phases and the conduction phase of the switching transistor.

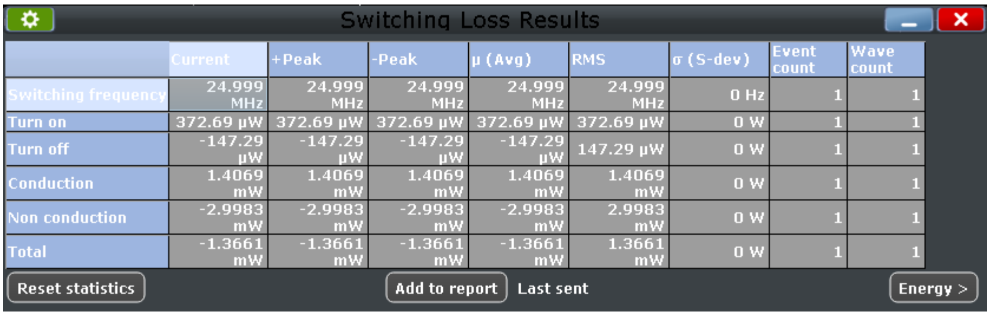
Required probes:

- Differential voltage probe
- Current probe

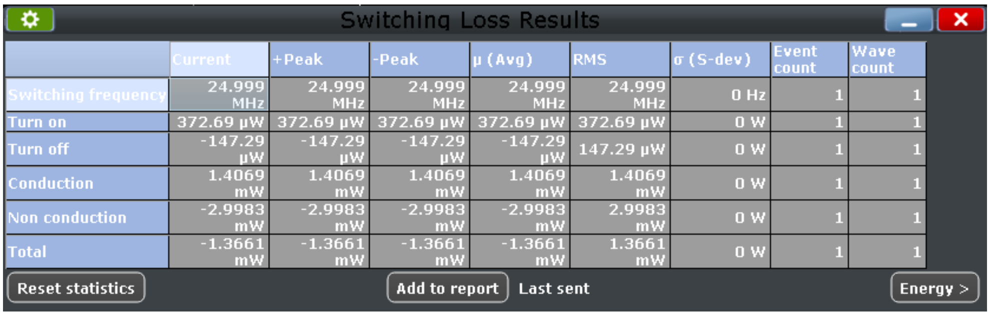
15.11.1 Switching Loss Results

The results of "Switching Loss" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the voltage waveform
  – the current waveform
  – the power waveform
- The result box displays the numeric measurement results for the enabled measurement parameters in dependence of the energy and the power. To switch the display, tap "Energy" or "Power" accordingly.
  For a detailed description, see Table 15-2.



	Current	+Peak	-Peak	μ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Switching frequency	24.999 MHz	24.999 MHz	24.999 MHz	24.999 MHz	24.999 MHz	0 Hz	1	1
Turn on	372.69 µW	372.69 µW	372.69 µW	372.69 µW	372.69 µW	0 W	1	1
Turn off	-147.29 µW	-147.29 µW	-147.29 µW	-147.29 µW	147.29 µW	0 W	1	1
Conduction	1.4069 mW	1.4069 mW	1.4069 mW	1.4069 mW	1.4069 mW	0 W	1	1
Non conduction	-2.9983 mW	-2.9983 mW	-2.9983 mW	-2.9983 mW	2.9983 mW	0 W	1	1
Total	-1.3661 mW	-1.3661 mW	-1.3661 mW	-1.3661 mW	1.3661 mW	0 W	1	1



	Current	+Peak	-Peak	μ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Switching frequency	24.999 MHz	24.999 MHz	24.999 MHz	24.999 MHz	24.999 MHz	0 Hz	1	1
Turn on	372.69 µW	372.69 µW	372.69 µW	372.69 µW	372.69 µW	0 W	1	1
Turn off	-147.29 µW	-147.29 µW	-147.29 µW	-147.29 µW	147.29 µW	0 W	1	1
Conduction	1.4069 mW	1.4069 mW	1.4069 mW	1.4069 mW	1.4069 mW	0 W	1	1
Non conduction	-2.9983 mW	-2.9983 mW	-2.9983 mW	-2.9983 mW	2.9983 mW	0 W	1	1
Total	-1.3661 mW	-1.3661 mW	-1.3661 mW	-1.3661 mW	1.3661 mW	0 W	1	1

To measure and display the switching loss, the instrument uses the following measurements and cursors:

- "P3"... "P8": "Meas 3" ... "Meas 8" to measure the voltage
- "M4" Math 4 to calculate the power
- "C3" Cursor 3 to define time points "t_0" and "t_1"

- "C4" Cursor 4 to define time points "t_2" and "t_3"

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

"Switching Loss" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The switching loss phases that can be defined during the measurement are shown in Figure 15-4 and described in Table 15-4.

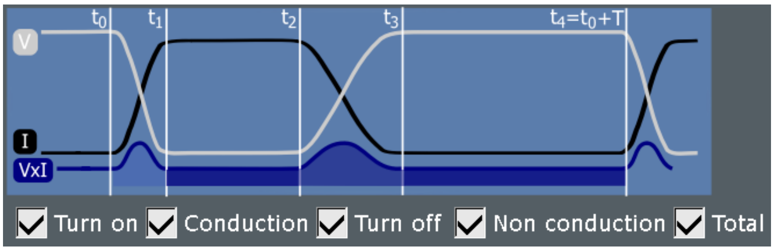


Figure 15-4: Switching loss phases

Table 15-4: Switching loss phases

Phase	Definition Points	Description
Turn on	The area between "t_0" and "t_1"	The time after switching the device, during which the current rises until it reaches the saturation current level.
Conduction	The area between "t_1" and "t_2"	The time during which the voltage is at the transistors saturated minimum and the current flows.
Turn off	The area between "t_2" and "t_3"	The time during which after a short delay time the voltage rises until it reaches its final value.
Non conduction	The area between "t_3" and "t_4"	The time during current does not flow. The losses during this period should be theoretically zero.
Total	The area between "t_0" and "t_4"	The period of one switching cycle.

The following remote commands are used for handling the measurement results:

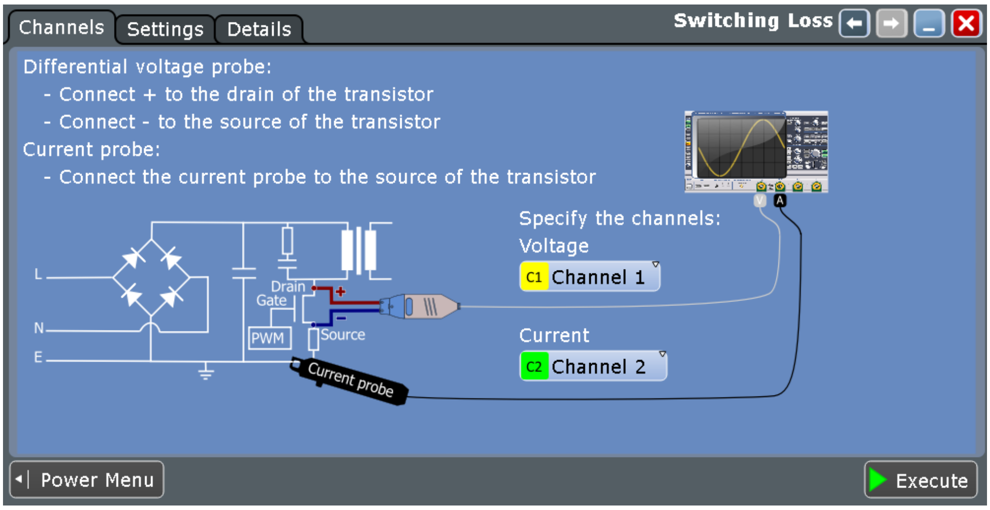
- POWer:SWITching:GATE:COND:STARt on page 1932
- POWer:SWITching:GATE:COND:STOP on page 1932
- POWer:SWITching:GATE:NCON:STARt on page 1932
- POWer:SWITching:GATE:TOFF:STARt on page 1932
- POWer:SWITching:GATE:TOFF:STOP on page 1932
- POWer:SWITching:GATE:TON:STARt on page 1932
- POWer:SWITching:GATE:TON:STOP on page 1932
- POWer:SWITching:RESult:ENERgy:ACTual? on page 1932
- POWer:SWITching:RESult:ENERgy:AVG? on page 1932
- POWer:SWITching:RESult:ENERgy:EVTCount? on page 1932
- POWer:SWITching:RESult:ENERgy:NPEak? on page 1932

- POWer:SWITching:RESult:ENERgy:PPEak? on page 1932
- POWer:SWITching:RESult:ENERgy:RMS? on page 1932
- POWer:SWITching:RESult:ENERgy:STDDev? on page 1932
- POWer:SWITching:RESult:ENERgy:WFMCount? on page 1932
- POWer:SWITching:RESult:POWer:ACTual? on page 1933
- POWer:SWITching:RESult:POWer:AVG? on page 1933
- POWer:SWITching:RESult:POWer:EVTCount? on page 1933
- POWer:SWITching:RESult:POWer:NPEak? on page 1933
- POWer:SWITching:RESult:POWer:PPEak? on page 1933
- POWer:SWITching:RESult:POWer:RMS? on page 1933
- POWer:SWITching:RESult:POWer:STDDev? on page 1933
- POWer:SWITching:RESult:POWer:WFMCount? on page 1933
- POWer:SWITching:REPort:ADD on page 1931

15.11.2 Configuring Switching Loss

For details of the configuration settings, see Chapter 15.11.3, "Switching Loss Settings", on page 973.

1. Select "Analysis" > "Power".
2. Under "Power Path", select "Switching Loss".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.

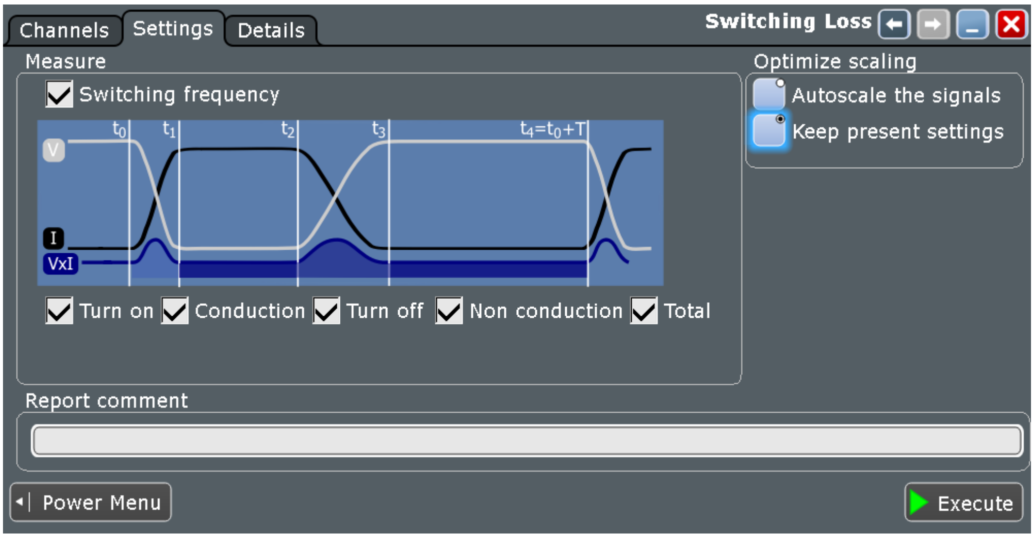
8. Enable the parameters you want to measure.
9. Select an"Optimize Scaling" option.
10. Tap "Execute".

    On the screen you can see the measurement waveforms of the current, the voltage and the power. Also, the result box with numeric measurement results is shown. For details, see Chapter 15.11.1, "Switching Loss Results", on page 970.

15.11.3 Switching Loss Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the switching loss parameters and display settings.



Measure
In this area, you can select the parameters that are included in the analysis after executing the measurement.

Switching frequency ← Measure
Enables the measurements of the switching frequency. If disabled you can enter the value of the switching frequency.

Remote command:
POWer:SWITching:SWIFrequency on page 1931
POWer:SWITching:SWIT on page 1932

Turn on
Enables the measurements during the turn on period.

Remote command:
POWer:SWITching:TON on page 1932

Conduction
Enables the measurements during the conduction period.

Remote command:
`POWer:SWITching:COND` on page 1932

Turn off
Enables the measurements during the turn off period.

Remote command:
`POWer:SWITching:TOFF` on page 1932

Non conduction
Enables the measurements during the non conduction period.

Remote command:
`POWer:SWITching:NCON` on page 1932

Total
Enables the measurements of the total period

Remote command:
`POWer:SWITching:TOTal` on page 1932

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:SWITching:AUTO` on page 1931

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Switching Loss" measurement.

Remote command:
`POWer:SWITching:EXECute` on page 1931

15.12 Power Efficiency

This measurement requires a 4-channel oscilloscope (R&S RTExxx4).

"Power Efficiency" analysis measures the input and the output power of a power supply. The power efficiency of the power supply is then calculated as the ratio of the output power and the input power.

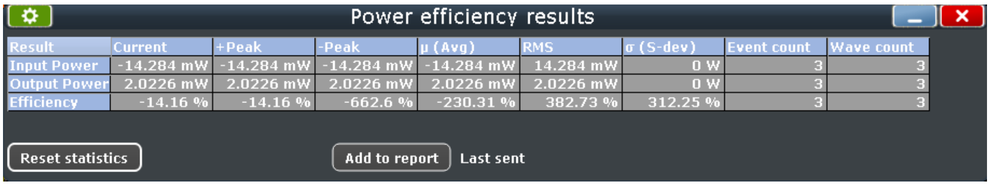
Required probes:

- Two differential voltage probes
- Two current probes

15.12.1 Power Efficiency Results

The results of "Power Efficiency" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage input waveform
  - the current input waveform
  - the voltage output waveform
  - the voltage input waveform
  - the power input waveform
  - the power output waveform
- The result box displays the numeric measurement results of the "Input power", "Output power" and the "Efficiency". For a detailed description, see Table 15-2.



Power efficiency results

Result	Current	+Peak	-Peak	μ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Input Power	-14.284 mW	-14.284 mW	-14.284 mW	-14.284 mW	14.284 mW	0 W	3	3
Output Power	2.0226 mW	2.0226 mW	2.0226 mW	2.0226 mW	2.0226 mW	0 W	3	3
Efficiency	-14.16 %	-14.16 %	-662.6 %	-230.31 %	382.73 %	312.25 %	3	3

To measure and display the power quality, the instrument uses the following measurements and waveforms:

- "P7" "Meas 7" to measure the input power waveform
- "P8" "Meas 8" to measure the output power waveform
- "M2" Math 2 to calculate the input power
- "M3" Math 3 to calculate the output power

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

"Power Efficiency" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The following remote commands are used for handling the measurement results:

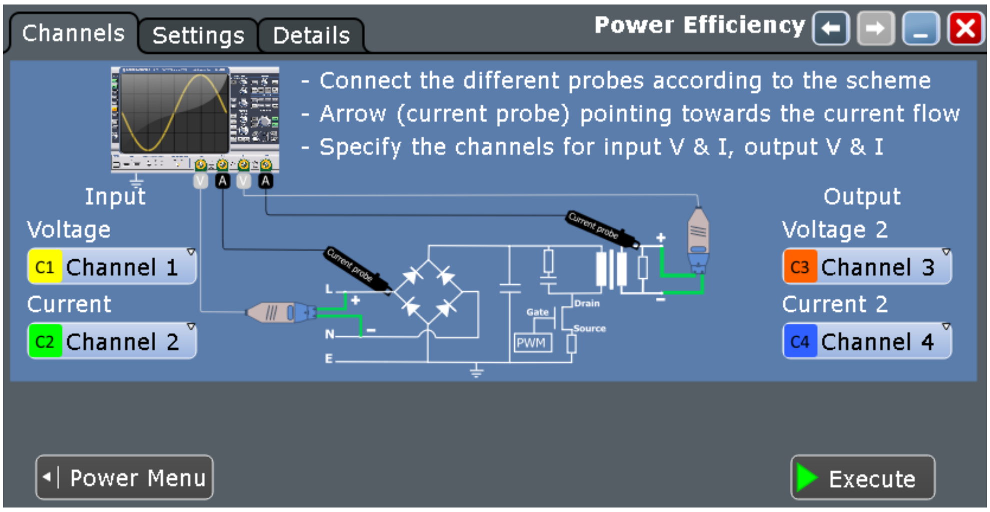
- `POWer:EFFiciency:RESult<m>:ACTual?` on page 1934
- `POWer:EFFiciency:RESult<m>:AVG?` on page 1934
- `POWer:EFFiciency:RESult<m>:EVTCount?` on page 1934
- `POWer:EFFiciency:RESult<m>:NPEak?` on page 1934
- `POWer:EFFiciency:RESult<m>:PPEak?` on page 1934
- `POWer:EFFiciency:RESult<m>:RMS?` on page 1934
- `POWer:EFFiciency:RESult<m>:STDDev?` on page 1934
- `POWer:EFFiciency:RESult<m>:WFMCount?` on page 1934

- `POWer:EFFiciency:REPort:ADD` on page 1934

15.12.2 Configuring Power Efficiency

For details of the configuration settings, see Chapter 15.12.3, "Power Efficiency Settings", on page 976.

1. Select "Analysis" > "Power".
2. Under "Power Path", select "Power Efficiency".
3. Connect the differential voltage probes and the current probes to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the graphic of the "Channels" tab:



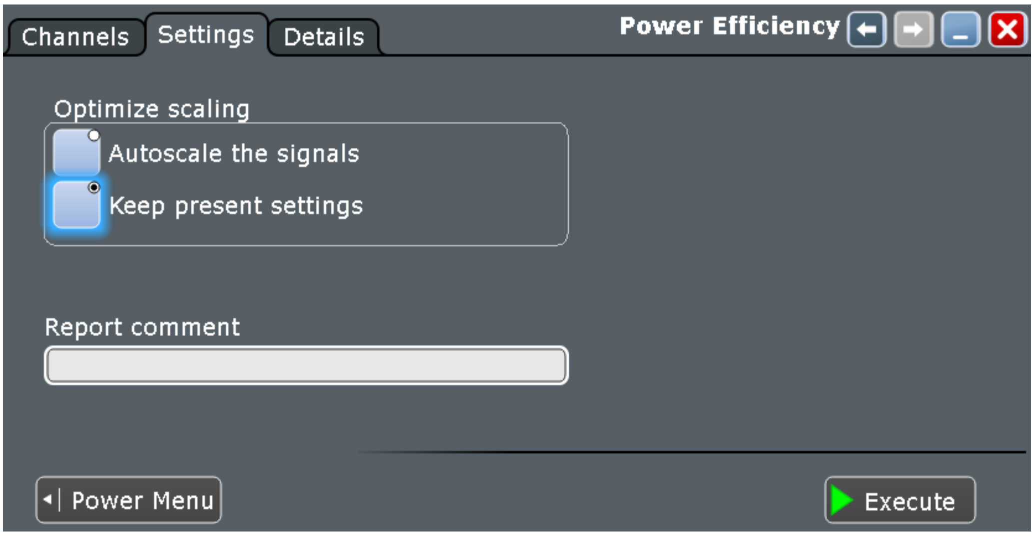
6. Select the correct channels for the "Current Source" and the "Voltage Source" of the input and the output.
7. Select the "Settings" tab.
8. Select an "Optimize Scaling" option.
9. Tap "Execute".

   On the screen, you can see the measurement waveforms of the input power and the output power. Also, the result box with numeric measurement results is shown. For details, see Chapter 15.12.1, "Power Efficiency Results", on page 975

15.12.3 Power Efficiency Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the power efficiency display settings.



Optimize scaling

Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:EFFiciency:AUTO` on page 1933

Report comment

In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute

Starts the "Power Effieciency" measurement.

Remote command:
`POWer:EFFiciency:EXECute` on page 1934

15.13 Output Ripple

The "Output Ripple" analysis measures the ripple of the device output. You can measure the voltage ripple alone or the voltage and the current ripple simultaneously. In this measurement, the peak-to peak extremes of the output DC signal are of interest. The measurement also includes the AC-RMS of the output DC signal, that is calculated as a standard derivation.

Required probes:

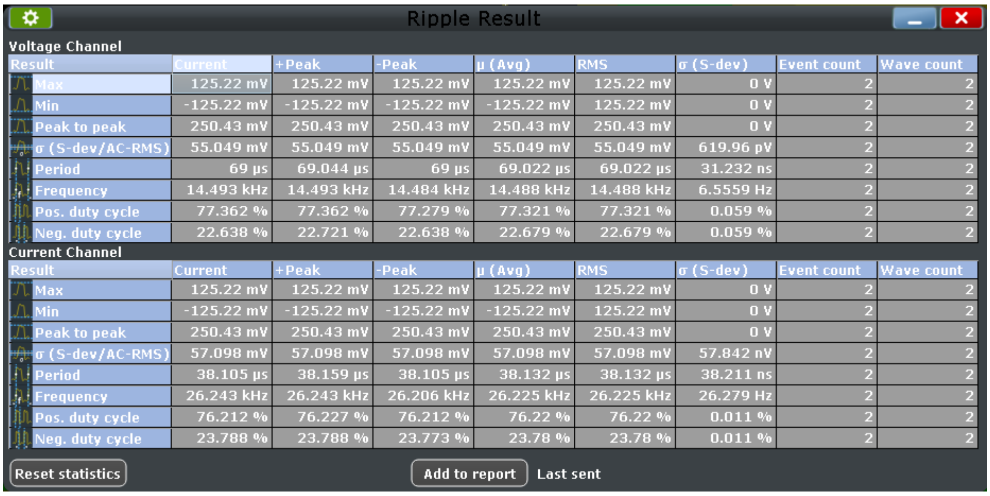
- Voltage probe

- (Optional) Current probe

15.13.1 Output Ripple Results

The results of "Output Ripple" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  – the voltage waveform
  – (optional) the current waveform
- The result box displays the numeric measurement results for the voltage and for the current ripple. For details, see Table 15-5 and Table 15-2.



Result	Current	+Peak	-Peak	µ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Max	125.22 mV	125.22 mV	125.22 mV	125.22 mV	125.22 mV	0 V	2	2
Min	-125.22 mV	-125.22 mV	-125.22 mV	-125.22 mV	125.22 mV	0 V	2	2
Peak to peak	250.43 mV	250.43 mV	250.43 mV	250.43 mV	250.43 mV	0 V	2	2
σ (S-dev/AC-RMS)	55.049 mV	55.049 mV	55.049 mV	55.049 mV	55.049 mV	619.96 pV	2	2
Period	69 µs	69.044 µs	69 µs	69.022 µs	69.022 µs	31.232 ns	2	2
Frequency	14.493 kHz	14.493 kHz	14.484 kHz	14.488 kHz	14.488 kHz	6.5559 Hz	2	2
Pos. duty cycle	77.362 %	77.362 %	77.279 %	77.321 %	77.321 %	0.059 %	2	2
Neg. duty cycle	22.638 %	22.721 %	22.638 %	22.679 %	22.679 %	0.059 %	2	2

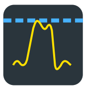
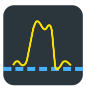
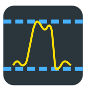
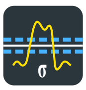
Result	Current	+Peak	-Peak	µ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Max	125.22 mV	125.22 mV	125.22 mV	125.22 mV	125.22 mV	0 V	2	2
Min	-125.22 mV	-125.22 mV	-125.22 mV	-125.22 mV	125.22 mV	0 V	2	2
Peak to peak	250.43 mV	250.43 mV	250.43 mV	250.43 mV	250.43 mV	0 V	2	2
σ (S-dev/AC-RMS)	57.098 mV	57.098 mV	57.098 mV	57.098 mV	57.098 mV	57.842 nV	2	2
Period	38.105 µs	38.159 µs	38.105 µs	38.132 µs	38.132 µs	38.211 ns	2	2
Frequency	26.243 kHz	26.243 kHz	26.206 kHz	26.225 kHz	26.225 kHz	26.279 Hz	2	2
Pos. duty cycle	76.212 %	76.227 %	76.212 %	76.22 %	76.22 %	0.011 %	2	2
Neg. duty cycle	23.788 %	23.788 %	23.773 %	23.78 %	23.78 %	0.011 %	2	2

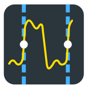
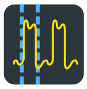
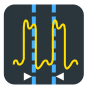
To measure and display the output ripple, the instrument uses the following measurements and waveforms:

- "P7" "Meas 7" to measure the current
- "P8" "Meas 8" to measure the voltage

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

Table 15-5: Properties output ripple

	Meas. type	Symbol	Description/Result
	Max	X_{Max}	Absolute maximum value of the waveform.
	Min	X_{Min}	Absolute minimum value of the waveform.
	Peak to peak	X_{PkPk}	Peak-to-peak value of the waveform: the difference of maximum and minimum values. $X_{Ampl} = X_{Max} - X_{Min}$
	σ (S-dev/AC-RMS)	σ_X	Standard deviation of the waveform samples

	Meas. type	Symbol	Description/Result
	Period	T_{Period}	Time of the left-most signal period of the waveform - the time difference between two consecutive waveform edges measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible.
	Frequency	f_{Period}	Frequency of the signal, reciprocal value of the period. $f_{Period} = 1 / T_{Period}$
	Pos. duty cycle	R_{PosCyc}	Positive duty cycle: Width of a positive pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $R_{PosCyc} = \frac{T_{PosPulse}}{T_{Period}} \cdot 100\%$
	Neg. duty cycle	R_{NegCyc}	Negative duty cycle: Width of a negative pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $R_{NegCyc} = \frac{T_{NegPulse}}{T_{Period}} \cdot 100\%$

"Ripple" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The following remote commands are used for handling the measurement results:

- POWer:RIPPle:RESult:FREQuency:AVG? on page 1937
- POWer:RIPPle:RESult:FREQuency:EVTCount? on page 1937
- POWer:RIPPle:RESult:FREQuency:NPEak? on page 1937
- POWer:RIPPle:RESult:FREQuency:PPEak? on page 1937
- POWer:RIPPle:RESult:FREQuency:RMS? on page 1937
- POWer:RIPPle:RESult:FREQuency:STDDev? on page 1937
- POWer:RIPPle:RESult:FREQuency:WFMCount? on page 1937
- POWer:RIPPle:RESult:FREQuency[:ACTual]? on page 1937
- POWer:RIPPle:RESult:MAXimum:AVG? on page 1937
- POWer:RIPPle:RESult:MAXimum:EVTCount? on page 1937
- POWer:RIPPle:RESult:MAXimum:NPEak? on page 1937
- POWer:RIPPle:RESult:MAXimum:PPEak? on page 1937
- POWer:RIPPle:RESult:MAXimum:RMS? on page 1937
- POWer:RIPPle:RESult:MAXimum:STDDev? on page 1937
- POWer:RIPPle:RESult:MAXimum:WFMCount? on page 1937
- POWer:RIPPle:RESult:MAXimum[:ACTual]? on page 1937
- POWer:RIPPle:RESult:MINimum:AVG? on page 1938
- POWer:RIPPle:RESult:MINimum:EVTCount? on page 1938

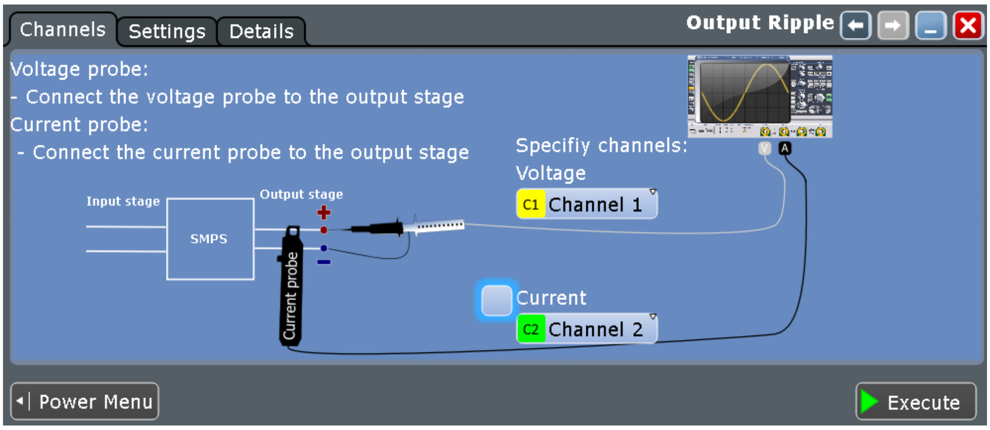
- POWer:RIPPle:RESult:MINimum:NPEak? on page 1938
- POWer:RIPPle:RESult:MINimum:PPEak? on page 1938
- POWer:RIPPle:RESult:MINimum:RMS? on page 1938
- POWer:RIPPle:RESult:MINimum:STDDev? on page 1938
- POWer:RIPPle:RESult:MINimum:WFMCount? on page 1938
- POWer:RIPPle:RESult:MINimum[:ACTual]? on page 1938
- POWer:RIPPle:RESult:NDCYcle:AVG? on page 1938
- POWer:RIPPle:RESult:NDCYcle:EVTCount? on page 1938
- POWer:RIPPle:RESult:NDCYcle:NPEak? on page 1938
- POWer:RIPPle:RESult:NDCYcle:PPEak? on page 1938
- POWer:RIPPle:RESult:NDCYcle:RMS? on page 1938
- POWer:RIPPle:RESult:NDCYcle:STDDev? on page 1939
- POWer:RIPPle:RESult:NDCYcle:WFMCount? on page 1939
- POWer:RIPPle:RESult:NDCYcle[:ACTual]? on page 1938
- POWer:RIPPle:RESult:PDCYcle:AVG? on page 1939
- POWer:RIPPle:RESult:PDCYcle:EVTCount? on page 1939
- POWer:RIPPle:RESult:PDCYcle:NPEak? on page 1939
- POWer:RIPPle:RESult:PDCYcle:PPEak? on page 1939
- POWer:RIPPle:RESult:PDCYcle:RMS? on page 1939
- POWer:RIPPle:RESult:PDCYcle:STDDev? on page 1939
- POWer:RIPPle:RESult:PDCYcle:WFMCount? on page 1939
- POWer:RIPPle:RESult:PDCYcle[:ACTual]? on page 1939
- POWer:RIPPle:RESult:PDEL:AVG? on page 1940
- POWer:RIPPle:RESult:PDEL:EVTCount? on page 1940
- POWer:RIPPle:RESult:PDEL:NPEak? on page 1940
- POWer:RIPPle:RESult:PDEL:PPEak? on page 1940
- POWer:RIPPle:RESult:PDEL:RMS? on page 1940
- POWer:RIPPle:RESult:PDEL:STDDev? on page 1940
- POWer:RIPPle:RESult:PDEL:WFMCount? on page 1940
- POWer:RIPPle:RESult:PDEL[:ACTual]? on page 1940
- POWer:RIPPle:RESult:PERiod:AVG? on page 1940
- POWer:RIPPle:RESult:PERiod:EVTCount? on page 1940
- POWer:RIPPle:RESult:PERiod:NPEak? on page 1940
- POWer:RIPPle:RESult:PERiod:PPEak? on page 1940
- POWer:RIPPle:RESult:PERiod:RMS? on page 1940
- POWer:RIPPle:RESult:PERiod:STDDev? on page 1940
- POWer:RIPPle:RESult:PERiod:WFMCount? on page 1940
- POWer:RIPPle:RESult:PERiod[:ACTual]? on page 1940

- POWer:RIPPle:RESult:STDDev:AVG? on page 1941
- POWer:RIPPle:RESult:STDDev:EVTCount? on page 1941
- POWer:RIPPle:RESult:STDDev:NPEak? on page 1941
- POWer:RIPPle:RESult:STDDev:PPEak? on page 1941
- POWer:RIPPle:RESult:STDDev:RMS? on page 1941
- POWer:RIPPle:RESult:STDDev:STDDev? on page 1941
- POWer:RIPPle:RESult:STDDev:WFMCount? on page 1941
- POWer:RIPPle:RESult:STDDev[:ACTual]? on page 1941
- POWer:RIPPle:REPort:ADD on page 1937

15.13.2 Configuring Output Ripple

For details of the configuration settings, see Chapter 15.13.3, "Output Ripple Settings", on page 982.

1. Select "Analysis" > "Power".
2. Under "Output", select "Ripple".
3. Connect the voltage probe to the oscilloscope.
4. If you want to measure the current ripple, enable the current channel .Connect the current probe to the oscilloscope.
5. If you want to measure both the voltage and the current ripple, deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
6. Connect the probes to the DUT as shown in the "Channels" tab:



7. Select the correct channels for the "Voltage Source" and the "Current Source".
8. Select the "Settings" tab.
9. Set the "SMPS switching frequency" according to your signal.
10. Select an "Optimize Scaling" option.

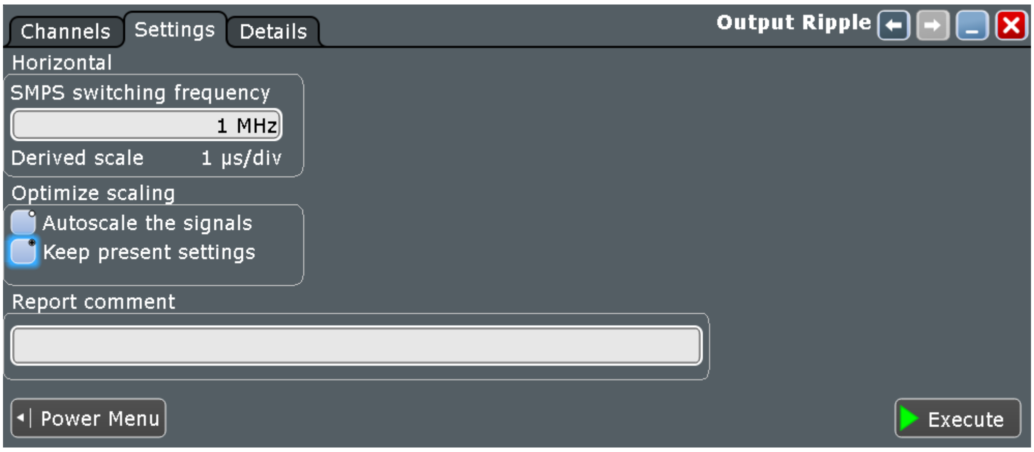
11. Tap "Execute".

    On the screen, you can see the measurement waveforms of the current and the voltage. Also, the result box with numeric measurement results is shown. For details, see Chapter 15.13.1, "Output Ripple Results", on page 978.

15.13.3 Output Ripple Settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933 and POWer:RIPPle:CURRent on page 1936)

In the "Settings" tab, you configure the ripple parameters and display settings.



Horizontal
Configures the horizontal scale of the result diagram.

SMPS switching frequency ← Horizontal
Sets the SMPS switching frequency. Set the value according to your DUT.

Remote command:
POWer:RIPPle:FREQuency on page 1936

Derived scale ← Horizontal
Shows the scale of the displayed results. This value is calculated as the inverse of the "SMPS switching frequency"

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:RIPPle:AUToscale` on page 1936

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Output Ripple" measurement.

Remote command:
`POWer:RIPPle:EXECute` on page 1936

15.14 Transient Response

The "Transient Response" analysis measures the response of a system to a change from equilibrium. This response is described by different properties like the rise time, the overshoot, the settling time, the peak time and the delay time.

Required probes:
- One or two voltage probes
- Current probe

15.14.1 Transient Response Results

The results of "Power Quality" measurements are provided in two ways:
- The diagram shows the graphical presentation of:
  – the voltage waveforms
  – the current waveform
- The result box displays the numeric measurement results.

To measure and display the transient response, the instrument uses the following measurements and waveforms:
- "P8" Meas 8 to measure the rise time, the overshoot and delay to trigger
- "C1" Cursor 3 to measure the peak time
- "C2" Cursor 2 to measure the delay to trigger

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The results describing the transient response of the system are shown in Figure 15-5 and described in Table 15-6.

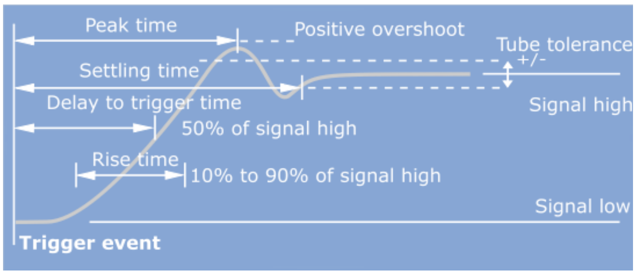


Figure 15-5: Graphical presentation of the transient response properties

Table 15-6: Transient response

Result	Description
Rise time	The time needed for the signal to change from 10% to 90% of the specified signal high.
Overshoot level	The maximum swing level above the signal high.
Settling time	The time elapsed from the trigger event to the time the output enters and remains within the tube tolerance band.
Peak time	The time needed for the response to reach the first peak of the overshoot.
Delay to trigger	The time needed for the response to reach half of the signal high value, after the trigger event.

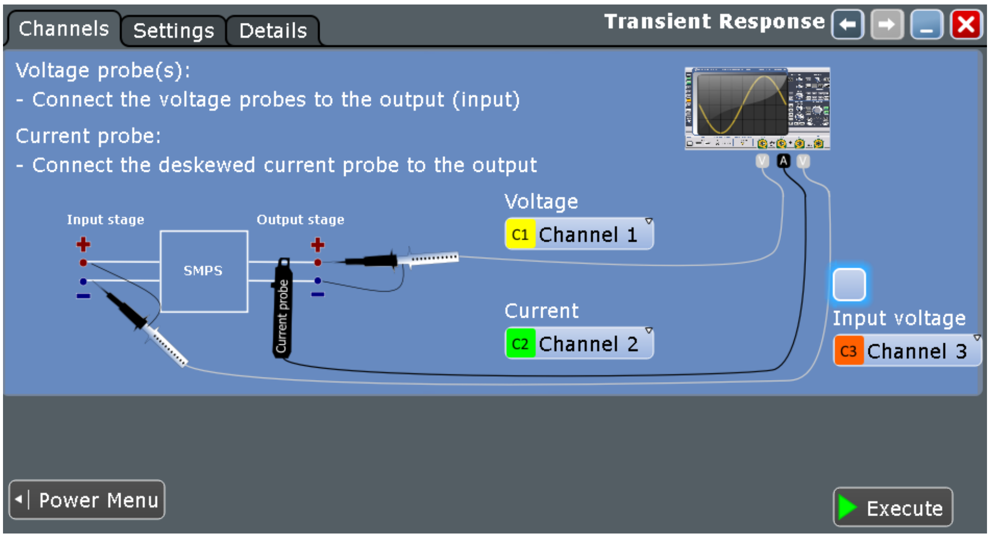
The following remote commands are used for handling the measurement results:

- `POWer:TRANsient:RESult[:ACTual]?` on page 1943
- `POWer:TRANsient:REPort:ADD` on page 1943

15.14.2 Configuring Transient Response

For details of the configuration settings, see Chapter 15.14.3, "Transient Response Settings", on page 985.

1. Select "Analysis" > "Power".
2. Under "Output", select "Transient Response".
3. Connect one or more voltage probes and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 15.1.1.1, "Auto Deskew", on page 923.
5. Connect the probes to the DUT as shown in the "Channels" tab:



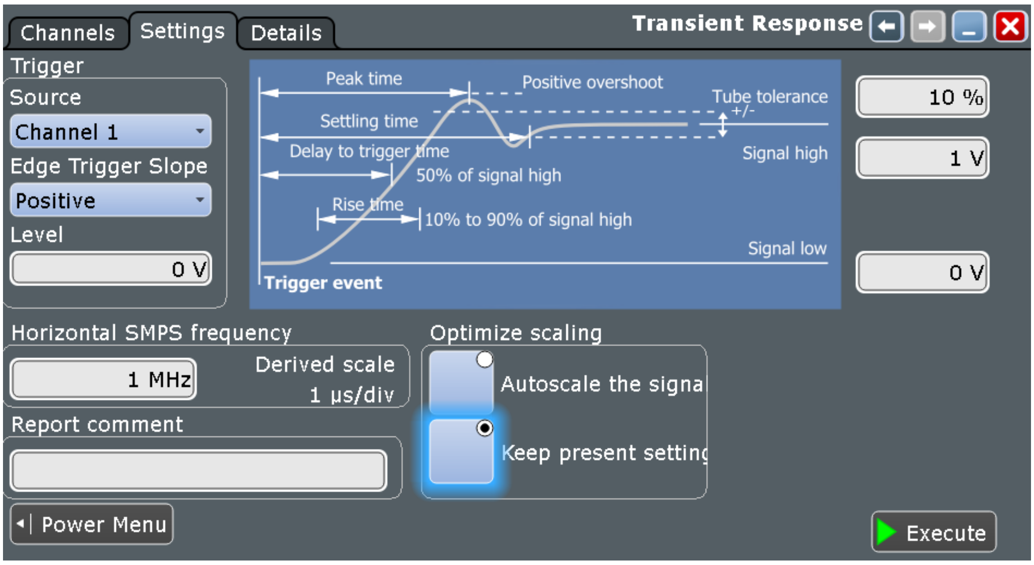
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Trigger" settings according to your signal.
9. Set the "Tube tolerance", "Signal high" and "Signal low" according to your requirements.
10. Set the "SMPS switching frequency" according to your device.
11. Select an "Optimize Scaling" option.
12. Tap "Execute".
13. If needed adjust the cursors manually. You can tap on a cursor and change its position with the [NAVIGATION] rotary knob.

    On the screen, you can see the measurement of the current and the voltage . Also, there is a table giving information about important measurement parameters. For details, see Chapter 15.14.1, "Transient Response Results", on page 983.

15.14.3 Transient Response Settings

In the "Channels" tab, you set the current source and the voltage sources, see also: Chapter 15.2.1, "Channels Tab", on page 933 and `POWer:TRANsient:INPut` on page 1942.

In the "Settings" tab, you configure the transient response measurement parameters and display settings.



Trigger

Sets the properties of the trigger.

Source ← Trigger

Sets the source channel of the trigger.

Remote command:

`POWer:TRANsient:TRGChannel` on page 1943

Edge Trigger Slope ← Trigger

Sets the edge type for the trigger event.

"Positive"	Selects the rising edge, that is a positive voltage change.
"Negative"	Selects the falling edge, that is a negative voltage change.
"Both"	Selects the rising as well as the falling edge.

Remote command:

`POWer:TRANsient:TRGSlope` on page 1944

Level ← Trigger

Sets the voltage or current level for the trigger event.

Remote command:

`POWer:TRANsient:TRGLevel` on page 1944

Tube tolerance

Specifies a tolerated error band for the signal level.

Remote command:

`POWer:TRANsient:HYSTeresis` on page 1942

Signal high

Sets the expected signal high-voltage value.

Remote command:
`POWer:TRANsient:SIGHigh` on page 1943

Signal low
Sets the expected signal low voltage value.

Remote command:
`POWer:TRANsient:SIGLow` on page 1943

Horizontal
Configures the horizontal scale of the result diagram.

SMPS switching frequency ← Horizontal
Sets the SMPS switching frequency. Set the value according to your DUT.

Remote command:
`POWer:TRANsient:FREQuency` on page 1942

Derived scale ← Horizontal
Shows the scale of the displayed results. This value is calculated as the inverse of the "SMPS switching frequency"

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:TRANsient:AUToscale` on page 1942

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Transient Response" measurement.

Remote command:
`POWer:TRANsient:EXECute` on page 1942

15.15 Output Spectrum

"Output Spectrum" analysis measures the spectrum of the output voltage. The results can be applied to see typical side effect problems of the SMPS application, such as switching frequency components of internal SMPS.

Required probes:

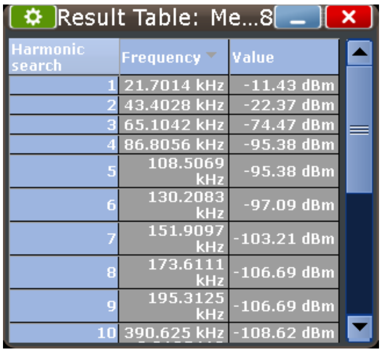
- Voltage probe

15.15.1 Output Spectrum Results

After executing the "Output Spectrum" measurement, the following windows are displayed:

The results of "Power Quality" measurements are provided in two ways:

- Two diagrams that show the graphical presentation of:
  - the voltage waveform
  - the spectrum
- The result box displays the positions of the measured peaks. The peaks are found by an automatic peak search.
  Note: If no results are found, check and correct the FFT settings.



Result Table: Me...8

Harmonic search	Frequency	Value
1	21.7014 kHz	-11.43 dBm
2	43.4028 kHz	-22.37 dBm
3	65.1042 kHz	-74.47 dBm
4	86.8056 kHz	-95.38 dBm
5	108.5069 kHz	-95.38 dBm
6	130.2083 kHz	-97.09 dBm
7	151.9097 kHz	-103.21 dBm
8	173.6111 kHz	-106.69 dBm
9	195.3125 kHz	-106.69 dBm
10	390.625 kHz	-108.62 dBm

To measure and display the output spectrum, the instrument uses the following measurements and waveforms:

- "M4" Math 4 to calculate the magnitude of the FFT for the voltage source values

The used resources are listed in the "Details" tab. See also: Chapter 15.2.3, "Details Tab", on page 935.

The measured peaks have different origin. Analyzing the frequencies gives information about the influences on the output signal.

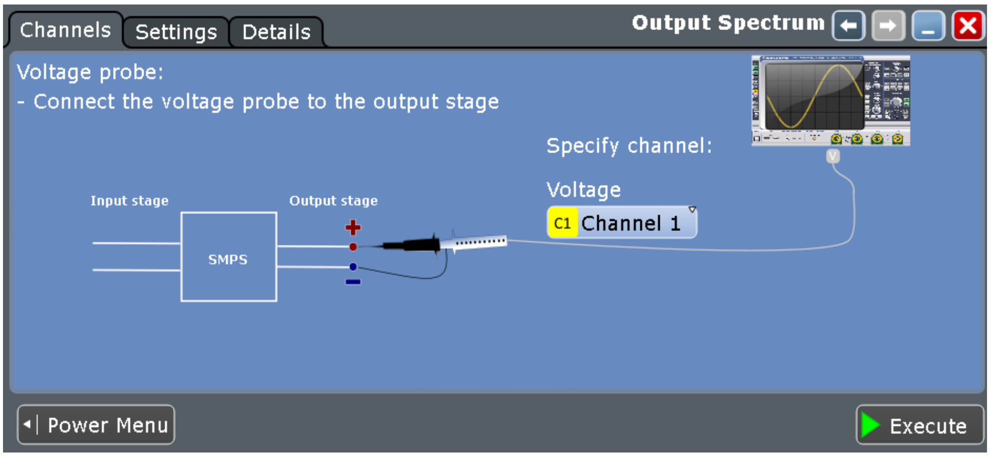
The following remote commands are used for handling the measurement results:

- `POWer:SPECtrum:RCOunt?` on page 1945
- `POWer:SPECtrum:RESult<m>:FREQuency?` on page 1945
- `POWer:SPECtrum:RESult<m>:LEVel?` on page 1945
- `POWer:SPECtrum:REPort:ADD` on page 1945

15.15.2 Configuring Output Spectrum

For details of the configuration settings, see Chapter 15.15.3, "Output Spectrum Settings", on page 989.

1. Select "Analysis">"Power".
2. Under "Output", select "Spectrum".
3. Select the "Channels" tab.
4. Connect the probe to the DUT and to the oscilloscope as shown in the graphic:



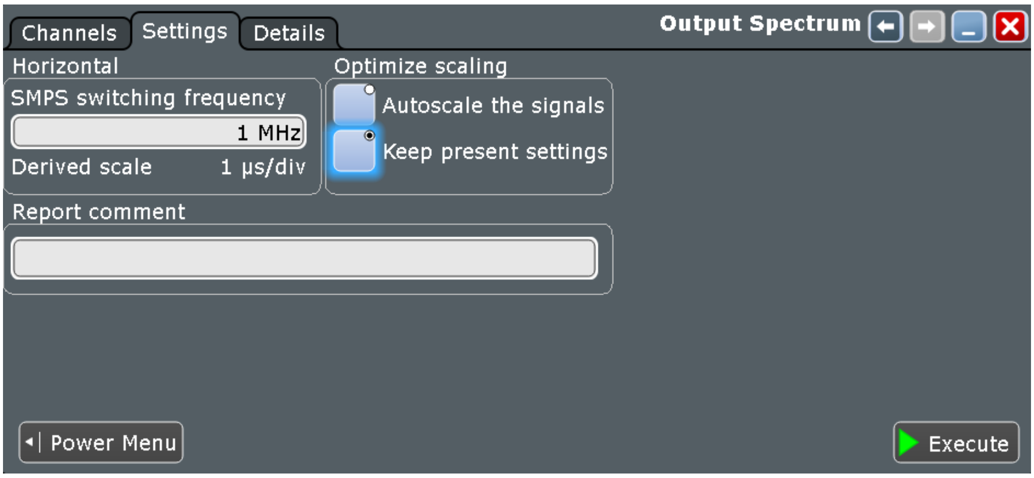
5. Select the correct channel for the "Voltage Source".
6. Select the "Settings" tab.
7. Set the "SMPS switching frequency" according to your signal.
8. Select an "Optimize Scaling"option.
9. Tap "Execute".
10. Set the positions of the cursors according to the measured spectrum. You can tap on a cursor and change its position with the [NAVIGATION] rotary knob.

    On the screen, you can see the measurement of the spectrum. Also, the result box shows the position of the peaks. For details, see Chapter 15.15.1, "Output Spectrum Results", on page 988.

15.15.3 Output Spectrum Settings

In the "Channels" tab, you set the voltage source, see also: Chapter 15.2.1, "Channels Tab", on page 933.

In the "Settings" tab, you configure the spectrum measurement parameters and display settings.



Horizontal

Configures the horizontal scale of the result diagram.

SMPS switching frequency ← Horizontal

Sets the SMPS switching frequency. Set the value according to your DUT.

Remote command:

`POWer:SPECtrum:FREQuency` on page 1944

Derived scale ← Horizontal

Shows the scale of the displayed results. This value is calculated as the inverse of the "SMPS switching frequency"

Optimize scaling

Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:

`POWer:SPECtrum:AUToscale` on page 1944

Report comment

In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute

Starts the "Output Spectrum" measurement.

Remote command:

`POWer:SPECtrum:EXECute` on page 1944

16 Network and Remote Operation

This chapter describes the usage of the embedded operating system on the instrument, the setup of network connections, and the interfaces and protocols used for remote control. It also explains how to start a remote control session.

This chapter contains the following sections:

- Operating System.. 991
- Setting Up a Network (LAN) Connection...996
- Web Interface..1001
- Remote Desktop Connection.. 1007
- Remote Control Interfaces and Protocols.. 1009
- Remote Settings...1013
- Starting and Stopping Remote Control.. 1015

16.1 Operating System

The R&S RTE has a Windows 10 operating system.

The operating system has been configured according to the instrument's features and needs. To ensure that the instrument software functions properly, certain rules must be observed when using the operating system.

NOTICE

Risk of rendering instrument unusable

The instrument is equipped with the Windows operating system. You can install additional software on the instrument, however, additional software can impair instrument function. Thus, run only programs that Rohde & Schwarz has tested for compatibility with the instrument software.

The drivers and programs used on the instrument under Windows are adapted to the instrument. Only install update software released by Rohde & Schwarz to modify existing instrument software.

Changes in the system setup are only required if the network configuration does not comply with the default settings (see Chapter 16.2.1, "Connecting the Instrument to the Network", on page 996).

16.1.1 Logon

Windows requires that users identify themselves by entering a user name and password in a logon window.

If the instrument is connected to the network, you are automatically logged on to the network when you log on to the operating system. As a prerequisite, the user name and the password must be identical under Windows and on the network. The instrument provides an auto-logon function that can be configured for standard user and administrator access. The configuration requires the user name and password. See also: "Log on as" on page 97

The R&S RTE provides two user accounts:

- "Instrument": an administrator account with unrestricted access to the computer/domain (default user)
- "NormalUser": a standard user account with limited access

For both users, the initial password is "894129". It is recommended that you change the password for both users after initial logon. You can change the password in the Windows configuration :"Start" menu > "Settings" > "Accounts" > "Sign-in options". Make sure to change the password also in the auto-logon of the R&S RTE, see below.

By default, the instrument logs on with administrator account and standard password. To restrict the access to system functions, you can change the autologon to "NormalUser".

The following tasks require administrator rights:

- Configuration of a LAN settings, network connection, and firewall settings
- Date and time setup
- Printer installation
- Firmware update
- BIOS update
- Using the web browser
- Touchscreen calibration
- Installation of options
- Moving portable licenses

To configure the auto-logon for a standard user

Default situation: the auto-logon is configured for the administrator ("Instrument").

1. Press the SETUP key and select the "System" tab.
2. Set "Logon as" to "User autologon".
3. Enter the "User name": *NormalUser*.
4. Enter the password of the standard user.
5. Restart the instrument.

To configure the auto-logon for administrator

Starting situation: the auto-logon is configured for a standard user ("NormalUser").

1. Press the SETUP key and select the "System" tab.
2. Set "Logon as" to "None".
3. Exit the firmware.
4. Sign out from the operating system. and

5. Log on as administrator ("Instrument").
6. Set the "Logon as" to "Admin autologon". Enter the "User name": *Instrument* and the administrator's password.
7. Restart the instrument.

16.1.2 Service Packs and Updates

Microsoft regularly creates security updates and other patches to protect Windows-based operating systems. These are released through the Microsoft Update website and associated update server. Instruments using Windows, especially those that connect to a network, should be updated regularly.

For details and recommendations, see the following Rohde & Schwarz white paper:

- 1DC01: Malware Protection Windows 7
- 1EF96: Malware Protection Windows 10

16.1.3 Virus Protection

Take appropriate steps to protect your instruments from infection. Use strong firewall settings and scan any removable storage device used with a Rohde & Schwarz instrument regularly. It is also recommended that you install anti-virus software on the instrument. Rohde & Schwarz does NOT recommend running anti-virus software in the background ("on-access" mode) on Windows-based instruments, due to potentially degrading instrument performance. However, Rohde & Schwarz does recommend running it during non-critical hours.

For details and recommendations, see the following Rohde & Schwarz white paper:

- 1DC01: Malware Protection Windows 7
- 1EF96: Malware Protection Windows 10

16.1.4 Backup and Recovery

The R&S RTE has a backup partition with an application for backup and restore the firmware. The backup of the factory default state is always available. You can save additional backups, for example, before a firmware update or to provide different system configurations for different environments.

The backup saves the current instrument installation and its configuration. Data are not saved in the backup.

To save a system backup

1. Before starting the backup, unplug the LAN cable and USB storage devices to minimize the risk of computer virus attacks.
2. Restart the R&S RTE.

The boot screen is displayed for about 2 seconds.

3. When the boot screen is displayed, immediately select "Backup", and press [ENTER].
4. Select "Create Backup".
5. Enter a name for the backup and the date. If necessary, you can add information to the description. By default, the description contains the current version of the firmware.
6. Select "Start Backup".

   A progress information dialog is displayed. You can terminate at any time with "Cancel". After the process has been finished, the dialog is closed automatically, and the main dialog is displayed again.
7. In the main dialog, select "Exit and Reboot".

   The backup and restore application is closed, and the R&S RTE is restarted.

To restore a backup version

When restoring a backup, the Windows operating system and installed anti-malware software are probably outdated. To minimize the risk of malware threats after restoring a backup, verify and adjust the "Windows Update" settings. Follow the recommendations from Rohde & Schwarz applicable to your instrument. Also, install all Windows security updates that have been published in the meanwhile.

1. Before starting the recovery, unplug the LAN cable and USB storage devices to minimize the risk of computer virus attacks.
2. Restart the R&S RTE.

   The boot screen is displayed for about 2 seconds.
3. When the boot screen is displayed, immediately select "Backup", and press [ENTER].
4. Select the backup that you want to restore.
5. Select "Restore Selected".
6. Confirm with "Yes".

   A progress information dialog is displayed. You can terminate at any time with "Cancel".

   During the restore process, the system partition is deleted, formatted and written newly. The data partition is not affected. After the process has been finished, the application is closed automatically, and the R&S RTE is restarted.

To delete a backup version

To provide space for new backups, you can remove older backups. The factory default cannot be deleted.

1. Restart the R&S RTE.
2. When the boot screen is displayed, immediately select "Backup", and press [ENTER].
3. Select the backup that you want to delete.
4. Select "Delete Selected".
5. Confirm with "Yes".
6. In the main dialog, select "Exit and Reboot", or continue with "To save a system backup" on page 993.

16.1.5 Accessing Windows Functionality

All required Windows settings can be changed using the touchscreen and the on-screen keyboard that is part of the Windows system. However, modification is easier if you connect a mouse and/or keyboard to the instrument.

To access Windows

► On the "File" menu, select "Minimize application".

The application is minimized to the taskbar and the "Start" menu becomes available.

To access Windows using an external keyboard

1. To open the "Start" menu, press the Windows key or the CTRL + ESC key combination on your keyboard.
2. To access the desktop, press the Windows key + D on your keyboard.

To access Windows settings directly from the firmware

Important Windows settings can be accessed directly from the R&S RTE interface.

1. Press the SETUP key and tap the "System" tab.
2. Select one of the settings buttons to access the corresponding Windows dialog box. Note that modification of most system settings requires administrator rights.

   Once you have opened a Windows dialog box, the taskbar and the "Start" menu are also available.

16.2 Setting Up a Network (LAN) Connection

Network environment

Before connecting the product to a local area network (LAN), consider the following:

- Install the latest firmware to reduce security risks.
- For internet or remote access, use secured connections if applicable.
- Ensure that the network settings comply with the security policies of your company. Contact your local system administrator or IT department before connecting your product to your company LAN.
- When connected to the LAN, the product may potentially be accessed from the internet, which may be a security risk. For example, attackers might misuse or damage the product.

Applications of LAN

The LAN connection settings can be configured directly in the Windows operating system. Provided the network administrator has assigned you the appropriate rights and adapted the Windows firewall configuration, you can use the interface, for example:

- To transfer data between a controlling device and the test device, e.g. to run a remote control program.
- To access or control the measurement from a remote computer using the "Remote Desktop" application (or a similar tool)
- To connect external network devices (e.g. printers)
- To transfer data from a remote computer and back, e.g. using network folders

This section describes how to configure the LAN interface. It includes the following topics:

- Chapter 16.2.1, "Connecting the Instrument to the Network", on page 996
- Chapter 16.2.2, "Assigning the IP Address", on page 997

Note that only user accounts with administrator rights can configure LAN networks.

LXI

The R&S RTE supports the LXI core features. LXI gives you direct access to the LAN settings described below.

16.2.1 Connecting the Instrument to the Network

There are two methods to establish a LAN connection to the instrument:

- A non-dedicated network (Ethernet) connection from the instrument to an existing network made with an ordinary RJ-45 network cable. The instrument is assigned an IP address and can coexist with a computer and with other hosts on the same network.

- A dedicated network connection (Point-to-point connection) between the instrument and a single computer made with a (crossover) RJ-45 network cable. The computer must be equipped with a network adapter and is directly connected to the instrument. The use of hubs, switches, or gateways is not required, however, data transfer is still performed using the TCP/IP protocol. You must assign an IP address to the instrument and the computer, see Chapter 16.2.2, "Assigning the IP Address", on page 997.

NOTICE

Risk of network failure

Consult your network administrator before performing the following tasks:

- Connecting the instrument to the network
- Configuring the network
- Changing IP addresses
- Exchanging hardware

Errors can affect the entire network.

► To establish a non-dedicated network connection, connect a commercial RJ-45 cable to one of the LAN ports.
To establish a dedicated connection, connect a (crossover) RJ-45 cable between the instrument and a single PC.

If the instrument is connected to the LAN, Windows automatically detects the network connection and activates the required drivers.

The network card can be operated with a 10/100/1000 Mbps Ethernet IEEE 802.3u interface.

16.2.2 Assigning the IP Address

Depending on the network capacities, the TCP/IP address information for the instrument can be obtained in different ways.

- If the network supports dynamic TCP/IP configuration using the Dynamic Host Configuration Protocol (DHCP), all address information can be assigned automatically.
- If the network does not support DHCP, or if the instrument is set to use alternate TCP/IP configuration, the addresses must be set manually.

By default, the instrument is configured to use dynamic TCP/IP configuration and obtain all address information automatically. This means that it is safe to establish a physical connection to the LAN without any previous instrument configuration.

NOTICE

Risk of network errors

Connection errors can affect the entire network. If your network does not support DHCP, or if you choose to disable dynamic TCP/IP configuration, you must assign valid address information before connecting the instrument to the LAN. Contact your network administrator to obtain a valid IP address.

Assigning the IP address on the instrument

1. Press the [SETUP] key.
2. Select the "System" tab.
3. Tap "Network".
4. Select "Ethernet".
5. Select "Properties".
6. On the "Networking" tab, select "Internet Protocol Version 4 (TCP/IPv4)", and then select "Properties".
7. Select "Use the following IP address".
8. Enter the address information as obtained from the network administrator.

16.2.3 Using Computer Names

In a LAN that uses a DNS server (Domain Name System server), each PC or instrument connected in the LAN can be accessed via an unambiguous computer name instead of the IP address. The DNS server translates the host name to the IP address. This is especially useful when a DHCP server is used, as a new IP address may be assigned each time the instrument is restarted.

Each instrument is delivered with an assigned computer name, but this name can be changed.

The default instrument name is a non-case-sensitive string with the following syntax:

<Type><variant>-<serial_number>

The serial number can be found on the rear panel of the instrument. It is the third part of the device ID printed on the bar code sticker:

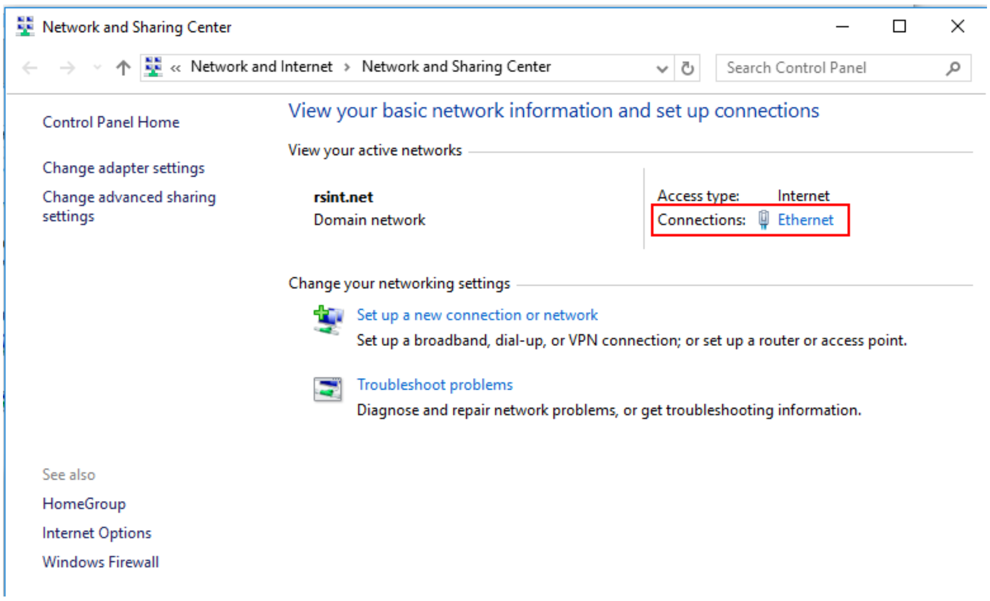
To change the computer name

1. Press the [SETUP] key and select the "System" tab.
   The current computer name is displayed and can be edited.
2. Alternatively, tap "System" on the "System" tab.
3. Select "Change", enter the new computer name and confirm the entry.

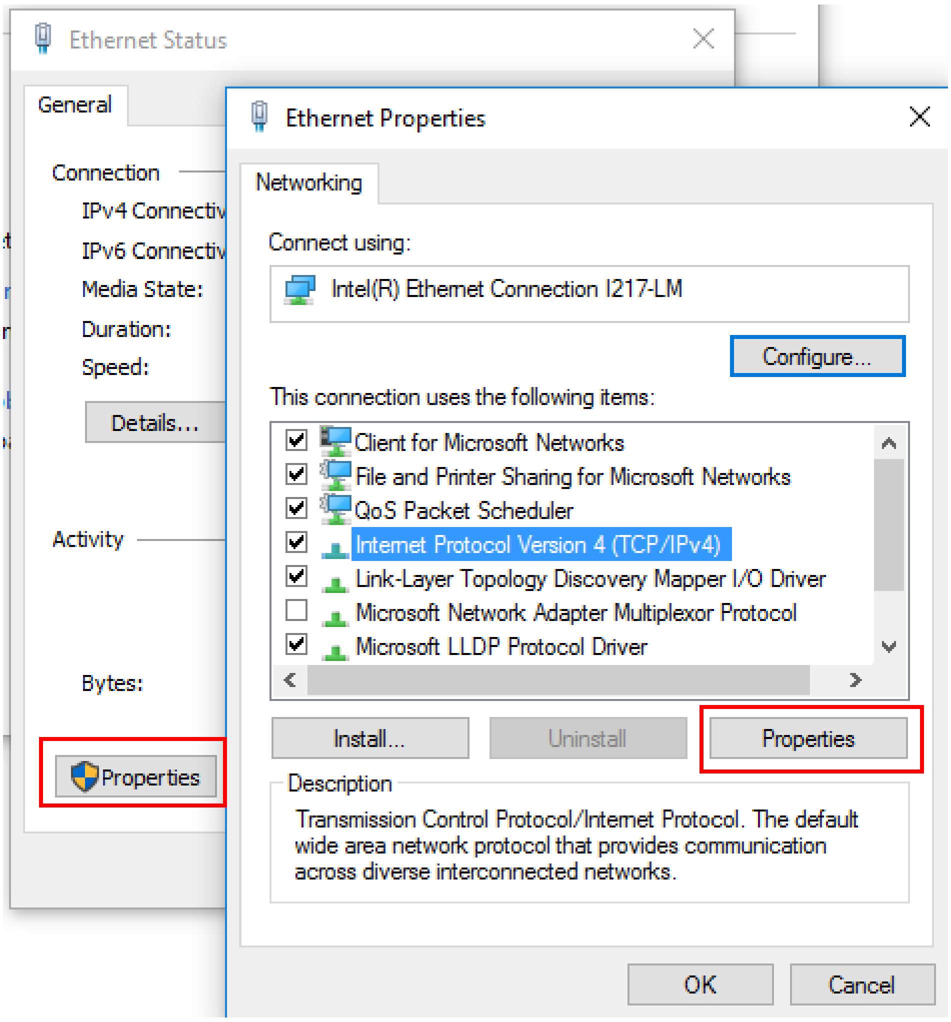
Using a DNS server to determine the IP address

If a DNS server is configured on the R&S RTE, the server can determine the current IP address for the connection using the permanent computer name.

1. Obtain the name of your DNS domain and the IP addresses of the DNS and WINS servers on your network.
2. Press the [SETUP] key.
3. Select the "System" tab.
4. Tap "Network".
5. Select "Ethernet".



6. On the "Networking" tab, select "Internet Protocol Version 4 (TCP/IPv4)", and then select "Properties".



7. On the "General" tab, select "Use the following DNS server addresses" and enter your own DNS addresses.

   For more information, refer to the Windows operating system help.

16.2.4 Changing the Windows Firewall Settings

A firewall protects an instrument by preventing unauthorized users from gaining access to it through a network. Rohde & Schwarz highly recommends the use of the firewall on your instrument. Rohde & Schwarz instruments are shipped with the Windows firewall enabled and preconfigured in such a way that all ports and connections for remote control are enabled.

For more details on firewall configuration, see the following Rohde & Schwarz White Paper:

- 1DC01: Malware Protection Windows 7
- 1EF96: Malware Protection Windows 10

Note that changing firewall settings requires administrator rights.

16.3 Web Interface

If the R&S RTE is connected to a computer via LAN, you can operate the instrument from a computer. No additional tools are required, you need only a web browser.

The R&S RTE supports the LXI core features. LAN extension for instrumentation (LXI) is an instrumentation platform for measuring instruments and test systems that is based on standard Ethernet technology. LXI is intended to be the LAN-based successor to GPIB, combining the advantages of Ethernet with the simplicity and familiarity of GPIB. For information about the LXI standard, refer to the LXI website at http://www.lxistandard.org.

16.3.1 Settings on the R&S RTE

Access: "File" menu > "Setup" >"LAN" tab.

The LAN status icon on the toolbar of the R&S RTE indicates the status of the LAN connection. A green icon indicates that the instrument is connected to the LAN; a red symbol indicates an error - mostly the LAN cable is not connected.

The "LAN" tab of the "Setup" dialog box provides network information.

Only users with administrator rights can change LAN settings.

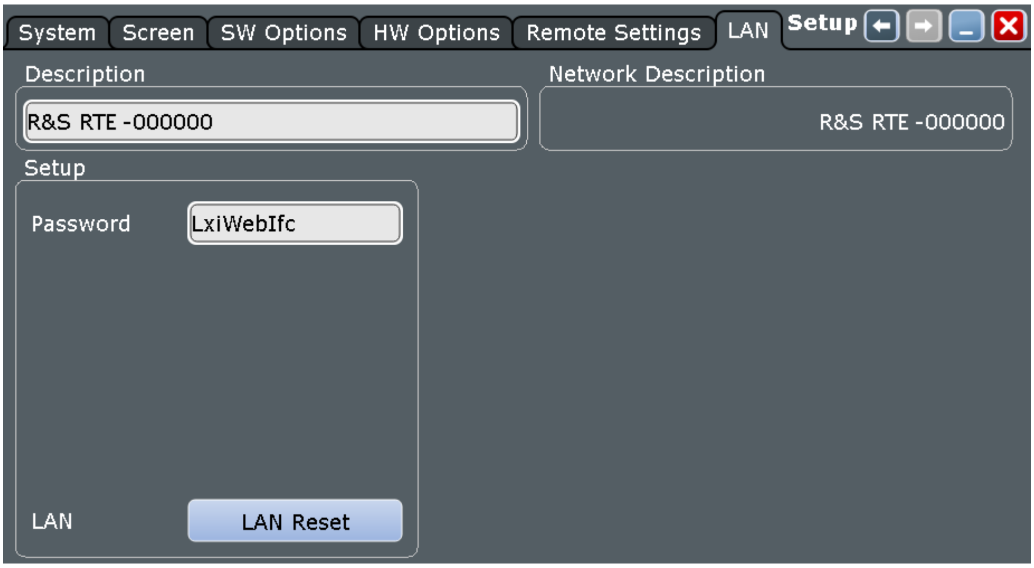


Figure 16-1: LAN settings with Windows 10

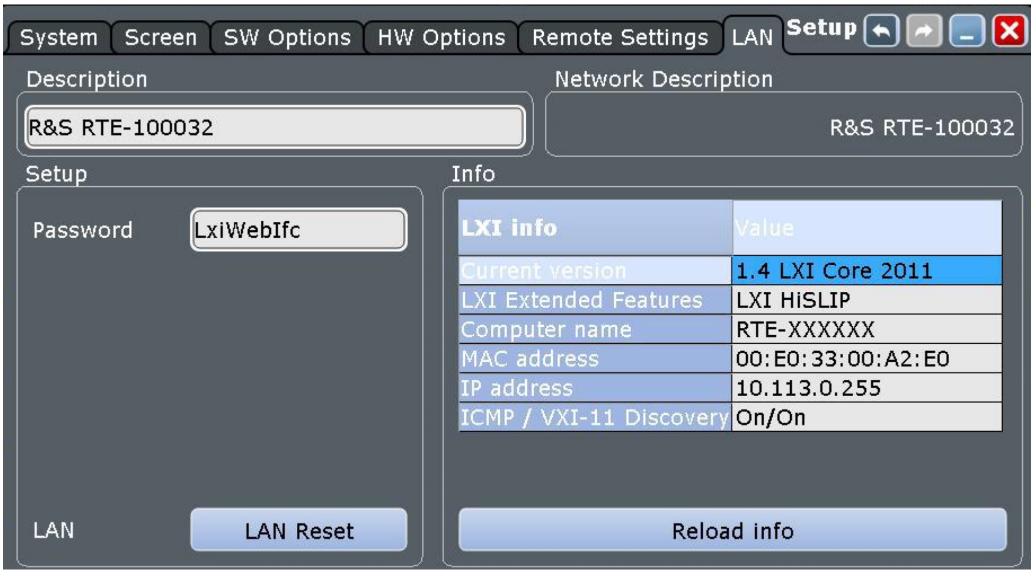


Figure 16-2: LAN settings with Windows 7

Description

Instrument description of the R&S RTE.

Password

Password for LAN configuration. The default password is *LxiWebIfc*.

LAN Reset

Resets the LAN configuration to its default settings using the network configuration reset mechanism (LCI) for the instrument. The following parameters are reset:

Parameter	Value
TCP/IP mode	DHCP + auto IP address
Dynamic DNS	Enabled
ICMP ping	Enabled
Password for LAN configuration	LxiWebIfc

The LAN settings are configured using the instrument's web browser.

LXI Info

Displays the current LXI information from the R&S RTE. Only available with Windows 7.

"Current version"
: Current LXI version

"LXI Extended Features"
: List of extended LXI features that the instrument supports

"Computer name"
: Name of the R&S RTE as defined in the operating system

"MAC address" Media Access Control address (MAC address), a unique identifier for the network card in the R&S RTE

"IP address" IP address of the R&S RTE as defined in the operating system.

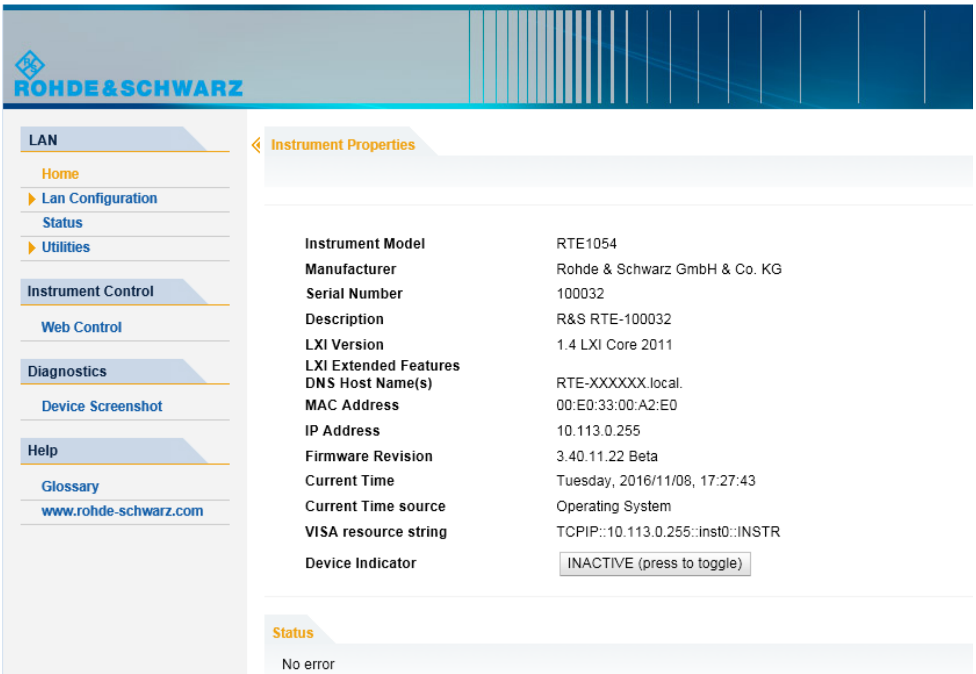
Reload Info
Reloads LXI configuration. Only available with Windows 7.

16.3.2 Web Browser

The instrument's web interface works with all W3C compliant browsers. Only users with administrator rights on the instrument can use the web interface.

1. Open a web browser on the computer.
2. Type the instrument's host name or IP address in the address field of the browser on your PC, e.g. "http://10.113.10.203".

   The instrument home page opens.



The instrument home page displays the device information required by the LAN standard including the VISA resource string in read-only format.

► To update the "Host Name", press the "Device Indicator" button. The "Device Indicator" is not password-protected.

The most important items in the navigation menu of the browser interface are the following :

- "LAN Configuration" opens the menu with configuration pages.

- "Status" displays information about the status of the instrument.
- "Utilities > Logging" shows log messages and provides buttons to download or clear the logfile.
- "Web Control" emulates the front panel and shows the instrument display. You see a live image of the instrument, and you can operate the instrument remotely. You can use the keys, the knobs and the menus in the same way as directly on the instrument.
- "Device Screenshot": creates a screenshot of the display.
- "Help" provides a glossary of terms related to the standard, and a link to the Rohde & Schwarz Internet site.

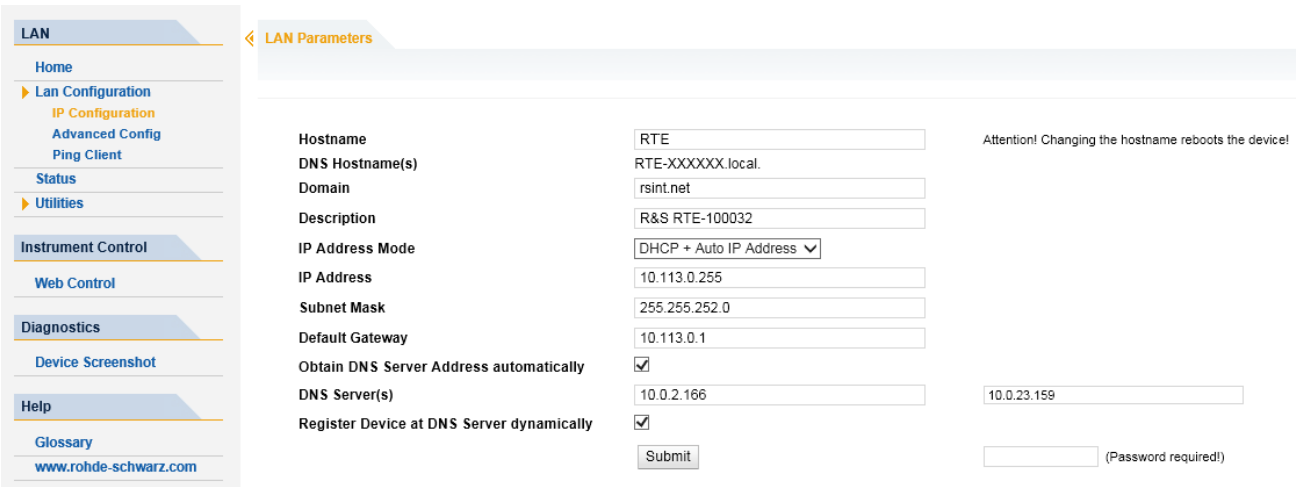
16.3.2.1 LAN Configuration

The LAN configuration consists of three parts:

- "IP configuration" provides all mandatory LAN parameters.
- "Advanced LAN Configuration" provides LAN settings that are not declared mandatory by the standard.
- "Ping Client" provides the ping utility to verify the connection between the instrument and other devices.

IP Configuration

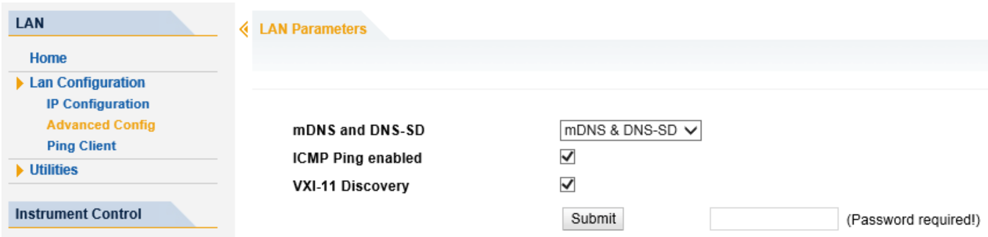
The "LAN Configuration > IP configuration" web page displays all mandatory LAN parameters and allows their modification.



The "TCP/IP Mode" configuration field controls how the IP address for the instrument gets assigned (see also Chapter 16.2.2, "Assigning the IP Address", on page 997). For the manual configuration mode, the static IP address, subnet mask, and default gateway are used to configure the LAN. The automatic configuration mode uses DHCP server or Dynamic Link Local Addressing (automatic IP) to obtain the instrument IP address.

Changing the LAN configuration is password-protected. The password is *LxiWebIfc* (notice upper and lower case characters). This password cannot be changed in the current firmware version.

Advanced Config



The "LAN Configuration > Advanced Config" parameters are used as follows:

- mDNS and DNS-SD are two additional protocols: Multicast DNS and DNS Service Discovery. They are used for device communication in zero configuration networks working without DNS and DHCP
- "ICMP Ping" must be enabled to use the ping utility.
- "VXI-11" is the protocol that is used to detect the instrument in the LAN. According to the standard, LXI devices must use VXI-11 to provide a detection mechanism; other additional detection mechanisms are permitted.

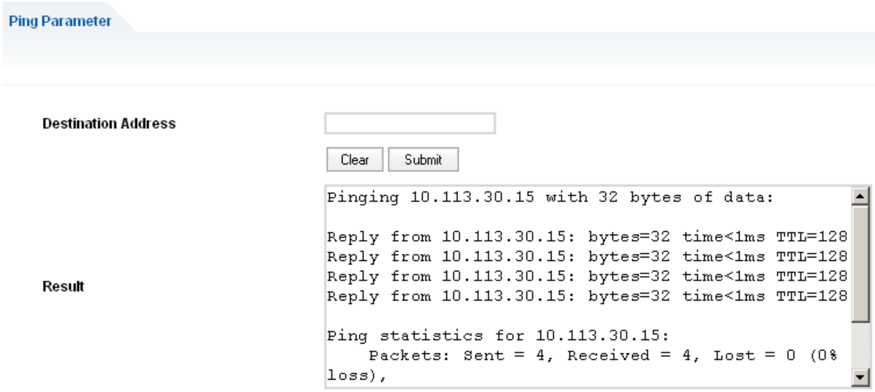
Ping Client

Ping is a utility that verifies the connection between the instrument and another device. The ping command uses the `ICMP` echo request and echo reply packets to determine whether the LAN connection is functional. Ping is useful for diagnosing IP network or router failures. The ping utility is not password-protected.

To initiate a ping between the compliant instrument and a second connected device:

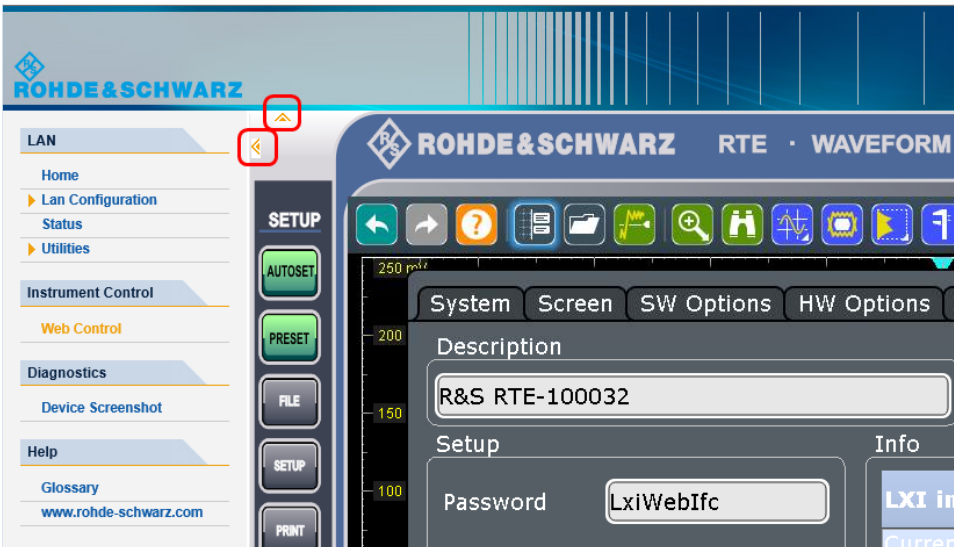
1. Enable "ICMP Ping" on the "Advanced Config" page (enabled by default).
2. On the "Ping Client" page, enter the IP address of the second device **without the ping command and without any further parameters** into the "Destination Address" field (e.g. *10.113.10.203*).

3. Click "Submit".



16.3.2.2 Web Control

"Web Control" emulates the front panel and shows the instrument display. You see a live image of the instrument, and you can operate the instrument remotely. You can use the keys, the knobs and the menus in the same way as directly on the instrument. The Web control replaces VNC as control tool for remote operation.

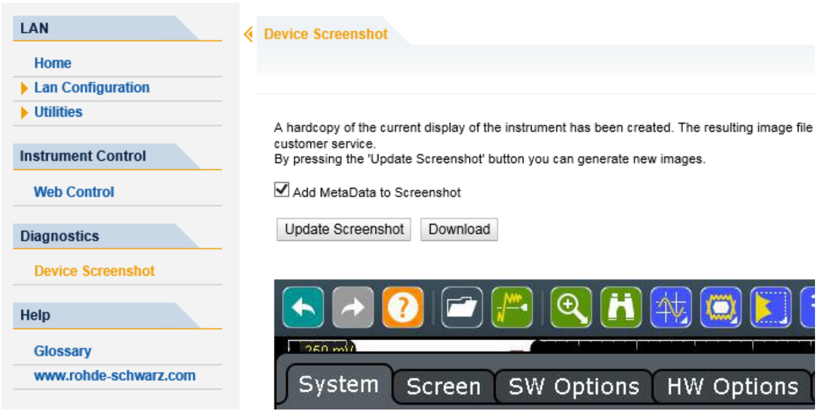


In the upper left corner of the emulated display, you see two arrows:

- The up/down arrow hides or shows the front panel.
- The left/right arrow hides or shows the menu.

16.3.2.3 Device Screenshot

If you click "Device Screenshot", a screenshot of the current instrument display is shown on the computer.



Add MetaData to Screenshot
Adds the instrument information to PNG and JPG files. Meta information is saved as EXIF information and can be read, for example, using the ExifTool, see "Meta information in screenshots" on page 462.

Update Screenshot
Updates the display.

Download
Saves the screenshot to the download directory of the computer. By default, JPG format is saved.

To save the screenshot in PNG format, select "Save as", select "All files" as type, and enter the filename with extension *.png*.

16.4 Remote Desktop Connection

Remote Desktop is a Windows application, which can be used to access and control the instrument from a remote computer through a LAN connection. While the instrument is in operation, the contents of the instrument screen are displayed on the remote computer. Remote Desktop provides access to all applications, files, and network resources of the instrument. Thus, remote operation of the instrument is possible.

NOTICE

Risk of unauthorized access

If you enable the Windows Remote Desktop application on the instrument, anyone using the network who knows the computer name and login data can access it. To prevent unauthorized access, make sure that the Remote Desktop application on the instrument is disabled: "Start" > "Settings" > "Control Panel" > "System"

To set up a Remote Desktop connection

1. Enable remote desktop control on the instrument. See "Enabling remote desktop control on the instrument" on page 1008 for details.

2. Connect the instrument and the remote computer to a LAN. See Chapter 16.2.1, "Connecting the Instrument to the Network", on page 996 for details.
3. Set up the Remote Desktop connection between the remote computer and the instrument.

Remote Desktop Connection is part of the operating system and can be accessed via "Start > Windows Accessories > Remote Desktop Connection.".

Enabling remote desktop control on the instrument

1. Press the [Setup] key and select the "System" tab.
2. Tap "System".
3. Select "Remote settings".

   The "Remote" tab of the "System Properties" is shown.
4. Under "Remote Desktop", activate "Allow remote connections to this computer".

   Note: Remote Desktop access and firewall settings.
   When you enable or disable the Windows Remote Desktop option (in the "System Properties"), the associated firewall settings are adapted automatically.
5. If necessary, click "Select Users" and select users who are allowed to access the R&S RTE via Remote Desktop.

   The user account under which configuration is carried out is automatically enabled for Remote Desktop.

Setting up the Remote Desktop connection on the remote computer

1. On the remote computer, select "Start > Windows Accessories > Remote Desktop Connection".
2. Enter the instrument's name or IP address in the dialog box.
   See also Chapter 16.2.2, "Assigning the IP Address", on page 997.
3. Enter the user ID and password for the instrument. See Chapter 16.1, "Operating System", on page 991 for details.
4. Click "Connect".

   When the connection has been set up, the instrument's screen appears on the remote computer.

For detailed information about Remote Desktop and the connection, refer to the Windows Help.

Helpful settings for Remote Desktop

The following settings for the Remote Desktop connection can make working on the remote PC more convenient.

1. When setting up the connection to the instrument, you can configure the connection settings in the "Remote Desktop Connection" dialog box.
   Click the "Show Options".

   The dialog box is expanded to display the configuration data.

2. Customize the settings:
   a) On the "Experience" tab, select the appropriate connection to optimize the connection speed.
   b) On the "Local Resources" tab:
      - To use printers connected to the remote PC while accessing them from the instrument, activate "Printers" under "Local devices and resources".
      - If you want access drives of the remote PC from the instrument, e.g. to store settings or to copy files from the PC to the instrument:
        – Select "More" under "Local devices and resources".
        – Select the "Drives" that are needed.
          Windows maps the selected drives of the remote PC to corresponding network drives. When a connection is established, a warning on the PC indicates that the drives are enabled for access from the instrument.
   c) On the "Display" tab:
      - Use the slider to set the size of the R&S RTE window on the remote PC desktop.
      - Activate "Display the connection bar when in full screen mode".
        A bar with the network address of the instrument is displayed on the screen, which you can use to reduce, minimize or close the window.
   d) On the "General" tab, you can save the connection settings for later use: click "Save As".

Terminating Remote Desktop Control

A Remote Desktop connection can be terminated either on the R&S RTE or on the remote PC. The connection can be established again any time as long as remote control is enabled on the instrument. Consider the notice "Risk of unauthorized access" on page 1007.

► To terminate the connection on the remote PC, close the "Remote Desktop" window, or select "Start > Disconnect".

16.5 Remote Control Interfaces and Protocols

The instrument supports different interfaces for remote control. The following table gives an overview.

Table 16-1: Remote control interfaces and protocols

Interface	Protocols, VISA address string	Remarks

Local Area Network (LAN)	**Protocol HiSLIP**
	VISA address string:
	`TCPIP::<host address>::hislip0[,<port>][::INSTR]`
	Protocol VXI-11
	VISA address string:
	`TCPIP::<host address>[::inst0]::[INSTR]`
		The LAN connector is located on rear panel of the instrument.
		The interface is based on TCP/IP and supports various protocols.
		See also:
		• Chapter 16.5.2.2, "VXI-11 Protocol", on page 1012
		• Chapter 16.5.2.3, "HiSLIP Protocol", on page 1012
		• Chapter 16.5.1, "VISA Libraries", on page 1010

GPIB (IEC/IEEE Bus Interface)	VISA address string: GPIB::primary address[::INSTR] (no secondary address)	The optional GPIB bus interface according to standard IEC 625.1/IEEE 488.1 is located on the rear panel of the instrument. See also: Chapter 16.5.3, "GPIB Interface (IEC/IEEE Bus Interface)", on page 1012.

Within this interface description, the term GPIB is used as a synonym for the IEC/IEEE bus interface.

SCPI (Standard Commands for Programmable Instruments)

SCPI commands - messages - are used for remote control. Commands that are not taken from the SCPI standard follow the SCPI syntax rules. The instrument supports the SCPI version 1999. The SCPI standard is based on standard IEEE 488.2 and aims at the standardization of device-specific commands, error handling and the status registers. The tutorial "Automatic Measurement Control - A tutorial on SCPI and IEEE 488.2" from John M. Pieper (R&S order number 0002.3536.00) offers detailed information on concepts and definitions of SCPI.

16.5.1 VISA Libraries

VISA is a standardized software interface library providing input and output functions to communicate with instruments. Instrument access via VXI11 protocol is usually achieved from high level programming platforms using VISA as an intermediate abstraction layer. VISA encapsulates the low level VXI or even GPIB function calls and thus makes the transport interface transparent for the user.

The I/O channel (LAN or TCP/IP, USB, GPIB,...) is selected at initialization time by the channel–specific address string ("VISA resource string") indicated in Table 16-1, or by an appropriately defined VISA alias (short name). A VISA installation is a prerequisite for remote control of R&S RTE.

For more information about VISA, refer to the VISA user documentation.

16.5.2 LAN Interface

To be integrated in a LAN, the instrument is equipped with a LAN interface, consisting of a connector, a network interface card and protocols. For remote control via a network, the PC and the instrument must be connected via the LAN interface to a common network with TCP/IP network protocol. They are connected using a commercial RJ45 cable. The TCP/IP network protocol and the associated network services are preconfigured on the instrument. Software for instrument control and the VISA program library for specified protocols must be installed on the controller.

16.5.2.1 IP Address

Only the address of the instrument is required to set up the connection. It is part of the "VISA resource string" used by programs to identify and control the instrument. The VISA resource string has the form:

TCPIP::<host address>::hislip0[,<port>][::INSTR] for HiSLIP protocol

TCPIP::<host address>[::inst0][::INSTR] for VXI-11 protocol

Where:

- host address identifies the instrument in the network, usually the IP address. If the LAN is supported by a DNS server, the host name can be used instead of the IP address. The DNS server (Domain Name System server) translates the host name to the IP address.
- hislip0 indicates the HiSLIP protocol
- inst0 is the default LAN device name. VISA supports several devices running on the instrument. On R&S RTE, only one device is configured, so the LAN device name can be omitted.
- INSTR specifies a VISA resource of the type INSTR. By default, the VISA resource name control is set to the INSTR class.

Example: HiSLIP

IP address is *192.1.2.3*: the valid resource string is: TCPIP::192.1.2.3::hislip0

Instrument name is *RSRT1*: the valid resource string is: TCPIP::RSRT1::hislip0.

DNS host name is *RTE-123456*: the valid resource string is: TCPIP::RTE-123456::hislip0.

Example: VXI-11

IP address is *192.1.2.3*: the valid resource string is: TCPIP::192.1.2.3

Instrument name is *RSRT1*: the valid resource string is: TCPIP::RSRT1.

DNS host name is *RTE-123456*: the valid resource string is: TCPIP::RTE-123456.

See also:

- Find IP address: [SETUP] > "System" tab, see "System" on page 96
- Chapter 16.2.2, "Assigning the IP Address", on page 997

16.5.2.2 VXI-11 Protocol

The VXI-11 standard is based on the ONC RPC (Open Network Computing Remote Procedure Call) protocol which in turn relies on TCP/IP as the network/transport layer. The TCP/IP network protocol and the associated network services are preconfigured. TCP/IP ensures connection-oriented communication, where the order of the exchanged messages is adhered to and interrupted links are identified. With this protocol, messages cannot be lost.

16.5.2.3 HiSLIP Protocol

The HiSLIP (**Hi**gh **S**peed **L**AN **I**nstrument **P**rotocol) is the successor protocol for VXI-11 for TCP-based instruments specified by the IVI foundation. The protocol uses two TCP sockets for a single connection - one for fast data transfer, the other for non-sequential control commands (e.g. `Device Clear` or `SRQ`).

HiSLIP has the following characteristics:

- High performance as with raw socket network connections
- Compatible IEEE 488.2 support for Message Exchange Protocol, Device Clear, Serial Poll, Remote/Local, Trigger, and Service Request
- Uses a single IANA registered port (4880), which simplifies the configuration of firewalls
- Supports simultaneous access of multiple users by providing versatile locking mechanisms
- Usable for IPv6 or IPv4 networks

Using VXI-11, each operation is blocked until a VXI-11 device handshake returns. However, using HiSLIP, data is sent to the device using the "fire and forget" method with immediate return. Thus, a successful return of a VISA operation such as `viWrite()` does not guarantee that the instrument has finished or started the requested command, but is delivered to the TCP/IP buffers.

For more information see also the application note:

1MA208: Fast Remote Instrument Control with HiSLIP

16.5.3 GPIB Interface (IEC/IEEE Bus Interface)

To be able to control the instrument via the GPIB bus, the instrument and the controller must be linked by a GPIB bus cable. A GPIB bus card, the card drivers and the program libraries for the programming language used must be provided in the controller. The controller must address the instrument with the GPIB bus address.

Characteristics

- Up to 15 instruments can be connected

- The total cable length is restricted to a maximum of 15 m; the cable length between two instruments should not exceed 2m.
- A wired "OR"-connection is used if several instruments are connected in parallel.

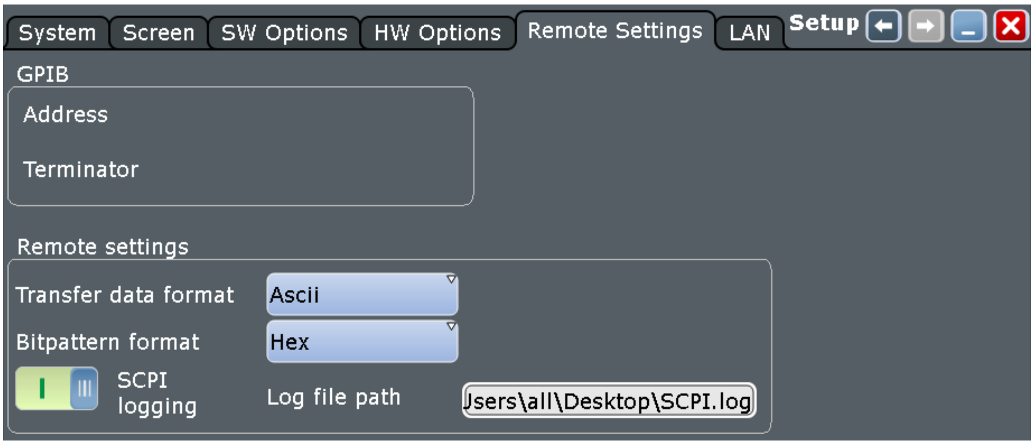
16.5.3.1 GPIB Instrument Address

To operate the instrument via remote control, it must be addressed using the GPIB address. The remote control address is factory-set to 20, but it can be changed in the network environment settings. For remote control, addresses 0 through 30 are allowed. The GPIB address is maintained after a reset of the instrument settings.

See also: "Address" on page 1013.

16.6 Remote Settings

Access: "File" menu > "Setup" >"Remote Settings" tab.



The settings on this tab are required for remote control of the instrument via a connected computer.

Address
Indicates the GPIB address of the instrument if an optional GPIB bus card is installed.

The address can be edited here. Be aware that changing the address has major effects on the communication to the remote computer. See also: Chapter 16.5.3, "GPIB Interface (IEC/IEEE Bus Interface)", on page 1012.

Remote command:
`GPIB:ADDRess` on page 1047

Terminator
Specifies the symbol that is used as a terminator in GPIB communication.

Remote command:
`GPIB:TERMinator` on page 1047

Transfer data format
Selects the data format that is used for transmission of waveform data from the instrument to the controlling computer.

Waveform data can be retrieved using the following commands:
- `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?`
- `CALCulate:MATH<m>:DATA[:VALues]?`
- `REFCurve<m>:DATA[:VALues]?`
- `DIGital<m>:DATA[:VALues]?`

The content of the data stream can be defined with [FILE] > "Save/Recall > Waveforms > Interleaved X/Y" (or `EXPort:WAVeform:INCXvalues`).

"Ascii"	Data values are returned in ASCII format as a list of comma separated values in floating point format.
"FLOAT"	Binary format. The data is stored as binary data (Definite Length Block Data according to IEEE 488.2).
"INT8"	Signed integer data with length 8 bit.
"INT16"	Signed integer data with length 16 bit. The Byte order can be set using . For details on the formats, refer to the description of the remote command.

Remote command:
`FORMat[:DATA]` on page 1045

Byte order
Sets the endianess for INT16 data:
- LSB first: little endian, least significant byte first
- MSB first: big endian, most significant byte first

Remote command:
`FORMat:BORDer` on page 1046

Bit pattern format
Sets the format for all bit pattern queries.

Remote command:
`FORMat:BPATtern` on page 1046

SCPI logging
If enabled, all received remote commands are written into a text file. Enter the path and filename with extension (log, txt, or csv) in "Log file path".

SCPI emulation
Not available for R&S RTE

16.7 Starting and Stopping Remote Control

16.7.1 Starting a Remote Control Session

When you switch on the instrument, it is always in manual operation state ("local" state). It can be operated via the front panel, the touch screen and external keyboard and/or mouse.

► To start remote control:
- Send a command from the controller.
- VXI-11 protocol (LAN interface): Use `&GTR` interface message.

While remote control is active, the instrument settings are optimized for maximum measurement speed; the display is switched off. Operation via the front panel is disabled.

On the touch screen, two buttons appear in the upper left corner: "Local" and "View".

16.7.2 Using the Display during Remote Control

You can observe the screen while a remote control script is executed. This is helpful for program test purposes but tends to slow down the measurement. Therefore it is recommended that you switch off the display in real measurement applications where a tested program script is to be executed repeatedly.

► To switch on the display, do one of the following:
- Tap the "View" button in the upper left corner of the touch screen.
- Use the `SYSTem:DISPlay:UPDate ON` command.

► To switch off the display, do one of the following:
- Tap the "View" button again.
- Use the `SYSTem:DISPlay:UPDate OFF` command.

16.7.3 Returning to Manual Operation

The instrument switches back to manual operation when the remote connection is closed. Besides, you can return to manual operation manually or via remote control.

► To return to manual operation:
- Tap the "Local" button in the upper left corner of the touch screen.
- VXI-11 protocol: Use `&GTL` interface message.

17 Remote Control Commands

This chapter describes all remote commands available for R&S RTE and provides examples and information how to use the commands.

Further information on remote control:

- Chapter 16.5, "Remote Control Interfaces and Protocols", on page 1009
- Chapter 16.7, "Starting and Stopping Remote Control", on page 1015
- Chapter B, "Remote Control - Basics", on page 1969
- Chapter C, "Remote Control - Status Reporting System", on page 1983

- Conventions used in Remote Command Description....................1016
- Finding the Appropriate Command....................1017
- Programming Examples....................1018
- Frequently Used Parameters and Suffixes....................1035
- Common Commands....................1039
- General Remote Settings....................1044
- Instrument Setup....................1048
- Acquisition and Setup....................1069
- Trigger....................1127
- Waveform Analysis....................1173
- Cursor Measurements....................1201
- Automatic Measurements....................1212
- Spectrum Analysis....................1278
- Mask Testing....................1294
- Search....................1314
- Data Management....................1352
- Protocols....................1381
- Mixed Signal Option (MSO, R&S RTE-B1)....................1863
- Waveform Generator (Option R&S RTE-B6)....................1886
- Power Analysis (Option R&S RTE-K31)....................1906
- Maintenance....................1945
- Status Reporting....................1948
- Remote Trace....................1952
- Deprecated Commands....................1956

17.1 Conventions used in Remote Command Description

Note the following conventions used in the remote command descriptions:

- **Command usage**
  If not specified otherwise, commands can be used both for setting and for querying parameters.
  If a command can be used for setting or querying only, or if it initiates an event, the usage is stated explicitly.
- **Parameter usage**

If not specified otherwise, a parameter can be used to set a value and it is the result of a query.
Parameters required only for setting are indicated as **Setting parameters**.
Parameters required only to refine a query are indicated as **Query parameters**.
Parameters that are only returned as the result of a query are indicated as **Return values**.

- **Conformity**
  Commands that are taken from the SCPI standard are indicated as **SCPI confirmed**. All commands used by the R&S RTE follow the SCPI syntax rules.
- **Asynchronous commands**
  A command which does not automatically finish executing before the next command starts executing (overlapping command) is indicated as an **Asynchronous command**.
- **Reset values (*RST)**
  Default parameter values that are used directly after resetting the instrument (`*RST` command) are indicated as ***RST** values, if available.
- **Default unit**
  The default unit is used for numeric values if no other unit is provided with the parameter.

17.2 Finding the Appropriate Command

In the following chapters, the commands are sorted according to the menu and dialog structure of the instrument.

A list of all commands in alphabetical order ist given in the "List of Commands" at the end of this documentation.

To find the appropriate command for a setting easily, you can use the context help:

1. Enable the "Tooltip" icon on the toolbar.

2. Tap the parameter for which you need information.

   The tooltip opens.

3. Tap the "Show Help" button in the lower right corner of the tooltip.

   The "Help" window opens and displays the comprehensive description and the corresponding remote command.

4. Tap the remote command link to open the command description.

17.3 Programming Examples

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

17.3.1 Display....1019
17.3.1.1 Creating Diagrams (SmartGrid)....1019
17.3.1.2 Creating Zoom Diagrams....1019
17.3.2 Automatic Measurements....1020
17.3.2.1 Performing Amplitude/Time Measurements....1020
17.3.2.2 Setting Reference Levels....1020
17.3.2.3 Waveform Histograms....1021
17.3.2.4 Long-Term Measurements....1022
17.3.3 Mask Testing....1023
17.3.3.1 Creating a User Mask....1023
17.3.4 Search....1023
17.3.4.1 Searching for a Pulse of Specified Width....1023
17.3.5 Data Management....1024
17.3.5.1 Saving a Screenshot to File....1024
17.3.5.2 Exporting Waveform Data to File....1024
17.3.5.3 Exporting Measurement Results to File....1030
17.3.6 Protocol Analysis....1031
17.3.6.1 SENT (Option R&S RTE-K10)....1031
17.3.7 Power Analysis (Option R&S RTE-K31)....1033
17.3.7.1 Auto Deskew....1033
17.3.7.2 Transient Response Measurement....1034

17.3.1 Display

17.3.1.1 Creating Diagrams (SmartGrid)

The example creates 3 diagrams at different positions on the screen, and a zoom diagram.

Command description in: Chapter 17.7.2.3, "SmartGrid", on page 1057.

In the following example *OPC prevents overlapping execution of asynchronous commands.

```
:SYSTem:DISPlay:UPDate ON
:CHANnel2:STATe 1
// Turn on Channel 2
:LAYout:ADD 'Diagram1',VERtical,OFF,C2W1,'MyDiag2'
// Create new diagram 'MyDiag2' below existing 'Diagram1' with waveform C2W1 in it
:CHANnel3:STATe 1
:LAYout:ADD 'MyDiag2',HORizontal,OFF,C3W1,'MyDiag3'  s
// Create new diagram 'MyDiag3' right of existing 'MyDiag2'
:CHANnel4:STATe 1
:LAYout:ADD 'Diagram1',TAB,OFF,C4W1,'MyDiag4'
// Create new diagram 'MyDiag4' tabbed to existing 'Diagram1'
:LAYout:SHOW 'Diagram1'
// Select 'Diagram1'
:LAYout:ZOOM:ADD 'Diagram1',VERT,OFF,-10e-9,10e-9,-0.05,0.04,'MyZoom1'
// Create zoom window of 'Diagram1'
*OPC?
```

17.3.1.2 Creating Zoom Diagrams

The example creates a zoom diagram, sets the relative size of the zoom area, and removes the zoom diagram.

Command description in: Chapter 17.10.1, "Zoom", on page 1174.

In the following example *OPC prevents overlapping execution of asynchronous commands.

```
LAYout:ZOOM:ADD 'Diagram1', VERT, OFF, -10e-9, 20e-9, -0.1, 0.05, 'MyZoom1'
// Create an new zoom diagram for Diagram1
LAYout:ZOOM:HORZ:MODE? 'Diagram1', 'MyZoom1'
<--ABS
// Query the horizontal zoom mode - return value: ABS
LAYout:ZOOM:HORZ:MODE 'Diagram1', 'MyZoom1', REL
// Set horizontal zoom mode to relative
LAYout:ZOOM:HORZ:REL:SPAN 'Diagram1', 'MyZoom1', 10
// Set horizontal zoom span in percent
LAYout:ZOOM:HORZ:REL:POS 'Diagram1', 'MyZoom1', 15
// Set horizontal zoom position in percent
*OPC?
```

```
LAYout:ZOOM:REM 'Diagram1', 'MyZoom1'
// Remove zoom diagram
*OPC?
```

17.3.2 Automatic Measurements

17.3.2.1 Performing Amplitude/Time Measurements

Command description is given in Chapter 17.12.1, "General Settings", on page 1213 and Chapter 17.12.2, "Results", on page 1217

In the following example *OPC prevents overlapping execution of asynchronous commands.

Simple Frequency and Amplitude Measurement

```
SING;*OPC?
MEAS1:SOUR C1W1        // Configure frequency measurement
MEAS1:MAIN FREQ
MEAS1 ON
*OPC?
MEAS2:SOUR C1W1        // Configure amplitude measurement
MEAS2:MAIN AMPL
MEAS2 ON
*OPC?
MEAS1:RES:ACT?        // Get frequency result
MEAS2:RES:ACT?        // Get amplitude result
*OPC?
```

17.3.2.2 Setting Reference Levels

Command description in Chapter 17.12.13, "Reference Levels", on page 1271

In the following example *OPC prevents overlapping execution of asynchronous commands.

Manual Reference Level Definition Using Relative Values

Reference levels are set to 15%, 50%, and 85% of the high signal level for waveform Ch2 (= suffix 5).

```
REFLevel5:LDETection MANual
REFLevel5:LMODe REL
REFLevel5:RELative:MODE USER
REFLevel5:RELative:LOWer 15
REFLevel5:RELative:MIDDle 50
REFLevel5:RELative:UPPer 85
*OPC?
```

Manual Reference Level Definition Using Absolute Values

Set reference levels manually for waveform C1W1 (= suffix 2), defining high and low signal levels and the distances between signal and reference levels.

```
REFLevel2:LDETection MANual
REFLevel2:LMODe ABS
REFLevel2:ABSolute:HIGH 0.12
REFLevel2:ABSolute:TDIStance 0.03
REFLevel2:ABSolute:Low -0.12
REFLevel2:ABSolute:BDIStance 0.04
*OPC?
REFLevel2:ABSolute:MLEVel?
<-- 0
```

17.3.2.3 Waveform Histograms

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

Creating and Reading Histograms

The example creates a histogram, activates two measurements (mean and standard deviation measurements of Histogram1), and queries the results of both measurements.

Command description in:

- Chapter 17.12.1, "General Settings", on page 1213
- Chapter 17.12.6.2, "Histogram Measurement", on page 1246
- Chapter 17.12.2, "Results", on page 1217

```
LAY:HIST:ADD 'Histogram1', C1W1, -2.5E-007, 2.5E-007, -1.32, 5.35, OFF, VERT *OPC?

MEAS1 ON
MEAS1:HIST:SEL 'Histogram1'
MEAS1:CAT HIST
MEAS1:MAIN HME
*OPC?

MEAS2 ON
MEAS2:HIST:SEL 'Histogram1'
MEAS2:CAT HIST
MEAS2:MAIN HSTD
*OPC?

MEAS1:RES:ACT?
MEAS2:RES:ACT?
```

Exporting Histogram Data to File

The example writes the absolute data values of Histogram1 to `C:\Histograms\Hist1.xml` in XML format.

Command description in Chapter 17.16.6, "Waveform Histogram Export to File", on page 1369.

```
EXPort:HISTogram:SELect 'Histogram1'
EXPort:HISTogram:INCidence ABS
EXPort:HISTogram:NAME 'C:\Histograms\Hist1.xml'
EXPort:HISTogram:SAVE
*OPC?
```

Transferring Histogram Data

The example transfers the absolute values of Histogram1 to a controlling computer in ASCII format.

Command description in Chapter 17.16.6, "Waveform Histogram Export to File", on page 1369.

```
EXP:HIST:SEL 'Histogram1'
EXP:HIST:INC ABS
*OPC?
FORM ASC
EXP:HIST:DATA?
<--0,0,0,0,0,2037,5754804,4683496,3100169,2874565,...
```

17.3.2.4 Long-Term Measurements

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

Exporting Long-Term Measurement Data to File

The example writes the long-term data of Meas1 to `C:\Measurements\Meas1.csv` in CSV format.

Command description in Chapter 17.16.9, "Long Term Measurement Results and Measurement Histogram Export to File", on page 1372.

```
EXPort:MEASurement:SEL MEAS1
EXPort:MEASurement:TYPE LONGTERM
EXPort:MEASurement:NAME 'C:\Measurements\Meas1.csv'
EXPort:MEASurement:SAVE
*OPC?
```

Transferring Long-Term Measurement Data

The example transfers the long-term data of Meas1 to a controlling computer in ASCII format.

Command description in Chapter 17.16.9, "Long Term Measurement Results and Measurement Histogram Export to File", on page 1372.

```
MEASurement:LTM ON
MEASurement:STAT ON
EXPort:MEASurement:SELect MEAS1
EXPort:MEASurement:TYPE LONGTERM
*OPC?
FORM ASC
EXPort:MEASurement:DATA?
<--50,0.24901185771,0.24731225296,0.24703557312,0.00069270717936,0,50,....
```

17.3.3 Mask Testing

17.3.3.1 Creating a User Mask

Creates new user mask "MyMask" with one inner segment, and turns on the mask test.

Command description in: Chapter 17.14, "Mask Testing", on page 1294.

In the following example *OPC prevents overlapping execution of asynchronous commands.

```
MTEST:ADD 'MyMask'
MTEST:SEGM:ADD 'MyMask'
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 0, -20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 0, -0.1
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 1, -20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 1, 0.1
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 2, 20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 2, 0.1
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 3, 20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 3, -0.1
MTEST:SEGM:REG 'MyMask', 0, INNER
MTEST:STAT 'MyMask', ON
*OPC?
```

17.3.4 Search

17.3.4.1 Searching for a Pulse of Specified Width

In the following example *OPC prevents overlapping execution of asynchronous commands.

Search for positive pulses with pulse width 12 ± 10 µs (2 µs to 22 µs).

Command description in: Chapter 17.15, "Search", on page 1314.

The usage of asynchronous commands is described in Chapter B.3, "Command Sequence and Synchronization", on page 1980.

```
SEAR:ADD 'MySearch'                    // Create a new search
SEAR:TRIG:WIDT:STAT 'MySearch',1       // Configure search type
SEAR:SOUR 'MySearch',M1                // Configure search source - here Math1
SEAR:TRIG:WIDT:RANG 'MySearch',WITH    // Configure search parameters
SEAR:TRIG:WIDT:WIDT 'MySearch',5e-4    // Configure search parameters
SEAR:TRIG:WIDT:DELT 'MySearch',5e-6    // Configure search parameters
SEAR:RES:LIM 'MySearch',1              // Set number of result lines in table to 1
*OPC?
SEAR:ALL 'MySearch';                  // Initiate search for all events, asynchr. command
```

17.3.5 Data Management

- Saving a Screenshot to File....................1024
- Exporting Waveform Data to File....................1024
- Exporting Measurement Results to File....................1030

17.3.5.1 Saving a Screenshot to File

Saves three display images in png format to the files `Print.png`, `Print_001.png`, and `Print_002.png` in the directory `C:\Temp`. To get a correct screenshot, turn on the display first.

Command description in: Chapter 17.16.10, "Screenshots", on page 1374.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
SYST:DISP:UPD ON
HCOP:DEST 'MMEM'
HCOP:DEV:LANG PNG
*OPC?
MMEM:NAME 'C:\Temp\Print.png'
HCOP:IMMediate; *OPC?
HCOP:IMM:NEXT; *OPC?
HCOP:IMM:NEXT; *OPC?
```

17.3.5.2 Exporting Waveform Data to File

Command description in:

- Chapter 17.16.5, "Waveform Data Export to File", on page 1362
- Chapter 17.16.1, "Instrument Settings", on page 1353
- Chapter 17.10.4, "History", on page 1194

In the following example *OPC prevents overlapping execution of asynchronous commands.

- Exporting a Single Waveform to XML File.. 1025
- Exporting Raw Data of a Single Waveform to BIN File.................................... 1025
- Exporting Raw Data of a Measurement Gate to BIN File..................................1026
- Exporting Interleaved x/y Data of a Single Waveform to CSV File.................... 1026
- Exporting Interleaved x/y Data of a Zoom to CSV File.......................................1027
- Exporting Multiple Running Acquisitions of a Single Waveform to XML File...... 1027
- Exporting a Single Acquisition of the History to BIN File...................................1028
- Exporting Multiple Acquisition of the History to XML File..................................1028
- Exporting and Reconstructing Multiple Raw Acquisitions of the History File...... 1029

Exporting a Single Waveform to XML File

Saves a single analog waveform completely to an XML file. Data logging is off.

The usage of asynchronous commands is described in Chapter B.3, "Command Sequence and Synchronization", on page 1980.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.xml'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.xml'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.xml'
```

Exporting Raw Data of a Single Waveform to BIN File

Saves the data of a single analog waveform in integer 8-bit format (raw data) to a BIN file. Data logging is off.

Data conversion is described in "Raw (ADC direct)" on page 449.

In the following example *OPC prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
RUNSingle; *OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.bin'
```

```
EXPort:WAVeform:RAW ON
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.bin'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.bin'
```

Exporting Raw Data of a Measurement Gate to BIN File

Saves the data of a measurement gate in integer 8-bit format (raw data) to a BIN file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
MEASurement2:CATegory AMPT
MEASurement2:MAIN MEAN
MEASurement2:ENABle 1
MEASurement2:SOURce C1W1
MEASurement2:GATE:MODE ABS
MEASurement2:GATE:ABS:STARt -0.00012
MEASurement2:GATE:ABS:STOP -5e-06
MEASurement2:GATE:STATe On
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe GATE
EXPort:WAVeform:MEAS Meas2
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.bin'
EXPort:WAVeform:RAW ON
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.bin'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.bin'
```

Exporting Interleaved x/y Data of a Single Waveform to CSV File

Saves the x- and y- values of a single analog waveform to a CSV file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
```

```
CHANnel1:WAVeform1:STATe 1
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.csv'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues ON
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.csv'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.csv'
```

Exporting Interleaved x/y Data of a Zoom to CSV File

Saves the x- and y- values that is displayed in a zoom diagram to a CSV file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
LAYout:ZOOM:ADD 'Diagram1',HORIZONTAL,OFF,-0.00012,-5e-06,0.308,-0.092,'ExportAreaZoom'
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe ZOOM
EXPort:WAVeform:ZOOM 'Diagram1', 'ExportAreaZoom'
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.csv'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues ON
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.csv'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.csv'
```

Exporting Multiple Running Acquisitions of a Single Waveform to XML File

Saves the data of 5 subsequent acquisitions of a single analog waveform to an XML file. Data logging is on.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
```

```
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.xml'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
*OPC?
ACQuire:COUNt 5
EXPort:WAVeform:DLOGging ON
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
RUNSingle;*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.xml'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.xml'
```

Exporting a Single Acquisition of the History to BIN File

Saves the oldest acquisition of the history to a BIN file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.bin'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
*OPC?
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
ACQuire:COUNt 5            //Acquire 5 waveforms
RUNSingle;*OPC?
CHANnel1:WAV1:HISTory:STATe ON
CHANnel1:WAV1:HISTory:CURRent -4;*OPC?  //Oldest waveform of 5 has index -4
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.bin'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.bin'
```

Exporting Multiple Acquisition of the History to XML File

Saves the data of 5 subsequent acquisitions of the history to an XML file. Data logging is on.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
```

```
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.xml'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging ON
*OPC?
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
ACQuire:COUNt 5               //Acquire 5 waveforms
RUNSingle;*OPC?
CHANnel1:WAV1:HISTory:STATe ON
CHANnel1:WAV1:HISTory:STARt -4
CHANnel1:WAV1:HISTory:STOP 0
CHANnel1:WAV1:HISTory:REPLay OFF
CHANnel1:WAV1:HISTory:PLAY
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.xml'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.xml'
```

Exporting and Reconstructing Multiple Raw Acquisitions of the History File

This example captures and exports 100 waveforms acquired in fast segmentation mode (minimize blind time). The data is transferred in blocks using SCPI commands.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
// --- Set data format to signed integers, 1 byte
FORMat:DATA INT,8

// --- Create history data (assuming on channel 1) ---
CHANnel1:WAVeform1:STATe 1           //turn on channel 1
ACQuire:COUNt 100                    //acquire 100 waveforms
ACQuire:SEGMented:STATe ON; *OPC?
//turn on fast segmentation, acquisition is started and data is stored in the memory


// --- Set Export variables and export data ---
EXPort:WAVeform:SOURce C1W1         //specify source for data export
EXPort:WAVeform:SCOPe WFM           //specify range: complete acq. time
MMEM:DEL "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExportWfm_analog.*"
        //delete data from previous runs of this script
EXPort:WAVeform:NAME "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExpWfm.bin"
        //store data in this path and filename
EXPort:WAVeform:RAW ON
//export as raw ADC integer values (saves memory) --> data needs to be converted later
EXPort:WAVeform:INCXvalues OFF
//disable time values in data file. Time can be constructed from the header file
CHANnel1:WAV1:HISTory:STATe ON      //switch to history view
EXPort:WAVeform:DLOGging ON         //enable data logging & history
EXPort:WAVeform:TIMestamps ON       //enable relative time stamp for each acq.
CHANnel1:WAV1:HISTory:STARt -99     //oldest waveform of n acq. has index =(-1)*(n-1)
CHANnel1:WAV1:HISTory:STOP 0        //newest waveforms has index 0
```

```
CHANnel1:WAV1:HISTory:REPLay OFF
CHANnel1:WAV1:HISTory:PLAY; *OPC?   //exports waveforms to defined file location.
//2 files are created: header file *.bin and waveform data file *.Wfm.bin


// --- Put files into output buffer of scope and collect ---
// The following code lines are mostly Pseudo Code.
//Sorting out the binary waveform data is more complex and require additional coding
binaryFormat = '1 byte'
//Pseudo Code, tell your language how to interpret the binary data,
//e.g. 'int8' for MATLAB
MMEM:DATA? "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExpWfm.bin"
//Put header file into output buffer
header = readSCPIBinary(visaInstrument, binaryFormat);
//Pseudo Code, use appropriate command from your programming language,
//e.g. binblockread in MATLAB
MMEM:DATA? "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExpWfm.Wfm.bin"
//Put data file into output buffer
//(your input buffer of the VISA resource might need to be increased)
wfmRaw = readSCPIBinary(visaInstrument, binaryFormat)
//Pseudo Code, use appropriate command from your programming language,
//e.g. binblockread in MATLAB.
//Note: Sort different acquisitions into an array separately after file transfer.


// --- Convert raw ADC values into voltage floating point values ---
// header is assumed to be a struct and the members are accessed via "." syntax.
vertOffsetByPosition = header.VerticalScale * header.VerticalPosition
conversionFactor = (1/header.NofQuantisationLevels) *
  header.VerticalScale * header.VerticalDivisionCount
for(i = 0; i<100; i++){
    wfmVolt(i) = wfmRaw(i) * conversionFactor +
      header.VerticalOffset - vertOffsetByPosition
}
// Note: Depending on settings, the waveform can contain more samples than the
//record length. Remove leading and trailing samples from the waveform.
```

17.3.5.3 Exporting Measurement Results to File

See:

- "Exporting Histogram Data to File" on page 1022
- "Transferring Histogram Data" on page 1022
- "Exporting Long-Term Measurement Data to File" on page 1022
- "Transferring Long-Term Measurement Data" on page 1022

17.3.6 Protocol Analysis

17.3.6.1 SENT (Option R&S RTE-K10)

Configuring SENT Bus

```
// Set protocol parameters
//*****************************************************
STOP;*OPC?
BUS:TYPE SENT
BUS:LABel "Bus 1 SENT"
BUS:SENT:DATA:SOURce C1W1
BUS:SENT:DATA:THReshold 2.0
BUS:SENT:CLKPeriod 0.000005
BUS:SENT:CLKTolerance 20.0
BUS:SENT:CRCVersion V2010
BUS:SENT:CRCMethod TLE
BUS:SENT:DNIBbles 5
BUS:SENT:PPULse NPP
BUS ON

//*****************************************************
// Load a label list and switch on
BUS:NEWList 'C:\Protocols\SENT_Labels.xml'
BUS:SYMBols ON
RUNSingle;*OPC?              //asynchronous command

//*****************************************************
// Display all results
BUS:SENT:RDSL ALL

//*****************************************************
// Display the fast channel transmission sequence
BUS:SENT:RDSL TRSQ

//*****************************************************
// Display the short serial message of slow channel
BUS:SENT:RDSL SMSG
```

Triggering on SENT Bus

```
//Set trigger source to serial bus
TRIGger1:SOURce SBUS
TRIGger:MODE NORMal

//Trigger on the fast channel transmission sequence
TRIGger1:SENT:TYPE TSEQ
TRIGger1:SENT:TTYPE STDA
```

```
TRIGger1:SENT:statbit #H2
TRIGger1:SENT:DCONdition INR
BUS1:SENT:DNibbles 5
TRIGger1:SENT:DMIN #H2 //Data MIN = 0010
TRIGger1:SENT:DMAX #H4 //Data MAX = 0100
```

Searching SENT Data

```
// Search for all frames with sync/calibration pulse
SEARch:TRIGger:SENT:CALibration 'Search1', ON
SEARch:ONLIne 'Search1',ON
SEARch:RESult:SORT:ASCending 'Search1', ON
SEARch:RESult:SENT:FCOunt? 'Search1'
SEARch:RESult:SENT:FRAMe1:STAT? 'Search1'
SEARch:RESult:SENT:FRAMe1:STARt? 'Search1'
SEARch:RESult:SENT:FRAMe1:STOP? 'Search1'
SEARch:RESult:SENT:FRAMe1:DATA? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB1:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB2:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:CSValue? 'Search1'
SEARch:RESult:SENT:FRAMe1:SCOM? 'Search1'
SEARch:TRIGger:SENT:CALibration 'Search1', OFF

//*****************************************************
// Search Short serial message
SEARch:TRIGger:SENT:SHORt 'Search1', ON
SEARch:RESult:SENT:FCOunt? 'Search1'

SEARch:RESult:SENT:FRAMe1:STAT? 'Search1'
SEARch:RESult:SENT:FRAMe1:IDValue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB1:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB2:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:CSValue? 'Search1'

SEARch:TRIGger:SENT:SHORt 'Search1', OFF

//*****************************************************
// Search Enhanced serial message
SEARch:TRIGger:SENT:ENHanced 'Search1', ON

SEARch:RESult:SENT:FCOunt? 'Search1'

SEARch:RESult:SENT:FRAMe1:STAT? 'Search1'
SEARch:RESult:SENT:FRAMe1:IDValue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB1:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB2:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:CSValue? 'Search1'

SEARch:TRIGger:SENT:ENHanced 'Search1', OFF
```

17.3.7 Power Analysis (Option R&S RTE-K31)

17.3.7.1 Auto Deskew

Configures the voltage and current probes for power measurements and executes the auto deskew.

Command description in Chapter 17.20.1, "General", on page 1906 and Chapter 17.20.2, "Deskew", on page 1907.

If the instrument refuses to accept `POWer` commands, activate the power mode using `:POWer:ENABle`.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
*RST; *OPC?
STOP;*OPC?
//Activate two channels
:CHANnel1:STATe 1
:CHANnel2:STATe 1

//Activate power
:POWer:ENABle

//Select current and voltage sources
:POWer:SOURce:VOLTage1 CHANnel1
:POWer:SOURce:CURRent1 CHANnel2
*OPC?

//Configure voltage probe on CH1 manually
//Not necessary if you use an active R&S voltage probe that is recognized by the instrument
:PROBe1:SETup:ATTenuation:MODE Manual
:PROBe:SETup:ATTenuation:DEFProbe ZD01a100
*OPC?
//selected high voltage differential probe 1:100

//Configure current probe on CH2 manually
//Always required because R&S current probes are not recognized automatically
PROBe2:SETup:ATTenuation:MODE Manual
PROBe2:SETup:ATTenuation:DEFProbe ZC20
*OPC?
//select 20MHz current probe ZC10 also possible

//Start deskew
//Overwrites the skew offset of CH2 (current probe), because :POWer:DESKew:RESet? == 1
//writes a user-defined preset file (UserDefinedPreset_AutoDeskew.dfl) and
//activates the user defined preset, because :POWer:DESKew:UDPReset? == 1
```

```
:PoWer:DESKew:EXECute;

//Check result
CHANnel2:SKEW:MAN?
CHANnel2:SKEW:TIMe?
POWer:DESKew:TIMe?
*OPC?
```

Effect of *RST and loading user-defined preset

Note that `*RST` resets the deskew values.

You can reload the deskew values as follows:

```
*RST;*OPC?
STOP;*OPC?
:POWer:ENABle
//Select voltage and current sources
:POWer:SOURce:VOLTage1 CHANnel1
:POWer:SOURce:CURRent1 CHANnel2
*OPC?
//Reload deskew values
:POWer:DESKew:CURRent
*OPC?
//Load default saveset after FW restart
MMEM:RCL 'C:\Users\Public\Documents\Rohde-Schwarz\RTx\SaveSets
\UserDefinedPreset_AutoDeskew.dfl'
```

17.3.7.2 Transient Response Measurement

Configures and executes a transient response measurement.

Command description in Chapter 17.20.15, "Transient Response", on page 1941.

Make sure to configure and deskew the probes before the measurement, see Chapter 17.3.7.1, "Auto Deskew", on page 1033.

```
//Activate power
*RST; *OPC?
:POWer:ENABle

//Expected smps frequency
:POW:TRANsient:FREQ         12500000

:POWer:TRANsient:AUToscale AUTO

:POWer:TRANsient:SIGHigh 0.1
:POWer:TRANsient:SIGLow    0.025
:POWer:TRANsient:HYSTeresis 20
```

```
//Trigger channel
:POWer:TRANsient:TRGC CHAN2

//Edge trigger slope
:POWer:TRANsient:TRGS POS

//Trigger level
:POWer:TRANsient:TRGL 0.08

//Run measurement
:POWer:TRANsient:EXECute;*OPC?

//Query results
:POWer:TRANsient:RESult? SETTling
:POWer:TRANsient:RESult? PEAKtime
:POWer:TRANsient:RESult? DELay
:POWer:TRANsient:RESult? RTIMe
:POWer:TRANsient:RESult? OVERshoot

//Add to report
:Power:TRANsient:REPort:Add
```

17.4 Frequently Used Parameters and Suffixes

This chapter describes in general those parameters and suffixes that are used in several subsystems.

17.4.1 Waveform Suffix

The numeric waveform suffix is used in some commands, for example, to indicate the source waveform number from which the reference level is taken, and to assign color tables to waveforms.

Depending on the command, not all suffix values are supported. For example, in REFLevel commands, only suffixes 2 to 21 are allowed. The range of supported suffix numbers is indicated in the description of the individual commands.

NOTICE

Suffix 1

Suffix 1 means that no waveform is assigned. The first waveform C1W1 corresponds to suffix number 2.

Waveform number	Description
1	None
2	C1 (channel 1)
3	not available
4	not available
5	C2 (channel 2)
6	not available
7	not available
8	C3 (channel 3)
9	not available
10	not available
11	C4 (channel 4)
12	not available
13	not available
14 to 17	Math waveforms: M1, M2, M3, M4 M5 to M8: suffixes 119 to 122
18 to 21	Reference waveforms: R1, R2, R3, R4
22 to 25	XY-waveforms: XY1, XY2, XY3, XY4
26 to 33	Measurement results: MRESult1, MRESult2, MRESult3, MRESult4, MRESult5, MRESult6, MRESult7, MRESult8
34 to 35	not used
36 to 39	Serial buses: SBUS1, SBUS2, SBUS3, SBUS4
40 to 55	Digital channels: D0 to D15 (option R&S RTE-B1)
56 to 59	Digital buses: MSO1, MSO2, MSO3, MSO4 (option R&S RTE-B1)
60	not used
61 to 68	Track waveforms: TRK1, TRK2, TRK3, TRK4, TRK5, TRK6, TRK7, TRK8
69 to 71	not used
72 to 75	Spectrograms: SG1, SG2, SG3, SG4. Available for option R&S RTE-K18.
76 to 83	Timeline spectrums: SG1TL1, SG1TL2, SG2TL1, SG2TL2, SG3TL1, SG3TL2, SG4TL1, SG4TL2. Available for option R&S RTE-K18 TL1 is timeline 1, TL2 is timeline 2
88 to 91	Voltage input channels of multi-channel probe R&S RT-ZVC (probe 1) R&S RT-ZVC04: Z1V1 | Z1V2 | Z1V3 | Z1V4 R&S RT-ZVC02: Z1V1 | Z1V2. Suffixes 90 to 91 are not available.

Waveform number	Description
92 to 95	Current input channels of multi-channel probe R&S RT-ZVC (probe 1) R&S RT-ZVC04: Z1I1 | Z1I2 | Z1I3 | Z1I4 R&S RT-ZVC02: Z1I1 | Z1I2. Suffixes 94 to 95 are not available.
96 to 99	Voltage input channels of multi-channel probe R&S RT-ZVC (probe 2) R&S RT-ZVC04: Z2V1 | Z2V2 | Z2V3 | Z2V4 R&S RT-ZVC02: Z2V1 | Z2V2. Suffixes 98 to 99 are not available.
100 to 103	Current input channels of multi-channel probe R&S RT-ZVC (probe 2) R&S RT-ZVC04: Z2I1 | Z2I2 | Z2I3 | Z2I4 R&S RT-ZVC02: Z2I1 | Z2I2. Suffixes 102 to 103 are not available.
104 to 107	not used
108 to 111	Resulting waveforms of differential deembedding: DIFF1 | DIFF2 | COMMON1 | COMMON2
115 to 118	Math waveforms: M5, M6, M7, M8 M1 to M4: suffixes 14 to 17

17.4.2 Waveform Parameter

Many commands requires one of the waveforms to be specified as source. The following table lists all waveforms. For each command using a waveform parameter, the available waveforms are specified in the command description.

Waveform	Description
C1W1	Channel 1
C2W1	Channel 2
C3W1	Channel 3
C4W1	Channel 4
M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8	Math waveforms
R1 | R2 | R3 | R4	Reference waveforms
XY1 | XY2 | XY3 | XY4	XY-waveforms
MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8	Measurement results
SBUS1 | SBUS2 | SBUS3 | SBUS4	Serial buses
D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15	Digital channels (option R&S RTE-B1)
MSOB1 | MSOB2 | MSOB3 | MSOB4	Digital buses (option R&S RTE-B1)
TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8	Track waveforms
SG1 | SG2 | SG3 | SG4	Spectrograms (option R&S RTE-K18)

Waveform	Description
SG1TL1, SG1TL2, SG2TL1, SG2TL2, SG3TL1, SG3TL2, SG4TL1, SG4TL2	Spectrograms (option R&S RTE-K18)
Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 Z1V1 | Z1V2 | Z1I1 | Z1I2 | Z2V1 | Z2V2 | Z2I1 | Z2I2	Input channels of multi-channel probe R&S RT-ZVC04 Input channels of multi-channel probe R&S RT-ZVC02

17.4.3 Slope Parameter

The slope parameter is used with several trigger and search condition commands.

Slope	Description
POSitive	Rising edge, that is a positive voltage change.
NEGative	Falling edge, that is a negative voltage change
EITHer	rising as well as the falling edge.

17.4.4 Polarity Parameter

The polarity parameter is used with several trigger and search condition commands.

Polarity	Description
POSitive	Positive going pulses.
NEGative	Negative going pulses.
EITHer	Both positive and negative going pulses.

17.4.5 Event Parameter

The event parameter is used with commands defining an action for mask testing, limit checks and margin checks.

Event	Description
NOACtion	The action is not initiated.
SUCCess	The action is initiated if the operation finished successfully: ● Limits or margins were not exceeded during the entire measurement ● Mask test passed
VIOLation	The action is initiated if the operation finished with error: ● Limits or margins were violated during the measurement ● Mask test failed

17.4.6 Bit Pattern Parameter

Bit pattern parameter are required with commands triggering on address, identifier, or data pattern.

To set the pattern value, you can use either a numeric parameter as defined in the SCPI standard, or a string parameter.

Bit pattern in numeric parameter

In a numeric parameter, the values are listed byte-by-byte, with bytes separated by commas and MSB first. The default numeral format is decimal, other formats can be indicated by a format identifier (#B = binary, #H = hexadecimal, #Q = octal). Currently, no format for signed values is available.

Example: Parameter with three bytes, decimal byte values are 10, 20, 30. The examples are given for CAN, the bit pattern in other commands is defined in the same way.

- `TRIGger:CAN:DMIN 10,20,30`
- `TRIGger:CAN:DMIN #B00001010,#B00010100,#B00011110`
- `TRIGger:CAN:DMIN #H0A,#H14,#H1E`
- `TRIGger:CAN:DMIN #Q012,#Q024,#Q036`

Bit pattern in string parameter

In a string, the complete binary pattern is written without separation of bytes, for example:

`TRIGger:CAN:DMIN '000010100001010000011110'`

Unlike a numeric parameter, the string parameter accepts wildcards for single bits (X = don't care). Whether wildcards can be used or not depends on the remote command. Usually, address and identifier parameter require unique patterns while data parameters may contain wildcards.

Mostly the length of the bit pattern is defined, for example, by the I²C address type, the CAN identifier type, or the data length code. In theses cases, it is recommended that you enter the complete bit pattern. If you enter a shorter pattern, the instrument fills up the pattern with X bits to the right of the defined pattern.

Example: You want to trigger on an 11 bit CAN address and enter the bit pattern '11100011' (8 bits only). The instrument uses the pattern '11100011XXX' for triggering.

Query for a pattern

The pattern format for the return value of a pattern is defined by the `FORMat:BPATtern` command.

17.5 Common Commands

Common commands are described in the IEEE 488.2 (IEC 625-2) standard. These commands have the same effect and are employed in the same way on different devices. The headers of these commands consist of "*" followed by three letters. Many common commands are related to the Status Reporting System.

Available common commands:

*CAL?....................1040
*CLS....................1040
*ESE....................1041
*ESR?....................1041
*IDN?....................1041
*IST?....................1041
*OPC....................1041
*OPT?....................1042
*PCB....................1042
*PRE....................1042
*PSC....................1042
*RCL....................1043
*RST....................1043
*SAV....................1043
*SRE....................1043
*STB?....................1043
*TRG....................1044
*TST?....................1044
*WAI....................1044

*CAL?

Starts a self-alignment of the instrument, and then queries a status response. Return values ≠ 0 indicate an error.

Return values:

<State>	0: no error 1: alignment failed 2: not aligned, e.g. init 3: device needs longer warmup time before selfalignment can start 4: input signal connected during selfalignment
Usage:	Query only

*CLS

Clear status

Sets the status byte (STB), the standard event register (ESR) and the `EVENt` part of the `QUEStionable` and the `OPERation` registers to zero. The command does not alter the mask and transition parts of the registers. It clears the output buffer.

Usage: Setting only

***ESE** <Value>

Event status enable

Sets the event status enable register to the specified value. The query returns the contents of the event status enable register in decimal form.

Parameters:
<Value> Range: 0 to 255

***ESR?**

Event status read

Returns the contents of the event status register in decimal form and then sets the register to zero.

Return values:
<Contents> Range: 0 to 255

Usage: Query only

***IDN?**

Identification

Returns the instrument identification.

Return values:
<ID> "Rohde&Schwarz,<device type>,<serial number>,<firmware version>"

Example: `Rohde&Schwarz,RTE,1326.2000k24/200153,3.50.0.2`

Usage: Query only

***IST?**

Individual status query

Returns the contents of the IST flag in decimal form. The IST flag is the status bit which is sent during a parallel poll.

Return values:
<ISTflag> 0 | 1

Usage: Query only

***OPC**

Operation complete

Sets bit 0 in the event status register when all preceding commands have been executed. This bit can be used to initiate a service request. The query writes a "1" into the output buffer when all preceding commands have been executed, which is useful for command synchronization.

***OPT?**

Option identification query

Queries the options included in the instrument. For a list of all available options and their description, refer to the data sheet.

Return values:

<Options>	The query returns a list of options. The options are returned at fixed positions in a comma-separated string. A zero is returned for options that are not installed.
Usage:	Query only

***PCB** <Address>

Pass control back

Indicates the controller address to which remote control is returned after termination of the triggered action.

Setting parameters:

<Address>	Range: 0 to 30
Usage:	Setting only

***PRE** <Value>

Parallel poll register enable

Sets parallel poll enable register to the indicated value. The query returns the contents of the parallel poll enable register in decimal form.

Parameters:

<Value>	Range: 0 to 255

***PSC** <Action>

Power on status clear

Determines whether the contents of the `ENABle` registers are preserved or reset when the instrument is switched on. Thus a service request can be triggered when the instrument is switched on, if the status registers ESE and SRE are suitably configured. The query reads out the contents of the "power-on-status-clear" flag.

Parameters:

<Action>	0 | 1

0
The contents of the status registers are preserved.

1
Resets the status registers.

***RCL** <Number>

Recall

Loads the instrument settings from an intermediate memory identified by the specified number. The instrument settings can be stored to this memory using the command `*SAV` with the associated number.

It also activates the instrument settings which are stored in a file and loaded using `MMEMory:LOAD:STATe`.

***RST**

Reset

Sets the instrument to a defined default status. The default settings are indicated in the description of commands.

Usage: Setting only

***SAV** <Number>

Save

Stores the current instrument settings under the specified number in an intermediate memory. The settings can be recalled using the command `*RCL` with the associated number.

To transfer the stored instrument settings to a file, use `MMEMory:STORe:STATe`.

***SRE** <Contents>

Service request enable

Sets the service request enable register to the indicated value. This command determines under which conditions a service request is triggered.

Parameters:

<Contents> Contents of the service request enable register in decimal form. Bit 6 (MSS mask bit) is always 0.

Range: 0 to 255

***STB?**

Status byte query

Reads the contents of the status byte in decimal form.

Usage: Query only

***TRG**

Trigger

Triggers all actions waiting for a trigger event. In particular, *TRG generates a manual trigger signal. This common command complements the commands of the `TRIGger` subsystem.

Usage: Event

***TST?**

Self-test query

Initiates self-tests of the instrument and returns an error code.

Return values:

<ErrorCode> **integer > 0 (in decimal format)**
An error occurred.
(For details, see the Service Manual supplied with the instrument).

0
No errors occurred.

Usage: Query only

***WAI**

Wait to continue

Prevents servicing of the subsequent commands until all preceding commands have been executed and all signals have settled (see also command synchronization and `*OPC`).

Usage: Event

17.6 General Remote Settings

This chapter describes commands that have effect on many other remote commands in different applications of the instrument.

FORMat[:DATA]......1045
FORMat:BORDer......1046
FORMat:BPATtern......1046
SYSTem:DISPlay:UPDate......1047
SYSTem:KLOCk......1047
GPIB:ADDRess......1047

GPIB:TERMinator.. 1047
SYSTem:DISPlay:MESSage:STATe... 1047
SYSTem:DISPlay:MESSage[:TEXT]..1048

FORMat[:DATA] <Format>, [<Length>]

Selects the data type that is used for transmission of data from analog channels, math and reference waveforms, and some measurement results from the instrument to the controlling computer.

The command sets the data format for the following query commands:

- `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?`
- `CALCulate:MATH<m>:DATA[:VALues]?`
- `REFCurve<m>:DATA[:VALues]?`
- `EXPort:HISTogram:DATA?` on page 1370
- `EXPort:MEASurement:DATA?` on page 1373
- `MEASurement<m>:TRACk:DATA[:VALues]?` on page 1265

Parameters:

<Format>,[<Length>] ASCii | REAL,32 | INT,8 | INT,16

ASCii
Data values are returned in ASCII format as a list of comma-separated values in floating point format. The length can be omitted. It is 0 which means that the instrument selects the number of digits to be returned. The query returns both values (`ASC,0`).

REAL,32
The data is stored as binary data (Definite Length Block Data according to IEEE 488.2). Each waveform value is formatted in 32-Bit IEEE 754 Floating Point Format.
The schema of the result string is as follows:
`#41024<value1><value2>…<value n>` with:
`#4` = number of digits (= 4 in the example) of the following number
`1024` = number of following data bytes (= 1024 in the example)
`<value>` = 4-byte floating point values
For large data (≥ 1 GB), the result string starts with "#(data length)". The number inside the parentheses indicates the real data length in bytes.

INT,8 | INT,16
Signed integer data with length 8 bit or 16 bit. It defines that `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?` returns the raw sample data of the ADC as integers. If format of the waveform data differs from the defined export format, the instrument converts the data to the required length.
The result string has the same schema as the REAL format.
Fot INT,16 you can set the byte order using the `FORMat:BORDer` command.
Data conversion is described in "Raw (ADC direct)" on page 449.
For digital channel data, math and histogram data, INT formats are not available.
`EXPort:WAVeform:INCXvalues` must be set `OFF`.

*RST: ASCii

Example:
```
FORMat:DATA REAL,32
FORMat:DATA?
REAL,32
```

Usage: SCPI confirmed

FORMat:BORDer <ByteOrder>

Sets the endianess.

The command is only relevant for raw data export in high definition mode (16 bit word length).

Parameters:
<ByteOrder> LSBFirst | MSBFirst

LSB first: little endian, least significant byte first
MSB first: big endian, most significant byte first

*RST: LSBFirst

Usage: Asynchronous command

FORMat:BPATtern <BitPatternFormat>

Sets the number format for remote bit pattern queries on serial protocols.

Parameters:
<BitPatternFormat> DEC | HEX | OCT | BIN | ASCII | ASCii | STRG

ASCII = ASCii

*RST: HEX

SYSTem:DISPlay:UPDate <Enable>

Defines whether the display is updated while the instrument is in the remote state. If the display is switched off, the normal GUI is replaced by a static image while the instrument is in the remote state. Switching off the display can speed up the measurement. This is the recommended state.

See also: Chapter 16.7.2, "Using the Display during Remote Control", on page 1015

Parameters:

<Enable> **ON | 1:** Display is shown and updated during remote control
OFF | 0: Display shows static image during remote control

Example: `SYSTem:DISPlay:UPDate 1`
Switch on the display update.

SYSTem:KLOCk <Enable>

Locks or unlocks the local controls of the instrument. This includes the front panel keys, the keyboard, or other local interfaces. except for the "View" button on the display.

Parameters:

<Enable> **ON | 1:** Locks the local keys
OFF | 0: Keys are unlocked

Usage: SCPI confirmed

GPIB:ADDRess <Address>

Sets the GPIB address of the instrument. Changing the address has major effects on the communication to the remote computer.

Parameters:

<Address> Range: 0 to 30
Increment: 1
*RST: 20

GPIB:TERMinator <Terminator>

Specifies the symbol that is used as a terminator in GPIB communication.

Parameters:

<Terminator> LFEoi | EOI
*RST: EOI

SYSTem:DISPlay:MESSage:STATe <DispMessSt>

Enables and disables the display of an additonal text in remote control.

To define the text, use `SYSTem:DISPlay:MESSage[:TEXT]`.

Parameters:

<DispMessState> ON | OFF

*RST: OFF

Firmware/Software: Version 2.70

SYSTem:DISPlay:MESSage[:TEXT] <DisplayMessage>

Defines an additional text that is displayed during remote control operation.

To enable the text display, use `SYSTem:DISPlay:MESSage:STATe` on page 1047.

Parameters:

<DisplayMessage> String that contains the text.

Firmware/Software: Version 2.70

17.7 Instrument Setup

This chapter describes commands related to SETUP > "System" and "File" > "Exit". For commands related to SETUP > "Remote Settings", see Chapter 17.6, "General Remote Settings", on page 1044.

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- System Setup....................1048
- Display Settings....................1052
- External Application....................1069

17.7.1 System Setup

SYSTem:EXIT

Starts the shutdown of the firmware.

Usage: Event

SYSTem:SHUTdown

Starts the shutdown of the instrument (firmware and operating system).

Usage: Event

SYSTem:PRESet

Resets the instrument to the factory default settings, to the initial state. Factory settings comprise all instrument settings, including display, intensity and transparency settings. After loading factory defaults, perform a self-alignment to synchronize the signal data.

Usage: Event

SYSTem:DATE <Year>, <Month>, <Day>

Sets the date of the internal calendar.

Parameters:

<Year> Year, to be entered as a four-digit number (including the century and millennium information)

<Month> Month, 1 (January) to 12 (December)

<Day> Day, 1 to the maximum number of days in the specified month
*RST: does not affect the date settings

Example: `SYSTem:DATE?`
Returned value: `2011,09,13`

Usage: SCPI confirmed

SYSTem:TIME? <Hour>, <Minute>, <Second>

Returns the UTC (Universal Time Coordinated) of the internal clock. To define the current local time, use the time zone setting of the operating system (SETUP > "Time, date")

Example: `SYSTem:TIME?`
Returned value: `15,09,20`. UTC is 15:09:20.

Usage: Query only
SCPI confirmed

SYSTem:DEVice:ID?

Returns the instrument ID - that is the material number and the serial number.

Return values:

<ID> String containing the material number and the serial number

Example: `1316.1000K24-001122-jT`

Usage: Query only

DIAGnostic:SERVice:FWVersion?

Returns the firmware version that is currently installed on the instrument.

Return values:
<FirmwareVersion> Version string

Usage: Query only

DIAGnostic:SERVice:COMPutername <ComputerName>

The query returns the computer name that is currently defined. The computer name is required when configuring a network.

The setting command changes the computer name. The change takes effect after the next reboot of the computer.

Parameters:
<ComputerName> Name string

DIAGnostic:SERVice:PARTnumber <MaterialNumber>

Returns the material number of your instrument. This number is required to order a new option, and in case of service.

Parameters:
<MaterialNumber> Number string

DIAGnostic:SERVice:SERialnumber?

Returns the serial number of your instrument. This number is required to order a new option, and in case of service.

Return values:
<SerialNumber> Number string

Usage: Query only

DIAGnostic:SERVice:CHANnelcount?

Queries the number of available channels.

Return values:
<ChannelCount> Range: 0 to 4
Increment: 1
*RST: 0

Usage: Query only

Firmware/Software: V 2.00

SYSTem:VERSion?

Queries the SCPI version number to which the instrument complies. The instrument complies to the final SCPI version 1999.0.

Usage: Query only
SCPI confirmed

SYSTem:DFPRint [<Path>]

The device footprint contains the configuration of the instrument, installed modules, installed software and software licenses. This information is written in the device footprint xml file might be useful in case of maintenance or support request.

The query returns the information as block data. The setting command saves the device footprint xml file in the specified path.

It is also possible to access the device footprint xml file via the instrument's web browser. Therefore, the directory containing the xml file must be enabled for sharing.

Setting parameters:

<Path> String parameter, specifying the target path of the footprint file.

Return values:

<DeviceFootprint> Content of the device footprint xml file as block data

SYSTem:ERRor:ALL?

Queries the error/event queue for all unread items and removes them from the queue. The response is a comma-separated list of error number and a short description of the error in FIFO order.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:

<Error> List of: Error/event_number,"Error/event_description>[;Device-dependent info]"
If the queue is empty, the response is 0,"No error".

Usage: Query only
SCPI confirmed

SYSTem:ERRor[:NEXT]?

Queries the error/event queue for the oldest item and removes it from the queue. The response consists of an error number and a short description of the error.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:

<Error> Error/event_number,"Error/event_description>[;Device-dependent info]"
If the queue is empty, the response is 0,"No error".

Usage: Query only
SCPI confirmed

SYSTem:ERRor:CODE:ALL?

Queries the error/event queue for all unread items and removes them from the queue. The response is a comma-separated list of error numbers in FIFO order.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:

<Error> If the queue is empty, the response is 0.

Usage: Query only
SCPI confirmed

SYSTem:ERRor:CODE[:NEXT]?

Queries the error/event queue for the oldest item and removes it from the queue. The response is the error number.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:

<Error> If the queue is empty, the response is 0.

Usage: Query only
SCPI confirmed

SYSTem:ERRor:COUNt?

Queries the number of entries in the error queue.

Return values:

<Count> If the queue is empty, the response is 0.

Usage: Query only
SCPI confirmed

17.7.2 Display Settings

- Signal Colors / Persistence....1053
- Color Tables....1055
- SmartGrid....1057

- Diagram Layout....1060
- Waveform Labels....1064
- Clear Results....1068

17.7.2.1 Signal Colors / Persistence

DISPlay:PERSistence[:STATe]....1053
DISPlay:PERSistence:INFinite....1053
DISPlay:PERSistence:TIME....1053
DISPlay:PERSistence:RESet....1054
DISPlay:INTensity....1054
DISPlay:DIAGram:STYLe....1054
DISPlay:COLor:SIGNal<m>:COLor....1054
DISPlay:COLor:SIGNal<m>:ASSign....1055
DISPlay:COLor:SIGNal<m>:USE....1055

DISPlay:PERSistence[:STATe] <State>

If enabled, each new data point in the diagram area remains on the screen for the duration defined using `DISPlay:PERSistence:TIME`, or as long as `DISPlay:PERSistence:INFinite` is enabled.

If disabled, the signal value is only displayed as long as it actually occurs.

Parameters:

<State> ON | OFF

*RST: ON

Usage: Asynchronous command

DISPlay:PERSistence:INFinite <State>

If persistence is enabled (`DISPlay:PERSistence[:STATe]`), each new data point in the diagram area remains on the screen infinitely until this command is set to "OFF".

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

DISPlay:PERSistence:TIME <Time>

If persistence is enabled (`DISPlay:PERSistence[:STATe]`), each new data point in the diagram area remains on the screen for the duration defined here.

Parameters:

<Time> Range: 0.05 to 50

Increment: 0.05

*RST: 0.05

Default unit: s

Usage: Asynchronous command

DISPlay:PERSistence:RESet

Resets the display, removing persistent values.

Usage: Event
Asynchronous command

DISPlay:INTensity <Intensity>

This value determines the strength of the waveform line in the diagram. Enter a percentage between 0 (not visible) and 100% (very strong).

The exact mapping of the cumulative value occurences according to the assigned color table is guaranteed only if the intensity is set to 50% (default). All other intensity values falsify the mapping but may improve the visibility of the signal.

See also: Chapter 3.4.3.2, "Changing Waveform Colors", on page 115.

Parameters:
<Intensity> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

Usage: Asynchronous command

DISPlay:DIAGram:STYLe <Style>

Select the style in which the waveform is displayed.

Parameters:
<Style> VECTors | DOTS

VECTors
The individual data points are connected by a line.

DOTS
Only the individual data points are displayed.

*RST: VECTors

Usage: Asynchronous command

DISPlay:COLor:SIGNal<m>:COLor <Value>

Suffix:
<m> 1...118
Waveform number, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value>

Decimal value of the ARGB color. Use the color dialog box on the instrument to get the hex value of the color, and convert the hex value to a decimal value.
0 is fully transparent black.
4278190080 (dec) = FF000000 (hex) is opaque black.
4294967295 (dec) = FFFFFFFF (hex) is opaque white.

DISPlay:COLor:SIGNal<m>:ASSign <ColorTable>

Assigns the color table to the specified signal.

Suffix:

<m>	1...118 Waveform number, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:

<ColorTable>	Color table name to be assigned to the signal.

DISPlay:COLor:SIGNal<m>:USE <State>

If enabled, the selected waveform is displayed according to its assigned color table.

If this option is disabled, the default color table is used, i.e. the intensity of the specific signal color varies according to the cumulative occurance of the values.

Suffix:

<m>	1...118 Waveform number, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:

<State>	ON | OFF

17.7.2.2 Color Tables

DISPlay:COLor:PALette:ADD 1055
DISPlay:COLor:PALette:REMove 1056
DISPlay:COLor:PALette:COUNt? 1056
DISPlay:COLor:PALette:POINt:ADD 1056
DISPlay:COLor:PALette:POINt:INSert 1056
DISPlay:COLor:PALette:POINt:REMove 1056
DISPlay:COLor:PALette:POINt[:VALue] 1056
DISPlay:COLor:PALette:POINt:COUNt? 1057

DISPlay:COLor:PALette:ADD <Name>

Adds a new color table with the specified name.

Setting parameters:

<Name>	color table

Usage: Setting only

DISPlay:COLor:PALette:REMove <Name>

Removes the specified color table.

Setting parameters:

<Name> color table

Usage: Setting only

DISPlay:COLor:PALette:COUNt?

Queries the number of configured color maps.

Usage: Query only

DISPlay:COLor:PALette:POINt:ADD <PaletteName>

Appends a new row at the end of the color table.

Setting parameters:

<PaletteName> color table

Usage: Setting only

DISPlay:COLor:PALette:POINt:INSert <PaletteName>, <PointIndex>

Inserts the entry at the specified index in the color table.

Setting parameters:

<PaletteName> color table

<PointIndex> row number in the color table

Usage: Setting only

DISPlay:COLor:PALette:POINt:REMove <PaletteName>, <PointIndex>

Removes the entry with the specified index from the color table.

Setting parameters:

<PaletteName> color table

<PointIndex> row number in the color table

Usage: Setting only

DISPlay:COLor:PALette:POINt[:VALue] <ColorTableName>, <Index>,<Position>, <Color>
DISPlay:COLor:PALette:POINt[:VALue]? <ColorTableName>, <Index>

Inserts a new entry or queries the specified entry in the specified color table.

Parameters:

<Position>	Cumulative occurance value Range: 0 to 100 Increment: 1 *RST: 50 Default unit: %
<Color>	ARGB value of the color to be used for the table entry. ARGB=<Opacity(alpha) value><red value><green value><blue value>, in hexadecimal or decimal format. Range: 0 to 4294967295 Increment: 1 *RST: 0

Parameters for setting and query:

<ColorTableName>	Color table to be edited
<Index>	Index (row number) of the new entry in the color table
Usage:	Asynchronous command

DISPlay:COLor:PALette:POINt:COUNt? <PaletteName>

Queries the number of entries in the color table.

Query parameters:

<PaletteName>	color table
Usage:	Query only

17.7.2.3 SmartGrid

The following `LAYout` commands configure the SmartGrid: add and remove diagrams, and assign signals to the diagrams. In manual operation, you configure the SmartGrid by drag&drop.

LAYout:ADD.. 1057
LAYout:REMove.. 1058
LAYout:SHOW... 1059
LAYout:SIGNal:ASSign.. 1059
LAYout:SIGNal:UNASsign.. 1059

LAYout:ADD <NodeName>, <ParentType>, <InsertBefore>, <FirstSource>, <DiagramName>

Adds a new diagram with a waveform on the screen, in relation to an existing diagram.

Setting parameters:

<NodeName>	String with the name of the existing diagram
<ParentType>	HORizontal | VERTical | TAB Position of the new diagram in relation to the existing one.

HORizontal
Besides the existing diagram

VERTical
Above or below the existing diagram

TAB
In a new tab in the existing diagram

<InsertBefore> ON | OFF

If on, the new diagram is inserted to the left (for HORizontal), above (for VERTical) or in a tab in front the existing diagram.
HOR, ON = left to the existing diagram, defined in <NodeName>
HOR, OFF = right to the existing diagram
VERT, ON = above the existing diagram
VERT, OFF = below the existing diagram

<FirstSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to be displayed in the new diagram, see Chapter 17.4.2, "Waveform Parameter", on page 1037.
Spectrum Analysis, option R&S RTE-K18: Spectrograms and timeline spectrums are automatically displayed in their own diagrams.

<DiagramName> String with the name of the new diagram.

Example:
```
LAYout:ADD 'Diagram2',TAB,ON,C4W1,'MyDiagram3'
```
Creates a new diagram 'MyDiagram3' with waveform C4W1 in a new tab that in front of 'Diagram2'.

Example: See Chapter 17.3.1.1, "Creating Diagrams (SmartGrid)", on page 1019.

Usage: Setting only
Asynchronous command

LAYout:REMove <DiagramName>

Closes the specified diagram. The waveforms are displayed as minimized waveforms in their signal icons on the signal bar.

Setting parameters:
<DiagramName> String with the name of the diagram

Usage: Setting only
Asynchronous command

LAYout:SHOW <DiagramName>

Selects the specified diagram.

Setting parameters:

<DiagramName> String with the name of the diagram

Usage: Setting only
Asynchronous command

LAYout:SIGNal:ASSign <DiagramName>, <Source>

Shows the specified waveform in the selected diagram.

Setting parameters:

<DiagramName> String with the diagram name

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to be assigned, see Chapter 17.4.2, "Waveform Parameter", on page 1037
Spectrum Analysis, option R&S RTE-K18: Spectrograms and timeline spectrums are automatically displayed in their own diagrams.

Usage: Setting only
Asynchronous command

LAYout:SIGNal:UNASsign <Source>

Removes the specified waveform from the diagram.

Setting parameters:

<Source> See `LAYout:SIGNal:ASSign` on page 1059 and Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Setting only
Asynchronous command

17.7.2.4 Diagram Layout

The following commands belong to the settings in "Display" > "Diagram Layout". To configure the SmartGrid, use the commands described in Chapter 17.7.2.3, "SmartGrid", on page 1057.

These settings are user-specific, they are *not* reset by [PRESET] and `*RST`. You can reset them to default values using [FILE] > "Save/Recall > User defined preset > Factory defaults" or using the `SYSTem:PRESet` command.

DISPlay:DIAGram:REName 1060
DISPlay:DIAGram:GRID 1060
DISPlay:DIAGram:CROSshair 1061
DISPlay:DIAGram:FINegrid 1061
DISPlay:DIAGram:LABels 1061
DISPlay:DIAGram:TITLe 1061
DISPlay:DIAGram:YFIXed 1061
DISPlay:DIAGram:XFIXed 1062
DISPlay:GATE:TRANsparency 1062
DISPlay:RESultboxes:DEFaultpos 1062
DISPlay:SIGBar[:STATe] 1062
DISPlay:SIGBar:POSition 1063
DISPlay:SIGBar:HIDE[:AUTO] 1063
DISPlay:SIGBar:HIDE:TIME 1063
DISPlay:SIGBar:HIDE:HEAD 1063
DISPlay:SIGBar:HIDE:TRANsparency 1063
DISPlay:SIGBar:COLor:BORDer 1064
DISPlay:SIGBar:COLor:FILL 1064

DISPlay:DIAGram:REName <DiagramName>, <NewName>

Changes the diagram name, which is displayed on the diagram tab.

Setting parameters:

<DiagramName> String with the existing diagram name.

<NewName> String with the new diagram name.

Example: `:DISPlay:DIAGram:REName "Diagram1",'Test A'`

Usage: Setting only

DISPlay:DIAGram:GRID <Show>

If enabled, a grid is displayed in the diagram area.

Parameters:

<Show> ON | OFF

Usage: Asynchronous command

DISPlay:DIAGram:CROSshair <Crosshair>

If selected, a crosshair is displayed in the diagram area. A crosshair allows you to select a specific data point by its co-ordinates.

Parameters:

<Crosshair>	ON | OFF
Usage:	Asynchronous command

DISPlay:DIAGram:FINegrid <ShowFineScale>

If ON, the crosshair is displayed as a ruler with scale markers. If OFF, the crosshair is shown as dashed lines.

Parameters:

<ShowFineScale>	ON | OFF
Usage:	Asynchronous command

DISPlay:DIAGram:LABels <ShowLabels>

If enabled, labels mark values on the x- and y-axes in specified intervals in the diagram.

Parameters:

<ShowLabels>	ON | OFF
Usage:	Asynchronous command

DISPlay:DIAGram:TITLe <DiagTitleSt>

If enabled, the tab titles of all diagrams are displayed: "Diagram1", "Diagram2" ...

If disabled, the tab titles are not shown except for those in a tabbed diagram. In tabbed diagrams, the tab titles are required to change the tabs.

Parameters:

<DiagTitleSt>	ON | OFF
Usage:	Asynchronous command

DISPlay:DIAGram:YFIXed <YGridFixed>

If enabled, the horizontal grid lines remain in their position when the position of the curve is changed. Only the values at the grid lines are adapted. This reflects the behavior of traditional oscilloscopes.

Parameters:

<YGridFixed>	ON | OFF
Usage:	Asynchronous command

DISPlay:DIAGram:XFIXed <XGridFixed>

If enabled, the vertical grid lines remain in their position when the horizontal position is changed. Only the values at the grid lines are adapted.

Parameters:

<XGridFixed> ON | OFF

*RST: OFF

Usage: Asynchronous command

DISPlay:GATE:TRANsparency <Transparency>

Sets the transparency of the area that is defined as measurement or search gate.

Parameters:

<Transparency> Range: 0 to 100

Increment: 1

*RST: 43

Default unit: %

Usage: Asynchronous command

Firmware/Software: FW 3.20

DISPlay:RESultboxes:DEFaultpos <State>

Defines where a new result box opens.

Parameters:

<State> PREV | FLOA

PREV

Preview: The result box opens as a minimized result icon on the signal bar. It shows only two colums and a few rows of the results.

FLOA

Floating: The result box opens as a box similar to a dialog box in front of the diagrams. It can be moved and shows all results.

Usage: Asynchronous command

DISPlay:SIGBar[:STATe] <State>

If enabled, the signal bar is displayed.

Parameters:

<State> ON | OFF

Usage: Asynchronous command

DISPlay:SIGBar:POSition <Position>

The signal bar can be placed vertically at the right (default position) or at the left, or horizontally at the top, bottom or center of the diagram to ensure best visibility of the waveforms.

Parameters:

<Position> LEFT | RIGHt

Usage: Asynchronous command

DISPlay:SIGBar:HIDE[:AUTO] <AutoHide>

If enabled, the signal bar disappears automatically after some time, similar to the Windows task bar. With the commands `DISPlay:SIGBar:HIDE:TIME` and `DISPlay:SIGBar:HIDE:TRANsparency`, you can define when and how the signal bar hides.

The signal bar reappears if you tap it, or if an action changes the content of the bar.

Parameters:

<AutoHide> ON | OFF

Usage: Asynchronous command

DISPlay:SIGBar:HIDE:TIME <AutoHideTime>

Sets the time when the signal bar is faded out if `DISPlay:SIGBar:HIDE[:AUTO]` is "ON".

Parameters:

<AutoHideTime> Range: 0.03 to 86.4E+3
Increment: 0.5
Default unit: s

Usage: Asynchronous command

DISPlay:SIGBar:HIDE:HEAD <HideHeadAlso>

If enabled, the "Auto hide" function hides also the horizontal and trigger label at the top of the signal bar.

Parameters:

<HideHeadAlso> ON | OFF

Usage: Asynchronous command

DISPlay:SIGBar:HIDE:TRANsparency <HidingTransp>

Sets the transparency of the signal bar when the signal bar is faded out with `DISPlay:SIGBar:HIDE[:AUTO]`.

Parameters:

<HidingTransp>	Range: 20 to 70 Increment: 5 Default unit: %

Usage: Asynchronous command

DISPlay:SIGBar:COLor:BORDer <BorderColor>

Defines the color of the signal bar border.

See also: "To change the colors" on page 79.

Parameters:

<BorderColor>	ARGB color value Range: 0 to 4294967295 Increment: 1

Usage: Asynchronous command

DISPlay:SIGBar:COLor:FILL <FillColor>

Define the fill color of the signal bar.

See also: "To change the colors" on page 79.

Parameters:

<FillColor>	ARGB color value Range: 0 to 4294967295 Increment: 1

Usage: Asynchronous command

17.7.2.5 Waveform Labels

To create a waveform label, use `DISPlay:SIGNal:LABel:ADD`. Using the other `DISP:SIGN:LAB:...` commands, you can query the text and position of a label, and modify the initial settings. The <LabelID> and <Source> parameters identify each label uniquely. Note that it is not possible to query the <LabelID>, or to read it on the user interface.

DISPlay:SIGNal:LABel:ADD........ 1065
DISPlay:SIGNal:LABel:REMove........ 1066
DISPlay:SIGNal:LABel:TEXT........ 1066
DISPlay:SIGNal:LABel:POSMode........ 1066
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition........ 1067
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition........ 1067
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition........ 1068
DISPlay:SIGNal:LABel:VERTical:RELative:POSition........ 1068
DISPlay:SIGNal:LABel:FONTsize........ 1068

DISPlay:SIGNal:LABel:ADD <LabelID>, <Source>, <LabelText>, <PositionMode>, <XPositon>, <YPositon>

Creates a new waveform label for the specified source waveform.

Setting parameters:

<LabelID>
String with the label identifier. The <LabelID> and <Source> parameters identify each label uniquely, so the label ID must be unique for the given waveform. Note the <LabelID> because it is not possible to query it, or to read it on the user interface.

<Source>
C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | QUICk | QUICK | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1 | SG2 | SG3 | SG4 | SG5 | SG6 | SG7 | SG8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to that the label belongs, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

<LabelText>
String with the label text that is shown on the display

<PositionMode>
ABS | REL

ABS
Position in time and voltage values, or in other units depending on the waveform character. Absolute positions move with the waveform display when the scales, the vertical position or offset, or the reference point are changed.

REL
Fixed label position in percent of the screen counting from the upper left corner.

<XPositon>
Horizontal position of the label text. Values, range and unit depend on the position mode, the waveform position and scaling. For relative position mode, the range is 0 to 100 %.

<YPositon>
Vertical position of the label text. Values, range and unit depend on the position mode, the waveform position and scaling. For relative position mode, the range is 0 to 100 %.

Example:

```
DISPlay:SIGNal:LABel:ADD 'Label1', C1W1, 'Label on C1', REL, 20, 20
```

Adds the label text 'Label on C1' to the channel1 waveform at relative position 20% from the upper left corner of the screen. The label ID is 'Label1'.

Example: `DISPlay:SIGNal:LABel:ADD 'Label2', C2W1, 'Label on C2', ABS, 10e-09, 0.1`
Adds the label text 'Label on C2' to the channel2 waveform at absolute position 10 ns and 0.1 V. The label ID is 'Label2'.

Usage: Setting only
Asynchronous command

DISPlay:SIGNal:LABel:REMove <LabelID>, <Source>

Deletes the specifies waveform label.

Setting parameters:
<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Example: `DISPlay:SIGNal:LABel:REMove 'Label1', C1W1`

Usage: Setting only
Asynchronous command

DISPlay:SIGNal:LABel:TEXT <LabelID>, <Source>, <LabelText>
DISPlay:SIGNal:LABel:TEXT? <LabelID>, <Source>

Modifies or queries the text of the specified label.

Parameters:
<LabelText> String with the label text that is shown

Parameters for setting and query:
<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Usage: Asynchronous command

DISPlay:SIGNal:LABel:POSMode <Source>, <PositionMode>
DISPlay:SIGNal:LABel:POSMode? <Source>

Modifies or queries the position mode: either relative to the diagram or with absolute values according to the units of the waveform. The position mode applies to all labels of the selected source. For different sources, different position modes can be selected.

Parameters:
<PositionMode> ABS | REL

ABS
Position in time and voltage values, or in other units depending on the waveform character. Absolute positions move with the waveform display when the scales, the vertical position or offset, or the reference point are changed.
Use `DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition` and `DISPlay:SIGNal:LABel:HORizontal:RELative:POSition` to set the position.

REL
Fixed label position in percent of the screen counting from the upper left corner.
Use `DISPlay:SIGNal:LABel:HORizontal:RELative:POSition` and `DISPlay:SIGNal:LABel:VERTical:RELative:POSition` to set the position.

Parameters for setting and query:

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Usage: Asynchronous command

DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition? <LabelID>, <Source>
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition? <LabelID>, <Source>

Modifies or queries the absolute horizontal and vertical positiions of the specified label if `DISPlay:SIGNal:LABel:POSMode` is set to `ABS`.

Parameters:

<Position> Range: Depends on waveform position and scaling
Default unit: s and V, or in other units depending on the waveform character

Parameters for setting and query:

<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Example:

```
:DISPlay:SIGNal:LABel:POSMode C2W1, ABS
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition 'Label1', C2W1, 5e-09
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition 'Label1', C2W1, -0.1
```

Move the label to 5 ns and -0.1 V.

DISPlay:SIGNal:LABel:HORizontal:RELative:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition? <LabelID>, <Source>
DISPlay:SIGNal:LABel:VERTical:RELative:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:VERTical:RELative:POSition? <LabelID>, <Source>

Modifies or queries the relative horizontal and vertical positiions of the specified label if `DISPlay:SIGNal:LABel:POSMode` is set to `REL`.

Parameters:

<Position> Position in percent of the screen counting from the upper left corner.

Range: 0 to 100
Default unit: %

Parameters for setting and query:

<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Example:

```
:DISPlay:SIGNal:LABel:POSMode C2W1, REL
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition 'Label1', C2W1, 30
DISPlay:SIGNal:LABel:VERTical:RELative:POSition 'Label1', C2W1, 70
```

Move the label to new relative position: horizontal at 30 % and vertical at 70 % of the screen.

DISPlay:SIGNal:LABel:FONTsize <Fontsize>

Sets the font size of waveform labels.

Parameters:

<Fontsize> Range: 7 to 30
Increment: 1
*RST: 15

Usage: Asynchronous command

17.7.2.6 Clear Results

DISPlay:CLR

Deletes all measurement results including long term measurement and statistics, all waveforms, and the history.

Usage: Event
Asynchronous command

17.7.3 External Application

The commands configure an external application that can be started at various events, for example, trigger event, or mask test violation.

EXECutable:NAME... 1069
EXECutable:PARameter.. 1069
EXECutable:WDIRectory... 1069

EXECutable:NAME <ApplicationPath>

Sets the path to the application executable.

Parameters:
<ApplicationPath> String parameter containing path, filename, and file extension

Example:
```
EXECutable:NAME 'C:\Program
Files\Wireshark\Wireshark.exe'
TRIGger:EVENt:RUNexec TRIGger
```

Usage: Asynchronous command

EXECutable:PARameter <AppParameters>

Sets optional parameters for the external executable.

Parameters:
<AppParameters> String parameter

Usage: Asynchronous command

EXECutable:WDIRectory <WorkDirectory>

Sets the working directory for the executable.

Parameters:
<WorkDirectory> String parameter

Usage: Asynchronous command

17.8 Acquisition and Setup

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.

- Chapter 17.3, "Programming Examples", on page 1018

- Starting and Stopping Acquisition....1070
- Time Base....1071
- Acquisition....1076
- Fast Segmentation....1078
- Vertical....1079
- Waveform Data....1084
- Probes....1086
- R&S RT-ZVC Probe....1106
- Differential Signals....1120
- Digital Filter....1123
- Skew....1124
- High Definition Mode....1125
- Reference Clock....1126

17.8.1 Starting and Stopping Acquisition

RUNContinous....1070
RUN....1070
RUNSingle....1070
SINGle....1070
STOP....1070

RUNContinous
RUN

Starts the continuous acquisition.

Usage: Event
Asynchronous command

RUNSingle
SINGle

Starts a defined number of acquisition cycles. The number of cycles is set with `ACQuire:COUNt`.

Usage: Event
Asynchronous command

STOP

Stops the running acquistion.

Usage: Event
Asynchronous command

17.8.2 Time Base

TIMebase:SCALe....1071
TIMebase:RANGe....1071
TIMebase:DIVisions?....1072
TIMebase:HORizontal:POSition....1072
TIMebase:REFerence....1072
TRIGger<m>:OFFSet:LIMited....1072
AUToscale....1073
ACQuire:POINts:AUTO....1073
ACQuire:POINts:AADJust....1073
ACQuire:POINts:MAXimum....1074
ACQuire:POINts:ARATe?....1074
ACQuire:SRATe....1074
ACQuire:SRReal....1074
ACQuire:RESolution....1075
ACQuire:POINts[:VALue]....1075
TIMebase:ROLL:ENABle....1075
TIMebase:ROLL:STATe?....1076
TIMebase:ROLL:MTIMe....1076

TIMebase:SCALe <TimeScale>

Sets the horizontal scale - the time per division on the x-axis - for all channel and math waveforms.

The setting accuracy depends on the current resolution (sample rate).

- No interpolation:
  The resolution is an integer multiple of the ADC sample rate.
- With interpolation:
  Any value for the horizontal scale can be set.

Parameters:

<TimeScale> Range: 25E-12 to 10000 (RTO, RTP) | 5000 (RTE)
Increment: 1E-12
*RST: 10E-9
Default unit: s/div

Usage: Asynchronous command

TIMebase:RANGe <AcquisitionTime>

Defines the time of one acquisition, that is the time across the 10 divisions of the diagram: *TimeScale*10*.

Parameters:

<AcquisitionTime> Range: 250E-12 to 100E+3 (RTO, RTP) | 50E+3 (RTE)
Increment: 1E-12
*RST: 0.5
Default unit: s

Usage: Asynchronous command

TIMebase:DIVisions?

Queries the number of horizontal divisions on the screen. The number cannot be changed.

Return values:

<HorizDivCnt>
Range: 4 to 20
Increment: 2
*RST: 10

Usage: Query only
Asynchronous command

TIMebase:HORizontal:POSition <RescalCenterTime>

Defines the time distance between the reference point and the trigger point (the zero point of the diagram). The reference point marks the rescaling center of the time scale.

Parameters:

<RescalCenterTime>
Range: -100E+24 to 100E+24
Increment: 1E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

TIMebase:REFerence <RescaleCtrPos>

Sets the position of the reference point in % of the screen. The reference point marks the rescaling center of the time scale. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compresses to both sides of the reference point.

Parameters:

<RescaleCtrPos>
Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

Usage: Asynchronous command

TRIGger<m>:OFFSet:LIMited <State>

If ON, the horizontal position cannot be set outside the visible waveform diagram.

See also: TIMebase:HORizontal:POSition on page 1072

Suffix:

<m>
1..3
The numeric suffix is irrelevant.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

AUToscale

Performs an autoset process: analyzes the enabled channel signals, and obtains appropriate horizontal, vertical, and trigger settings to display stable waveforms.

Rohde & Schwarz does not recommend using the autoset in remote control. To adjust the oscilloscope remotely, especially for automated testing applications, use the remote commands that adjust the horizontal, vertical and trigger settings."

Usage: Event
Asynchronous command

ACQuire:POINts:AUTO <ReclgthManual>

Selection to keep constant either the resolution or the record length when you adjust the time scale (`TIMebase:SCALe`) or acquisition time (`TIMebase:RANGe`).

Parameters:

<ReclgthManual> RESolution | RECLength

RESolution
Resolution is kept constant. Set the required resolution value with `ACQuire:RESolution`.

RECLength
The record length is kept constant. Set the required record length value with `ACQuire:POINts[:VALue]`.

*RST: RESolution

Usage: Asynchronous command

ACQuire:POINts:AADJust <AutoAdjust>

Prevents undersampling and ensures a sufficient resolution to acquire the correct waveform if the time scale is changed. The setting takes effect if the changed parameter - resolution or record length - reaches a limit. The instrument automatically keeps this parameter constant at its limit, and changes the other parameter regardless of the `ACQuire:POINts:AUTO` setting.

Parameters:

<AutoAdjust> ON | OFF

*RST: ON

Usage: Asynchronous command

ACQuire:POINts:MAXimum <ReclgthLim>

Sets a limit for the record length to prevent very large records. This value only takes effect if a constant resolution is selected with ACQuire:POINts:AUTO. If you increase the time scale, the resolution remains constant and the record length increases until the limit is reached. Further increase of the time scale changes the resolution and keeps the record length limit.

Parameters:

<ReclgthLim>	Range:	1000 to 800 MSa. The actual maximum can be lower depending on the installed options, number of active channels, measurements and math waveforms.
	Increment:	2
	*RST:	10E+6
	Default unit:	Sa

Usage: Asynchronous command

ACQuire:POINts:ARATe?

Retrieves the sample rate of the ADC, that is the number of points that are sampled by the ADC in one second.

Return values:

<ADCSampleRate>	Range:	5 Gsample/s
	Default unit:	Sa/s

Usage: Query only
Asynchronous command

ACQuire:SRATe <SampleRate>

Defines the sample rate, that is the number of recorded waveform samples per second.

Parameters:

<SampleRate>	Range:	5 Gsample/s
	Increment:	1
	*RST:	5 Gsample/s
	Default unit:	Sa/s

Usage: Asynchronous command

ACQuire:SRReal <RealSampleRate>

Sets the number of captured waveform points per second. It considers the samples of the ADC, and the reduction of waveform points by decimation.

If interpolation is not active, the sample rate is the reciprocal value of the resolution and thus also depends on the acquisition time and the record length.

If interpolation is active, the sample rate is limited to the ADC sample rate.

Parameters:
<RealSampleRate> Range: 5 Gsample/s
Increment: 1
*RST: 5 Gsample/s
Default unit: Sa/s

Usage: Asynchronous command

Firmware/Software: FW 3.40

ACQuire:RESolution <Resolution>

Indicates the time between two waveform points in the record.

Parameters:
<Resolution> A fine resolution with low values produces a more precise waveform record.
Range: 1E-15 to 0.5
Increment: 10E-12
*RST: 100E-12
Default unit: s

Usage: Asynchronous command

ACQuire:POINts[:VALue] <RecordLength>

Indicates the record length, the number of recorded waveform points that build the waveform across the acquisition time. `[:VALue]` can be omitted.

Parameters:
<RecordLength> Number of recorded waveform points.
Range: 1000 to 1000000000
Increment: 2
*RST: 1000
Default unit: Sa

Usage: Asynchronous command

TIMebase:ROLL:ENABle <Mode>

Activates the automatic roll mode.

Parameters:
<Mode> AUTO | OFF
AUTO: the instrument activates the roll mode under specific conditions.
See also "Roll mode" on page 142.
*RST: AUTO

Usage: Asynchronous command

TIMebase:ROLL:STATe?

Returns the status of the roll mode.

Return values:

<State> ON | OFF

*RST: OFF

Usage: Query only
Asynchronous command

TIMebase:ROLL:MTIMe <MinHorizGain>

The roll mode is enabled automatically if the acquisition time exceeds the given value, and if `TIMebase:ROLL:ENABle` is set to AUTO.

Parameters:

<MinHorizGain> Treshold value for roll mode enabling.

Range: 1 to 600
Increment: 1
*RST: 10
Default unit: s

Usage: Asynchronous command

17.8.3 Acquisition

CHANnel<m>[:WAVeform<n>][:STATe]....1076
CHANnel<m>[:WAVeform<n>]:TYPE....1077
CHANnel<m>[:WAVeform<n>]:ARIThmetics....1077
ACQuire:COUNt....1078
ACQuire:ARESet:IMMediate....1078

CHANnel<m>[:WAVeform<n>][:STATe] <State>

Activates or deactivates a waveform. `[:STATe]` can be omitted.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:

<State> ON | OFF

*RST: OFF

Example: `CHAN2 ON`
Activates analog channel 2.

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:TYPE <DecimationMode>

Selects the method to reduce the data stream of the ADC to a stream of waveform points with lower sample rate.

See also:"Decimation" on page 143.

Suffix:

<m>	1..4 Selects the input channel.
<n>	1..3 [:WAVeform<n>] is irrelevant, omit it.

Parameters:

<DecimationMode> SAMPle | PDETect | HRESolution | RMS

SAMPle
One of n samples in a sample interval of the ADC is recorded as waveform point.

PDETect
Peak Detect: the minimum and the maximum of n samples in a sample interval are recorded as waveform points.

HRESolution
High resolution: The average of n sample points is recorded as waveform point.

RMS
The waveform point is the root mean square of n sample values.

*RST: SAMPle

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:ARIThmetics <Arithmetics>

Selects the method to build the resulting waveform from several consecutive acquisitions of the signal. To define the number of acquisitions, use `ACQuire:COUNt`.

See also: "Arithmetic" on page 144.

Suffix:

<m>	1..4 Selects the input channel.
<n>	1..3 [:WAVeform<n>] is irrelevant, omit it.

Parameters:

<Arithmetics> OFF | ENVelope | AVERage

OFF
The data of the current acquisition is recorded according to the decimation settings.

ENVelope
Detects the minimum and maximum values in an sample interval over a number of acquisitions. To define the reset method, use ...

AVERage
Calculates the average from the data of the current acquisition and a number of acquisitions before.

*RST: OFF

Usage: Asynchronous command

ACQuire:COUNt <MaxAcqCnt>

The acquisition and average count has a double effect:

- It sets the number of waveforms acquired with `RUNSingle`.
- It defines the number of waveforms used to calculate the average waveform.

Parameters:
<MaxAcqCnt>
Range: 1 to 16777215
Increment: 10
*RST: 1

Usage: Asynchronous command

ACQuire:ARESet:IMMediate

Forces the immediate restart of the envelope and average calculation for all wave-forms.

Usage: Event
Asynchronous command

17.8.4 Fast Segmentation

ACQuire:SEGMented:STATe....................1078
ACQuire:SEGMented:MAX....................1079
ACQuire:SEGMented:AUToreplay....................1079

ACQuire:SEGMented:STATe <State>

Switches the fast segmentation mode on and off.

See also: Chapter 4.2.3, "Fast Segmentation", on page 145.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

ACQuire:SEGMented:MAX <MaxAcquisitions>

The number of acquisitions in a fast segmentation acquisition series depends on the record length.

Parameters:

<MaxAcquisitions> ON | OFF

ON
Acquires the maximum possible number of acquisitions in a series.

OFF
Acquires the number of acquisitions defined using `ACQuire:COUNt`.

*RST: OFF

Usage: Asynchronous command

ACQuire:SEGMented:AUToreplay <ReplayAfterAcq>

If enabled, the instrument starts processing and displaying the data as soon as the acquisition series is captured completely. Depending on the number of acquisitions, it may take some time until the acquisition series is displayed. If the setting is disabled, the instrument only captures the data and stores it in the sample memory.

Parameters:

<ReplayAfterAcq> ON | OFF

*RST: ON

Usage: Asynchronous command

17.8.5 Vertical

CHANnel<m>:STATe ... 1079
CHANnel<m>:COUPling ... 1080
CHANnel<m>:GND ... 1080
CHANnel<m>:SCALe ... 1080
CHANnel<m>:RANGe ... 1081
CHANnel<m>:POSition ... 1081
CHANnel<m>:OFFSet ... 1082
CHANnel<m>:INVert ... 1082
CHANnel<m>:BANDwidth ... 1083
CHANnel<m>:CPLing ... 1083
CHANnel<m>:IMPedance ... 1083
CHANnel<m>:OVERload ... 1084

CHANnel<m>:STATe <State>

Switches the channel signal on or off.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>:COUPling <Value>

Selects the connection of the indicated channel signal.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Value> DC | DCLimit | AC

DC
Direct connection with 50 Ω termination, passes both DC and AC components.

DCLimit
Direct connection with 1 MΩ termination, passes both DC and AC components.

AC
Connection with 1 MΩ termination, removes DC and very low-frequency components.

CHANnel<m>:GND <State>

Connects the signal to the ground.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

CHANnel<m>:SCALe <Scale>

Sets the vertical scale for the indicated channel.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:

<Scale> Scale value, given in Volts per division.

Range: Depends on attenuation factors, coupling, and instrument model, see below.
Increment: Depends on vertical and probe settings
*RST: 0.05
Default unit: V/div

With 1:1 probe, external attenuation = 1, and 50 Ω input coupling, the vertical scale (input sensitivity) is minimum 1 mV/div (RTE 1317.2500) or 0.5 mV/div (RTE 1326.2000) to maximum 1 V/div. For 1 MΩ input coupling, the upper limit is 10 V/div. If the probe and/or external attenuation is changed, multiply the values by the attenuation factors to get the actual scale range.

Usage: Asynchronous command

CHANnel<m>:RANGe <Range>

Sets the voltage range across the 10 vertical divisions of the diagram. Use the command alternativly instead of `CHANnel<m>:SCALe`.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Range> Voltage range value

Range: Dependencies see below.
Increment: Depends on vertical and probe settings
*RST: 0.5
Default unit: V/div

Depends on attenuation factors and coupling. With 1:1 probe, external attenuation = 1 and 50 Ω input coupling, the range is 10 mV to 10 V. For 1 MΩ input coupling, it is 10 mV to 100 V. If the probe and/or external attenuation is changed, multiply the range values by the attenuation factors.

Usage: Asynchronous command

CHANnel<m>:POSition <Position>

Sets the vertical position of the indicated channel as a graphical value.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Position> Positive values move the waveform up, negative values move it down.

Range: -5 to 5
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CHANnel<m>:OFFSet <Offset>

The offset voltage is subtracted to correct an offset-affected signal. The offset of a signal is determined and set by the autoset procedure.

See also: "Offset" on page 149

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Offset> Negative values move the waveform up, positive values move it down.

Range: Depends on attenuation factors, input coupling, and the offset compensation range of active probes. The nominal offset range for 1:1 attenuation and probe offset compensation = 0 is specified in the data sheet.
Increment: Depends on vertical and probe settings
*RST: 0
Default unit: V

Usage: Asynchronous command

CHANnel<m>:INVert <InvertChannel>

Turns the inversion of the signal amplitude on or off. To invert means to reflect the voltage values of all signal components against the ground level. If the inverted channel is the trigger source, the instrument triggers on the inverted signal.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<InvertChannel> ON | OFF

*RST: OFF

Usage: Asynchronous command

Firmware/Software: FW 3.30

CHANnel<m>:BANDwidth <BandwidthLimit>

Selects the bandwidth limit for the indicated channel.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<BandwidthLimit> FULL | B200 | B20

FULL
Use full bandwidth.

B200 | B20
Limit to 200 MHz or 20 MHz.

*RST: FULL

Usage: Asynchronous command

CHANnel<m>:CPLing <Channel>, <State>
CHANnel<m>:CPLing? <Channel>

Sets the vertical settings of the coupled channel to the values of the active channel.

Suffix:

<m> 1..4
Active channel

Parameters:

<State> OFF | ON

Parameters for setting and query:

<Channel> CHAN1 | C1 | CHAN2 | C2 | CHAN3 | C2 | CHAN4 | C4

CHAN1 = C1, CHAN2 = C2, CHAN3 = C2, CHAN4 = C4
Input channel that takes the vertical settings of CHANnel<m>.

Usage: Asynchronous command

CHANnel<m>:IMPedance <Impedance>

Sets the impedance of the channel for power calculations and measurements.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Impedance> Range: 0.1 to 100E+3
Increment: 1
*RST: 50
Default unit: Ohm

Usage: Asynchronous command

CHANnel<m>:OVERload <Overload>

Retrieves the overload status of the specified channel from the status bit. When the overload problem is solved, the command resets the status bit.

The overload status is returned asynchronously. Therefore, it is not possible to assign an overload to a specific acquisition during continuous acquisition. This assignment is only possible for a single acquisition.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Overload> ON | OFF
Use OFF to reset the overload status bit.
*RST: OFF

Example:
```
ACQuire:COUNt 1
SINGle
CHANnel2:OVERload?
```
Queries the overload status of channel 2 after a single acquisition.
```
CHANnel2:OVERload OFF
```
Resets the overload status bit.

Usage: Asynchronous command

17.8.6 Waveform Data

To set the export data format, see FORMat[:DATA] on page 1045.

CHANnel<m>[:WAVeform<n>]:DATA:HEADer?.. 1084
CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?.. 1085

CHANnel<m>[:WAVeform<n>]:DATA:HEADer?

Returns the header of channel waveform data.

Table 17-1: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of one waveform	200000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of samples (record length).	1

If multichannel export is enabled, the number of returned samples is *Record length * Number of exported waveforms*. See also `EXPort:WAVeform:MULTichannel`.

Suffix:

<m>	1..4 Selects the input channel.
<n>	1..3 [:WAVeform<n>] is irrelevant, omit it.

Example:
```
CHAN1:WAV1:DATA:HEAD?
-9.477E-008,9.477E-008,200000,1
```

Usage: Query only

CHANnel<m>[:WAVeform<n>]:DATA[:VALues]? [<Offset>], [<Length>]

Returns the data of the channel waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Without parameters, the complete waveform is retrieved. Using the offset and length parameters, data can be retrieved in smaller portions, which makes the command faster.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values (usually voltage values), or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define which values are retrieved.

If multichannel export is active (`EXPort:WAVeform:MULTichannel`), the channel suffix is ignored. To select the channels to be exported, use `CHANnel<m>:EXPortstate`. The Y-values are written in interleaved order, for example, $YCh1_0$; $YCh2_0$; $YCh1_1$; $YCh2_1$... for a 2-channel instrument.

Suffix:

<m>	1..4 Selects the input channel.
<n>	1..3 [:WAVeform<n>] is irrelevant, omit it.

Query parameters:

[<Offset>]	Number of offset waveform points. Range: 0 to m. Limit: n + m >= record length
[<Length>]	Number of waveform points to be retrieved. Range: 1 to n. Limit: n + m >= record length

Return values:

<Data>	List of values according to the format and content settings.

Example: Retrieve the complete channel 1 waveform, only Y-values:

```
FORM ASC
EXP:WAV:INCX OFF
CHAN1:WAV1:DATA?
<-- -0.125000,-0.123016,-0.123016,-0.123016,-0.123016,-0.123016,...
```

Example: Retrieve the first 10 values of the waveform:

```
CHANnel:WAVeform:DATA:VALues? 0,10
<-- -0.10079051554203,-0.098814234137535,-0.098814234137535,
    -0.096837945282459,-0.094861663877964,-0.094861663877964,
    -0.092885382473469,-0.090909093618393,-0.090909093618393,
    -0.088932812213898
```

Example: Skip 5 samples and retrieve the next 5 samples:

```
CHANnel:WAVeform:DATA:VALues? 5,5
<-- -0.094861663877964,-0.092885382473469,-0.090909093618393,
    -0.090909093618393,-0.088932812213898
```

Usage: Query only

17.8.7 Probes

`TRPRobe:...` command are dedicated commands for the external trigger input.

- Common Probe Settings....1086
- Micro Button and R&S ProbeMeter....1089
- Passive Probes....1094
- Active Voltage Probes....1095
- Modular Probes....1098
- Predefined Probes....1101
- Current Probes....1102
- Probe Attributes....1103
- Probe Adapter....1105

17.8.7.1 Common Probe Settings

TRPRobe:SETup:STATe?....1087
PROBe<m>:SETup:STATe?....1087
TRPRobe:SETup:TYPE?....1087
PROBe<m>:SETup:TYPE?....1087
TRPRobe:SETup:NAME?....1087
PROBe<m>:SETup:NAME?....1087
TRPRobe:SETup:BANDwidth?....1087
PROBe<m>:SETup:BANDwidth?....1087
TRPRobe:SETup:ATTenuation[:AUTO]?....1088
PROBe<m>:SETup:ATTenuation[:AUTO]?....1088
PROBe<m>:SETup:OFFSet:AZERo....1088
PROBe<m>:SETup:OFFSet:USEautozero....1088
CHANnel<m>:EATScale....1089
CHANnel<m>:EATTenuation....1089

TRPRobe:SETup:STATe?
PROBe<m>:SETup:STATe?

Queries if the probe at the specified input channel is active (detected) or not active (not detected). To switch the probe on, use `CHANnel<m>:STATe`.

Suffix:

<m> 1..4

Return values:

<State> DETected | NDETected

*RST: NDETected

Usage: Query only
Asynchronous command

TRPRobe:SETup:TYPE?
PROBe<m>:SETup:TYPE?

Queries the type of the probe.

Suffix:

<m> 1..4
Selects the input channel.

Return values:

<Type> String containing one of the following values:
– None (no probe detected)
– Passive Probe
– active single-ended

Usage: Query only
Asynchronous command

TRPRobe:SETup:NAME?
PROBe<m>:SETup:NAME?

Queries the name of the probe.

Suffix:

<m> 1..4
Selects the input channel.

Return values:

<Name> Name string

Usage: Query only
Asynchronous command

TRPRobe:SETup:BANDwidth?
PROBe<m>:SETup:BANDwidth?

Queries the bandwidth of the probe.

Suffix:

<m> 1..4

Selects the input channel.

Return values:

<Bandwidth> Range: 10000 to 20E+9

*RST: 1E+9

Default unit: Hz

Usage: Query only

Asynchronous command

TRPRobe:SETup:ATTenuation[:AUTO]?
PROBe<m>:SETup:ATTenuation[:AUTO]?

Queries the attenuation of the probe.

Suffix:

<m> 1..4

Selects the input channel.

Return values:

<PrbAttMdAuto> Range: 1E-3 to 1000

*RST: 1

Default unit: V/V

Usage: Query only

Asynchronous command

PROBe<m>:SETup:OFFSet:AZERo

Performs an automatic correction of the zero error. If the DUT is ground-referenced, the AutoZero function can improve the measurement results.

See also: "Detect AutoZero, Use AutoZero" on page 157

Suffix:

<m> 1..4

Selects the input channel.

Usage: Event

Asynchronous command

PROBe<m>:SETup:OFFSet:USEautozero <AutoZeroOffset>

Includes the AutoZero offset in measurement results. The auto zero error is detected with `PROBe<m>:SETup:OFFSet:AZERo`.

Suffix:

<m> 1..4

Selects the input channel.

Parameters:

<AutoZeroOffset> ON | OFF

*RST: OFF

Usage: Asynchronous command

Firmware/Software: Version 2.70

CHANnel<m>:EATScale <ExtAttScl>

Sets the attenuation scale for an external divider.

Suffix:

<m> 1..4

Selects the input channel.

Parameters:

<ExtAttScl> LIN | LOG

*RST: LIN

Usage: Asynchronous command

CHANnel<m>:EATTenuation <ExtAtt>

Sets the attenuation of an external voltage divider that is part of the DUT before the measuring point. The external attenuation is included in the measurement, and the instrument shows the results that would be measured before the divider.

Suffix:

<m> 1..4

Selects the input channel.

Parameters:

<ExtAtt> Values and unit depend on the selected scale (`CHANnel<m>:EATScale`).

Range: Linear scale: 1E-3 to 1E+6, logarithmic scale: -60 dB to 120 dB

Increment: 0.01

*RST: 1

Usage: Asynchronous command

17.8.7.2 Micro Button and R&S ProbeMeter

PROBe<m>:SETup:MODE ... 1090
TRPRobe:SETup:DISPlaydiff ... 1091
PROBe<m>:SETup:DISPlaydiff ... 1091
TRPRobe:PMETer:VISibility ... 1091
PROBe<m>:PMETer:VISibility ... 1091
TRPRobe:PMETer:RESults:SINGle? ... 1092
PROBe<m>:PMETer:RESults:SINGle? ... 1092
TRPRobe:PMETer:RESults:COMMon? ... 1092

PROBe<m>:PMETer:RESults:COMMon?.. 1092
TRPRobe:PMETer:RESults:DIFFerential?.. 1092
PROBe<m>:PMETer:RESults:DIFFerential?.. 1092
TRPRobe:PMETer:RESults:NEGative?... 1093
PROBe<m>:PMETer:RESults:NEGative?.. 1093
TRPRobe:PMETer:RESults:POSitive?..1093
PROBe<m>:PMETer:RESults:POSitive?.. 1093

PROBe<m>:SETup:MODE <Mode>

Select the action that is started when you press the micro button on the probe head.

See also: "Micro button action" on page 159.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Mode> RCONtinuous | RSINgle | AUToset | AZERo | SEToffsettomean | PRINt | SITFile | NOACtion | FINDtriglevel | REPort | PROBemode | PRSetup

RCONtinuous
Run continuous: The acquisition is running as long as the probe button is pressed.

RSINgle
Run single: Starts a defined number of acquisitions (same as [RUN N× SINGLE] key).

AUTOSET
Starts the autoset procedure.

AZero
AutoZero: performs an automatic correction of the zero error.

SEToffsettomean
Set offset to mean: performs an automatic compensation for a DC component of the input signal.

PRINt
Prints the current display according to the printer set with `SYSTem:COMMunicate:PRINter:SELect<1..2>`.

SITFile
Save Image To File:
Directs the display image to a file. The `MMEMory:NAME` command defines the file name. The file format is defined with `HCOPy:DEVice<m>:LANGuage`.

NOACtion
Nothing is started on pressing the micro button.

FINDtriglevel
Sets the trigger level automatically to*0.5 * (MaxPeak – MinPeak)*. The function is not available for an external trigger source.

REPort
Creates and saves a report of the current results.

PROBemode
Only available for R&S RT-ZM probes. Changes the measurement mode of the probe.

PRSetup
Opens the "Probes Setup" dialog box.

*RST: RCONtinuous

Usage: Asynchronous command

TRPRobe:SETup:DISPlaydiff <DisplayDiff>
PROBe<m>:SETup:DISPlaydiff <DisplayDiff>

Selects the input voltages to be measured by the ProbeMeter of an R&S differential active probe.

See also: "Differential Active Probes" on page 137.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<DisplayDiff> DIFFerential | SINGleended

DIFFerential
Measures differential and common mode voltages

SINGleended
Measures the voltage between the positive/negative signal socket and the ground.

*RST: DIFFerential

Usage: Asynchronous command

TRPRobe:PMETer:VISibility <Visibility>
PROBe<m>:PMETer:VISibility <Visibility>

Activates the integrated R&S ProbeMeter of active R&S probes.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Visibility> ON | OFF

*RST: OFF

Usage: Asynchronous command

Firmware/Software: FW 2.25

TRPRobe:PMETer:RESults:SINGle?
PROBe<m>:PMETer:RESults:SINGle?

Returns the ProbeMeter measurement result of single-ended active R&S probes, the voltage measured between the probe tip and the ground.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Result> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/Software: FW 2.25

TRPRobe:PMETer:RESults:COMMon?
PROBe<m>:PMETer:RESults:COMMon?

Returns the ProbeMeter measurement result of differential active R&S probes: the common mode voltage, which is the mean voltage between the signal sockets and the ground socket.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Result> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/Software: FW 2.25

TRPRobe:PMETer:RESults:DIFFerential?
PROBe<m>:PMETer:RESults:DIFFerential?

Returns the ProbeMeter measurement result of differential active R&S probes, the differential voltage - the voltage between the positive and negative signal sockets.

Suffix:
<m> 1..4
Selects the input channel.

Return values:

<DiffMeasRes> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/Software: FW 2.25

TRPRobe:PMETer:RESults:NEGative?
PROBe<m>:PMETer:RESults:NEGative?

Returns the ProbeMeter measurement result of differential active R&S probes, the voltage that is measured between the negative signal socket and the ground.

Suffix:

<m> 1..4
Selects the input channel.

Return values:

<SgEndNegMeasRes> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/Software: FW 2.25

TRPRobe:PMETer:RESults:POSitive?
PROBe<m>:PMETer:RESults:POSitive?

Returns the ProbeMeter measurement result of differential active R&S probes, the voltage that is measured between the negative signal socket and the ground.

Suffix:

<m> 1..4
Selects the input channel.

Return values:

<Result> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/Software: FW 2.25

17.8.7.3 Passive Probes

TRPRobe:SETup:ATTenuation:MODE....1094
PROBe<m>:SETup:ATTenuation:MODE....1094
TRPRobe:SETup:ATTenuation:UNIT....1094
PROBe<m>:SETup:ATTenuation:UNIT....1094
TRPRobe:SETup:ATTenuation:MANual....1094
PROBe<m>:SETup:ATTenuation:MANual....1094
TRPRobe:SETup:GAIN:MANual....1095
PROBe<m>:SETup:GAIN:MANual....1095

TRPRobe:SETup:ATTenuation:MODE <AttenuationMode>
PROBe<m>:SETup:ATTenuation:MODE <PrbAttMd>

Set the mode to MANual if the instrument does not detect the probe.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<PrbAttMd> AUTO | MANual
*RST: AUTO

Usage: Asynchronous command

TRPRobe:SETup:ATTenuation:UNIT <AttenuationUnit>
PROBe<m>:SETup:ATTenuation:UNIT <AttenuationUnit>

Sets the unit for the connected probe type if `PROBe<m>:SETup:ATTenuation:MODE` on page 1094 is set to `MANual`.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<AttenuationUnit> V | A | W
Voltage probe (V), current probe (A), power probe (W)
*RST: V

Usage: Asynchronous command

TRPRobe:SETup:ATTenuation:MANual <PrbAttMdManual>
PROBe<m>:SETup:ATTenuation:MANual <ManualAttenuation>

Sets the attenuation for the connected probe if `PROBe<m>:SETup:ATTenuation:MODE` on page 1094 is set to `MANual`.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<ManualAttenuation>	Range:	100E-6 to 10000
	Increment:	0.1
	*RST:	1
	Default unit:	Depends on the selected unit

Usage: Asynchronous command

TRPRobe:SETup:GAIN:MANual <GainManual>
PROBe<m>:SETup:GAIN:MANual <GainManual>

Sets the gain of a current probe.

Suffix:

<m> 1..4
Selects the input channel. The number of channels depends on the instrument.

Parameters:

<GainManual>	Range:	100E-6 to 10000
	Increment:	100E-6
	*RST:	1
	Default unit:	V/A

Usage: Asynchronous command

17.8.7.4 Active Voltage Probes

TRPRobe:SETup:CMOFfset........1095
PROBe<m>:SETup:CMOFfset........1095
TRPRobe:SETup:ZAXV........1096
PROBe<m>:SETup:ZAXV........1096
TRPRobe:SETup:ACCoupling........1096
PROBe<m>:SETup:ACCoupling........1096
PROBe<m>:SETup:ADVanced:PMToffset........1096
PROBe<m>:SETup:ADVanced:RANGe........1097
TRPRobe:SETup:ADVanced:FILTer........1097
PROBe<m>:SETup:ADVanced:FILTer........1097
TRPRobe:SETup:ADVanced:AUDioverload........1097
PROBe<m>:SETup:ADVanced:AUDioverload........1097

TRPRobe:SETup:CMOFfset <CMOffset>
PROBe<m>:SETup:CMOFfset <CMOffset>

Sets the common-mode offset. The setting is only available for differential probes.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:
<CMOffset> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRPRobe:SETup:ZAXV <ExtAttRTZA15>
PROBe<m>:SETup:ZAXV <ExtAttRTZA15>

If you use the external attenuator R&S RT-ZA15 together with one of the differential active probes R&S RT-ZD10/20/30, enable it to include the external attenuation in the measurements.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ExtAttRTZA15> ON | OFF
*RST: OFF

Usage: Asynchronous command

TRPRobe:SETup:ACCoupling <ProbeCouplingAC>
PROBe<m>:SETup:ACCoupling <ProbeCouplingAC>

Enables AC coupling in the R&S RT-ZPR20 probe, which removes DC and very low-frequency components. The R&S RT-ZPR20 probe requires 50 Ω input termination, for which the channel AC coupling is not available. The probe setting allows AC coupling also at 50 Ω inputs.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ProbeCouplingAC> ON | OFF
*RST: OFF

Usage: Asynchronous command

PROBe<m>:SETup:ADVanced:PMToffset

Sets the measured ProbeMeter value as offset. Make sure that the ProbeMeter is active before you use this command.

Only available for power rail probes R&S RT-ZPR.

Suffix:
<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

PROBe<m>:SETup:ADVanced:RANGe <ProbeRange>

Sets the voltage range of a R&S RT-ZHD probe.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ProbeRange> AUTO | MHIGh | MLOW

AUTO
The voltage range is set with `CHANnel<m>:SCALe`.

MHIGh
Sets the higher voltage range of the connected probe. To query the value, use `PROBe<m>:SETup:ATTenuation[:AUTO]?`.

MLOW
Sets the lower voltage range of the connected probe. To query the value, use `PROBe<m>:SETup:ATTenuation[:AUTO]?`.

*RST: AUTO

Usage: Asynchronous command

TRPRobe:SETup:ADVanced:FILTer <BwFilterSt>
PROBe<m>:SETup:ADVanced:FILTer <BwFilterSt>

Enables the lowpass filter in the probe control box.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<BwFilterSt> ON | OFF

*RST: OFF

Usage: Asynchronous command

TRPRobe:SETup:ADVanced:AUDioverload <AudibOvrrg>
PROBe<m>:SETup:ADVanced:AUDioverload <AudibOvrrg>

Activates the acustic overrange warning in the probe control box.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Sound> ON | OFF
*RST: OFF

Usage: Asynchronous command

17.8.7.5 Modular Probes

The commands of this chapter are relevant for R&S RT-ZM modular probes. The commands are available in firmware version 3.40 and higher.

The suffix <m> selects the input channel to which the probe is connected.

TRPRobe:SETup:PRMode....1098
PROBe<m>:SETup:PRMode....1098
PROBe<m>:SETup:DMOFfset....1099
PROBe<m>:SETup:CMOFfset....1099
PROBe<m>:SETup:NOFFset....1099
PROBe<m>:SETup:POFFset....1100
PROBe<m>:SETup:TERM:STATe....1100
PROBe<m>:SETup:TERM:MODE....1100
PROBe<m>:SETup:TERM:MEASure?....1101
PROBe<m>:SETup:TERM:ADJust....1101

TRPRobe:SETup:PRMode <PrbMeasMd>
PROBe<m>:SETup:PRMode <MeasMode>

Sets the measurement mode of the modular probe.

Suffix:
<m> 1..4

Parameters:
<MeasMode> DMODe | CMODe | PMODe | NMODe

DMODe
Differential mode input voltage (V_{dm}), the voltage between the positive and negative input terminal.

CMODe
Common mode input voltage (V_{cm}) , the mean voltage between the positive and negative input terminal vs. ground.

PMODe
Positive single-ended input voltage (V_p). the voltage between the positive input terminal and ground.

NMODe
Negative single-ended input voltage (V_N). the voltage between the negative input terminal and ground.

*RST: DMODe

Usage: Asynchronous command

PROBe<m>:SETup:DMOFfset <DMOffset>

Sets the differential offset to compensate a DC voltage applied to the positive and the negative input terminal.

Suffix:
<m> 1..4

Parameters:
<DMOffset> In "DM" probe mode (`DMODe`), this offset is used as channel offset and considered automatically for correction.

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:CMOFfset <CMOffset>

Sets the common-mode offset. The setting is only available for differential probes.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<CMOffset> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:NOFFset <NOffset>

Sets the negative offset to compensate a DC voltage applied to the negative input terminal (Vp) referenced to ground.

Suffix:
<m> 1..4

Parameters:
<NOffset> In "N" probe mode (`NMODe`), this offset is used as channel offset and considered automatically for correction.

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:POFFset <POffset>

Sets the positive offset to compensate a DC voltage applied to the positive input terminal (Vp) referenced to ground.

Suffix:
<m> 1..4

Parameters:
<POffset> In "P" probe mode (`PMODe`), this offset is used as channel offset and considered automatically for correction.

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:TERM:STATe <VoltageState>

Activates the instrument control of the termination voltage.

Suffix:
<m> 1..4

Parameters:
<VoltageState> ON | OFF

*RST: OFF

Usage: Asynchronous command

PROBe<m>:SETup:TERM:MODE <Mode>

Setting the termination voltage is relevant if you use the R&S RT-ZMA40 SMA module.

You can set a termination voltage to correct the internal 50 Ω termination of the SMA module by the common mode voltage. To control the termination voltage by the instrument, connect the VT terminal of the R&S RT-ZM probe amplifier to the VT terminal of the SMA module using the red DC lead (see R&S RT-ZM User Manual).

Suffix:
<m> 1..4

Parameters:
<Mode> AUTO | MANual

AUTO
The instrument uses the measured common mode voltage to control the termination.

MANual
Enter the voltage to be used for termination.

*RST: AUTO

Usage: Asynchronous command

PROBe<m>:SETup:TERM:MEASure?

Returns the measured common mode voltage.

Suffix:

<m>	1..4

Return values:

<VoltageMeas> Common mode voltage

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

PROBe<m>:SETup:TERM:ADJust <VoltageAdjust>

Sets the voltage to be used for termination correction.

Suffix:

<m>	1..4

Parameters:

<VoltageAdjust> Correction voltage

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

17.8.7.6 Predefined Probes

TRPRobe:SETup:ATTenuation:DEFProbe....1101
PROBe<m>:SETup:ATTenuation:DEFProbe....1101
PROBe<m>:SETup:OFFSet:TOMean....1102

TRPRobe:SETup:ATTenuation:DEFProbe <SelcPredefPrb>
PROBe<m>:SETup:ATTenuation:DEFProbe <PredefinedProbe>

Selects a predefined probe. These are probes that are not recognized automatically but the parameters of the probe are known to the instrument.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<PredefinedProbe> ZC10 | ZC20 | ZC30 | ZD01A100 | ZD01A1000 | ZZ80 | FREE | ZS10L | ZD02 | ZD08 | ZC02100 | ZC021000 | ZC03 | ZD002A10 | ZD002A100 | ZD003A20 | ZD003A200 | ZC3110 | ZC311 | ZC3101

ZC10 | ZC20 | ZC30 | ZZ80 | ZS10L | ZD02 | ZD08 | ZC03
Type of the probe

ZD01A100 | ZD01A1000 | ZD002A10 | ZD002A100 | ZD003A20 | ZD003A200
High voltage differential probes R&S RT-ZD0xx, attenuation ratio according to the setting on the probe.
A10 = 10:1
A20 = 20:1
A100 = 100:1
A200 = 200:1
A1000 = 1000:1

ZC02100 | ZC021000
Current probes 100 A or 1000 A according to the setting on the probe.

FREE
Any other probe that is not recognized by the instrument.

*RST: FREE

Usage: Asynchronous command

PROBe<m>:SETup:OFFSet:TOMean

Performs an automatic compensation for a DC component of the specified input signal using the result of a background mean measurement.

Suffix:

<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

17.8.7.7 Current Probes

To set up R&S RT-ZC10 and R&S RT-ZC20, use `PROBe<m>:SETup:ATTenuation:DEFProbe`.

TRPRobe:SETup:DEGauss....1103
PROBe<m>:SETup:DEGauss....1103
PROBe<m>:SETup:OFFSet:STPRobe....1103
PROBe<m>:SETup:OFFSet:ZADJust....1103

TRPRobe:SETup:DEGauss
PROBe<m>:SETup:DEGauss

Demagnetizes the core if it has been magnetized by switching the power on and off, or by an excessive input. Always carry out demagnetizing before measurement. The demagnetizing process takes about one second.

Suffix:

<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

Firmware/Software: FW 2.50

PROBe<m>:SETup:OFFSet:STPRobe

Saves the zero adjust value in the probe box. If you connect the probe to another channel or to another R&S RTx oscilloscope, the value is read out again.

Suffix:

<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

PROBe<m>:SETup:OFFSet:ZADJust <ZroAdjVal>

set the waveform to zero position. It corrects the effect of a voltage offset or temperature drift. To set the value by the instrument, use `PROBe<m>:SETup:OFFSet:AZERo`.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<ZroAdjVal> Range: -100 to 100
Increment: 0.1
*RST: 0
Default unit: %

Usage: Asynchronous command

Firmware/Software: FW 2.50

17.8.7.8 Probe Attributes

TRPRobe:ID:SWVersion?..........1104
PROBe<m>:ID:SWVersion?..........1104
TRPRobe:ID:PRDate?..........1104
PROBe<m>:ID:PRDate?..........1104

TRPRobe:ID:PARTnumber?..1104
PROBe<m>:ID:PARTnumber?...1104
TRPRobe:ID:SRNumber?...1105
PROBe<m>:ID:SRNumber?..1105
TRPRobe:SETup:CAPacitance?..1105
PROBe<m>:SETup:CAPacitance?...1105
TRPRobe:SETup:IMPedance?...1105
PROBe<m>:SETup:IMPedance?..1105

TRPRobe:ID:SWVersion?
PROBe<m>:ID:SWVersion?

Queries the version of the probe firmware.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<Softwareversion>	Version number in a string.

Usage: Query only
Asynchronous command

TRPRobe:ID:PRDate?
PROBe<m>:ID:PRDate?

Queries the production date of the probe.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<ProductionDate>	Date in a string.

Usage: Query only
Asynchronous command

TRPRobe:ID:PARTnumber?
PROBe<m>:ID:PARTnumber?

Queries the R&S part number of the probe.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<PartNumber>	Part number in a string.

Usage: Query only
Asynchronous command

TRPRobe:ID:SRNumber?
PROBe<m>:ID:SRNumber?

Queries the serial number of the probe.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<SerialNo>	Serial number in a string.

Usage: Query only
Asynchronous command

TRPRobe:SETup:CAPacitance?
PROBe<m>:SETup:CAPacitance?

Queries the input capacitance of the probe.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<InputCapacity>	Range: 100E-15 to 1E-9 *RST: 10E-12 Default unit: F

Usage: Query only
Asynchronous command

TRPRobe:SETup:IMPedance?
PROBe<m>:SETup:IMPedance?

Queries the termination of the probe.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<InputImpedance>	Range: 0 to 1E+9 *RST: 50 Default unit: Ω

Usage: Query only
Asynchronous command

17.8.7.9 Probe Adapter

PROBe<m>:SETup:ADAPter?....................1106
PROBe<m>:SETup:ATTenuation:TDEFprobe....................1106

PROBe<m>:SETup:ADAPter?

Queries the adapter status, whether the instrument identified the adapter.

Suffix:

<m>	1..4 Selects the input channel.

Return values:

<Type>	NONE | Z2T *RST: NONE

Usage:	Query only Asynchronous command

PROBe<m>:SETup:ATTenuation:TDEFprobe <TekPredefProbe>

Selects the Tektronics probe that is connected to the R&S RT-Z2T adapter.

Suffix:

<m>	1..4

Parameters:

<TekPredefProbe>	NONE | P5205A50 | P5205A500 | P5210A100 | P5210A1000 | P6205 | P6241 | P6243 | P6245 | P6246A1 | P6246A10 | P6247A1 | P6247A10 | P6248A1 | P6248A10 | P6249 | P6250A5 | P6250A50 | P6251A5 | P6251A50 | P6701B | P6703B | P6711 | P6713 | TCP202 *RST: NONE

Example:

```
PROBe2:SETup:ADAPter
<-- Z2T
PROBe2:SETup:ATTenuation:TDEFprobe P5205A50
```

Checks the adapter state on channel 2, and selects the probe P5205A50.

Usage:	Asynchronous command

17.8.8 R&S RT-ZVC Probe

- Probe Setup 1106
- Resolution and Data 1118

17.8.8.1 Probe Setup

In the `ZVC:Z<m>:` commands, the following suffixes are used:

- `<m>` selects the multi-channel probe. The available range is `1..2`
- `<n>` selects the voltage or current channel. The available range depends on the probe characteristics and is `1..2` or `1..4`

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

ZVC:BANDwidth....1107
ZVC:TYPE....1108
ZVC:Z<m>:I<n>:BANDwidth?....1108
ZVC:Z<m>:I<n>:IMPedance....1108
ZVC:Z<m>:I<n>:OFFSet....1109
ZVC:RESCoupled....1109
ZVC:Z<m>:I<n>:OVERload:RSTO....1109
ZVC:Z<m>:I<n>:OVERload:VALue?....1110
ZVC:Z<m>:I<n>:POSition....1110
ZVC:Z<m>:I<n>:SCALe....1111
ZVC:Z<m>:I<n>:SHUNt:EVALue....1111
ZVC:Z<m>:I<n>:SHUNt:MODE....1111
ZVC:Z<m>:I<n>:SHUNt:MAXCurrent....1111
ZVC:Z<m>:I<n>:SHUNt:MXCValue?....1112
ZVC:Z<m>:I<n>:SHUNt:MAXVoltage....1112
ZVC:Z<m>:I<n>:SKEW....1113
ZVC:Z<m>:I<n>[:STATe]....1113
ZVC:Z<m>:V<n>:BANDwidth?....1113
ZVC:Z<m>:V<n>:IMPedance....1114
ZVC:Z<m>:V<n>:OFFSet....1114
ZVC:Z<m>:V<n>:OVERload:RSTO....1114
ZVC:Z<m>:V<n>:OVERload:VALue?....1115
ZVC:Z<m>:V<n>:POSition....1115
ZVC:Z<m>:V<n>:SCALe....1115
ZVC:Z<m>:V<n>:SKEW....1116
ZVC:Z<m>:V<n>[:STATe]....1116
ZVC:Z<m>:ID:NAME?....1116
ZVC:Z<m>:ID:PARTnumber?....1116
ZVC:Z<m>:ID:SRNumber?....1117
ZVC:Z<m>:ID:SWVersion?....1117
ZVC:Z<m>:I<n>:ZERComp:DETect....1117
ZVC:Z<m>:I<n>:ZERComp:USE....1117
ZVC:Z<m>:I<n>:ZERComp:STATe?....1118

ZVC:BANDwidth <Bandwidth>

Sets the bandwidth limit of the probe. The bandwidth specifies the maximum frequency at which a purely sinusoidal signal is still transferred at 89 % (0.1 dB) of its amplitude.

Parameters:

<Bandwidth>	Range:	5000 to 1E+6
	Increment:	5000
	*RST:	1E+6
	Default unit:	Hz

Usage: Asynchronous command

ZVC:TYPE <DecimationMode>

Sets the decimation mode for the R&S RT-ZVC probe. Decimation reduces the data stream of the ADC to a stream of waveform points with lower sample rate and a less precise time resolution.

Parameters:

<DecimationMode> SAMPle | PDETect | HRESolution

*RST: SAMPle

Usage: Asynchronous command

ZVC:Z<m>:I<n>:BANDwidth?

Queries the bandwidth of the current channel. You can set the probe bandwidth with `ZVC:BANDwidth`.

The bandwidth of some current channels is restricted to 300KHz due to their vertical settings.

Suffix:

<m> 1..2

<n> 1..4

Return values:

<Bandwidth>	Range:	5000 to 1E+6
	Increment:	5000
	*RST:	1E+6
	Default unit:	Hz

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:IMPedance <MeasImp>

Sets the current impedance of the probe channel for power calculations and measurements.

Suffix:

<m> 1..2

<n> 1..4

Parameters:

<MeasImp>
Range: 1 to 100E+3
Increment: 1
*RST: 50
Default unit: Ω

Usage: Asynchronous command

ZVC:Z<m>:I<n>:OFFSet <VerticalOffset>

Sets the offset current for the current channel.

Suffix:

<m> 1..2

<n> 1..4

Parameters:

<VerticalOffset>
Range: -1 to 1
Increment: 0.01
*RST: 0
Default unit: V

Usage: Asynchronous command

ZVC:RESCoupled <CoupleToAnalog>

Sets the resolution of all R&S RT-ZVC channels.

Parameters:

<CoupleToAnalog> ON | OFF

ON
The resolution of the analog channels is applied to R&S RT-ZVC channels. The signal is automatically interpolated or decimated to get the analog resolution.

OFF
The resolution of R&S RT-ZVC channels is set in a way so that the record length of the waveforms is minimum 1000 samples.

*RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:I<n>:OVERload:RSTO <Settings>

Resets the overload indication at the probe.

In internal shunt mode (`ZVC:Z<m>:I<n>:SHUNt:MODE` is set to `INTShunt`), chooses whether to keep the operation range adjusted by the R&S RT-ZVC during overload or to restore the original settings for the given current channel.

In external shunt mode (`ZVC:Z<m>:I<n>:SHUNt:MODE` is set to `EXTShunt`), there is no adjustment of the operation range during overload.

Suffix:

<m>	1..2
<n>	1..4

Setting parameters:

<Settings> ADJusted | ORIGinal

ADJusted
The operation range that is automatically adjusted by the probe during overload is kept.
In internal shunt mode, during an overload at a current channel, the probe switches automatically to the next higher range. If the 10A range is also exceeded, the amperemeter switches to external shunt mode to protect the probe against permanent damage.

ORIGinal
The original operation range is restored as before the overload and the adjustment of the operation range.

Usage: Setting only
Asynchronous command

ZVC:Z<m>:I<n>:OVERload:VALue?

Queries if an overload of the current channel was detected.

Suffix:

<m>	1..2
<n>	1..4

Return values:

<Overload> ON | OFF
*RST: OFF

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:POSition <VertPosi>

Sets the vertical position of the indicated current channel as a graphical value.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VertPosi>
Range: -5 to 5
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SCALe <VerticalScale>

Sets the vertical scale for the current channel in Volts per division. Increasing the scale compresses the display of the signal.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VerticalScale>
Range: 1E-15 to 10E+24
Increment: 10E-6
*RST: 3
Default unit: V/div

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:EVALue <ExtShuntVal>

Defines the value of the external shunt resistor to calculate the correct current values.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<ExtShuntVal>
Range: 10E-6 to 10000
Increment: 0.02
*RST: 1
Default unit: Ω

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MODE <ShuntMode>

Sets the internal or external shunt mode.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<ShuntMode> INTShunt | EXTShunt
*RST: INTShunt

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MAXCurrent <IntShuntMeasRg>

Sets the maximum current and the internal shunt value.

The internal shunt and maximum current parameter pairs are defined as described in the table below.

Parameter	Internal shunt	Maximum current
A100r01	10 A	10 mΩ
A40R01	4.5 A	10 mΩ
A45M10r	45 mA	10 Ω
A4M510r	4.5 mA	10 Ω
A45u10k	45 µA	10 KΩ
A4U510k	4.5 µA	10 KΩ

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<IntShuntMeasRg> A100r01 | A40R01 | A45M10r | A4M510r | A45u10k | A4U510k
*RST: A100r01

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MXCValue?

Queries the maximum current.

Suffix:
<m> 1..2

<n> 1..4

Return values:
<MaxCurrentValue> Default unit: A

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MAXVoltage <ExtShuntMeasRg>

Sets the maximum voltage for the external shunt.

Suffix:
<m> 1..2

<n> 1..4

Parameters:

<ExtShuntMeasRg> V450m | V045m

`V450m`: 450 mV
`V045m`: 45 mV

*RST: V450m

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SKEW <DeskewOffset>

Sets the skew offset value for the current probe channel. This is a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Suffix:

<m> 1..2

<n> 1..4

Parameters:

<DeskewOffset> Range: -6E-6 to 6E-6
Increment: 200E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

ZVC:Z<m>:I<n>[:STATe] <State>

Enables the corresponding current channel of the probe.

Suffix:

<m> 1..2

<n> 1..4

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:V<n>:BANDwidth?

Queries the bandwidth of the voltage channel. You can set the probe bandwidth with `ZVC:BANDwidth`.

Suffix:

<m> 1..2

<n> 1..4

Return values:

<Bandwidth>	Range: 5000 to 1E+6 Increment: 5000 *RST: 1E+6 Default unit: Hz
Usage:	Query only Asynchronous command

ZVC:Z<m>:V<n>:IMPedance <MeasImp>

Sets the voltage impedance of the probe channel for power calculations and measurements.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<MeasImp>	Range: 1 to 100E+3 Increment: 1 *RST: 50 Default unit: Ω
Usage:	Asynchronous command

ZVC:Z<m>:V<n>:OFFSet <VerticalOffset>

Sets the vertical offset for the voltage channel.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VerticalOffset>	Range: -1 to 1 Increment: 0.01 *RST: 0 Default unit: V
Usage:	Asynchronous command

ZVC:Z<m>:V<n>:OVERload:RSTO

Resets the overload indication at the probe.

Suffix:

<m>	1..2
<n>	1..4

Usage: Event
Asynchronous command

ZVC:Z<m>:V<n>:OVERload:VALue?

Queries if an overload of the voltage channel was detected.

Suffix:

<m>	1..2
<n>	1..4

Return values:

<Overload> ON | OFF
*RST: OFF

Usage: Query only
Asynchronous command

ZVC:Z<m>:V<n>:POSition <VertPosi>

Sets the vertical position of the indicated voltage channel as a graphical value.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VertPosi> Range: -5 to 5
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

ZVC:Z<m>:V<n>:SCALe <VerticalScale>

Sets the vertical scale for the voltage channel in Volts per division. Increasing the scale compresses the display of the signal.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VerticalScale> Range: 1E-15 to 10E+24
Increment: 10E-6
*RST: 3
Default unit: V/div

Usage: Asynchronous command

ZVC:Z<m>:V<n>:SKEW <DeskewOffset>

Sets the skew offset value for the voltage probe channel. This is a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Suffix:

<m> 1..2

<n> 1..4

Parameters:

<DeskewOffset> Range: -6E-6 to 6E-6
Increment: 200E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

ZVC:Z<m>:V<n>[:STATe] <State>

Enables the corresponding voltage channel of the probe.

Suffix:

<m> 1..2

<n> 1..4

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:ID:NAME?

Queries the name of the probe.

Suffix:

<m> 1..2

Return values:

<Name>

Usage: Query only
Asynchronous command

ZVC:Z<m>:ID:PARTnumber?

Queries the R&S part number of the probe.

Suffix:

<m> 1..2

Return values:
<PartNumber>

Usage: Query only
Asynchronous command

ZVC:Z<m>:ID:SRNumber?

Queries the serial number of the probe.

Suffix:
<m> 1..2

Return values:
<SerialNo>

Usage: Query only
Asynchronous command

ZVC:Z<m>:ID:SWVersion?

Queries the version of the probe firmware.

Suffix:
<m> 1..2

Return values:
<Softwareversion>

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:ZERComp:DETect

Measures the zero offset, the mean value on a currentless DUT. If temperature changes, or if you change the shunt or other probe settings, repeat the measurement.

Suffix:
<m> 1..2

<n> 1..4

Usage: Event
Asynchronous command

ZVC:Z<m>:I<n>:ZERComp:USE <UseZeroComp>

Enables the automatic compensation of the measured zero offset.

Suffix:
<m> 1..2

<n> 1..4

Parameters:

<UseZeroComp>	ON | OFF *RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:I<n>:ZERComp:STATe?

Returns the status of the zero compensation of the indicated current channel.

Suffix:

<m>	1..2
<n>	1..4

Return values:

<ZeroCompState>	ON | OFF ON: Zero compensation is used Off: Zero Compensation is not used *RST: OFF

Usage: Query only
Asynchronous command

17.8.8.2 Resolution and Data

ACQuire:POINts:ZVALue?....1118
ACQuire:ZRESolution?....1118
ZVC:Z<m>:I<n>:DATA:HEADer?....1119
ZVC:Z<m>:V<n>:DATA:HEADer?....1119
ZVC:Z<m>:I<n>:DATA[:VALues]?....1119
ZVC:Z<m>:V<n>:DATA[:VALues]?....1119

ACQuire:POINts:ZVALue?

Returns the current record length used by the R&S RT-ZVCmulti-channel power probe channel.

Return values:

<ZUIRecordLength>	Range: 1000 to 400E6 Increment: 2 *RST: 1000 Default unit: Sa

Usage: Query only
Asynchronous command

ACQuire:ZRESolution?

Returns the current resolution of the R&S RT-ZVCmulti-channel power probe channel.

Return values:

<ZUIResolution> Range: 1E-15 to 0.5
Increment: 10E-12
*RST: 500E-6
Default unit: s

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:DATA:HEADer?
ZVC:Z<m>:V<n>:DATA:HEADer?

Return the header of amperemeter and voltmeter waveform data.

Table 17-2: Header data

Position	Meaning	Example
1	Start time in s	-0.0001
2	Stop time in s	0.0001
3	Number of values in one waveform (record length)	1000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of values (record length)..	1

Suffix:

<m> 1..2

<n> 1..4

Example:

```
ZVC:Z2:V2:DATA:HEADer?
<-- -0.0001,0.0001,1000,1
```

Usage: Query only

ZVC:Z<m>:I<n>:DATA[:VALues]?
ZVC:Z<m>:V<n>:DATA[:VALues]?

Return the waveform data of the amperemeter and voltmeter for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Set the export format to ASCii or REAL,32 using `FORMat[:DATA]`. Export of raw data is not supported.

Suffix:

<m> 1..2

<n> 1..4

Return values:

<Data> Liste of values according to the format setting.

Example:

```
FORMat:DATA ASCii
ZVC:Z2:V2:DATA:HEADer?
<-- -0.0001,0.0001,1000,1
ZVC:Z2:V2:DATA:VaLues?
<-- 0.00083049136446789,-0.0042279558256269,-0.0076707201078534,
-0.011294682510197,-0.015915233641863,-0.018905002623796,
-0.022423267364502, ….
```

1000 values are returned.

Usage: Query only

17.8.9 Differential Signals

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

DIFFerential<m>:STATe....1120
DIFFerential<m>:PSIGnal[:SELect]....1121
DIFFerential<m>:NSIGnal[:SELect]....1121
DIFFerential<m>:AOUTput....1121
DIFFerential<m>:BOUTput....1121
DIFFerential<m>:COMMon:SCALe....1121
DIFFerential<m>:DIFFerential:SCALe....1121
DIFFerential<m>:COUPling....1122
DIFFerential<m>:COMMon:OFFSet....1122
DIFFerential<m>:DIFFerential:OFFSet....1122
DIFFerential<m>:COMMon:POSition....1122
DIFFerential<m>:DIFFerential:POSition....1122

DIFFerential<m>:STATe <State>

Activates the differential signal. Two differential signals are available.

Suffix:

<m> 1..2
Differential signal

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

DIFFerential<m>:PSIGnal[:SELect] <Signal>
DIFFerential<m>:NSIGnal[:SELect] <Signal>

Select the analog input channel of the positive and negative signal.

For differential signal 1, analog Ch1 and Ch3 are used. Differential signal 2 uses Ch2 and Ch4.

Suffix:

<m>	1..2 Differential signal

Parameters:

<Signal>	C1W1 | C2W1 | C3W1 | C4W1 *RST: C1W1

Usage: Asynchronous command

DIFFerential<m>:AOUTput <Output>
DIFFerential<m>:BOUTput <Output>

Select the waveform that is the result of differential processing. 2 output waveforms per digital signal can be defined.

Suffix:

<m>	1..2 Differential signal

Parameters:

<Output>	P | N | DIFF | COM **P | N** Positive or negative waveform. **DIFF** Differential waveform. **COM** Common mode waveform. *RST: COM

Usage: Asynchronous command

DIFFerential<m>:COMMon:SCALe <Scale>
DIFFerential<m>:DIFFerential:SCALe <Scale>

Set the vertical scale of differential and common mode waveforms, respectively.

Vertical settings of P and N output are the vertical settings of the input channels.

Suffix:

<m>	1..2 Differential signal

Parameters:
<Scale> Scale value, given in Volts per division.

Range: 1E-3 to 100
Increment: 1E-3
*RST: 0.05
Default unit: V/div

Usage: Asynchronous command

DIFFerential<m>:COUPling <ScaleCoupling>

If enabled for a differential signal, the vertical scales of P, N, differential and common mode waveforms are coupled. The scales are related as follows:

$VertScale_P = VertScale_N = VertScale_{CM} = VertScale_{Diff}/2$

You can disable the scale coupling and set an individual scale for each waveform.

Suffix:
<m> 1..2
Differential signal

Parameters:
<ScaleCoupling> ON | OFF

*RST: ON

Usage: Asynchronous command

DIFFerential<m>:COMMon:OFFSet <Offset>
DIFFerential<m>:DIFFerential:OFFSet <Offset>

Set the offset of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Suffix:
<m> 1..2
Differential signal

Parameters:
<Offset> Offset value

Range: -1 to 1
Increment: 0.01
*RST: 0
Default unit: V

Usage: Asynchronous command

DIFFerential<m>:COMMon:POSition <Position>
DIFFerential<m>:DIFFerential:POSition <Position>

Set the vertical position of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Suffix:

<m>	1..2 Differential signal

Parameters:

<Position> Position value

Range: -5 to 5
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

17.8.10 Digital Filter

CHANNel<m>:DIGFilter:STATe....1123
TRIGger<m>:RFReject....1123
TRIGger<m>:RFSReject....1124

CHANNel<m>:DIGFilter:STATe <State>

Enables the DSP filter for input channels.

Suffix:

<m>	1..4 Irrelevant, omit the suffix. The filter is enabled to all channels.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:RFReject <Bandwidth>

Sets the limit frequency for input channels and the trigger source.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Parameters:

<Bandwidth>

Range: 100E+3 to 2E+9
Increment: 1000
*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

TRIGger<m>:RFSReject <HFRejectTrigger>

Enables the DSP filter for the trigger channel. Frequencies higher than the bandwidth (`TRIGger<m>:RFReject`) are rejected, lower frequencies pass the filter.

Suffix:

<m>	1..3 The suffix is irrelevant.

Parameters:

<HFRejectTrigger>	ON | OFF *RST: OFF

Usage: Asynchronous command

17.8.11 Skew

CHANnel<m>:SKEW:MANual....1124
CHANnel<m>:SKEW:TIME....1124
PROBe<m>:SKEState....1125

CHANnel<m>:SKEW:MANual <ManualCompens>

If enabled, the skew offset value (`CHANnel<m>:SKEW:TIME`) is used for compensation. This improves horizontal and trigger accuracy.

Suffix:

<m>	1..4 Selects the input channel.

Parameters:

<ManualCompens>	ON | OFF *RST: ON

Usage: Asynchronous command

CHANnel<m>:SKEW:TIME <Offset>

Sets an delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically. It affects only the selected input channel.

Suffix:

<m>	1..4 Selects the input channel.

Parameters:

<Offset>	Range: -100E-9 to 100E-9 Increment: 10E-15 *RST: 0 Default unit: s

Usage: Asynchronous command

PROBe<m>:SKEState <ProbeSkew>

If enabled, the skew of all connected active probes is measured, displayed, and used for deskewing.

Suffix:

<m> 1..4
The suffix is irrelevant. The setting affects all active channels.

Parameters:

<ProbeSkew> ON | OFF
*RST: OFF

Usage: Asynchronous command

17.8.12 High Definition Mode

HDEFinition:STATe ... 1125
HDEFinition:BWIDth ... 1125
HDEFinition:RESolution? ... 1126
FORMat:BORDer ... 1126

HDEFinition:STATe <State>

Activates the high definition mode of the instrument.

Parameters:

<State> ON | OFF
ON: high definition mode, up to 16 bit digital resolution
OFF: normal oscilloscope mode
*RST: OFF

Usage: Asynchronous command

HDEFinition:BWIDth <Bandwidth>

Sets the filter bandwidth for the high definition mode.

Parameters:

<Bandwidth> Range: 10000 to max. 2 GHz, depending on the instrument bandwidth.
Increment: 1000
*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

See "Bandwidth" on page 152 for bandwidth limits.

HDEFinition:RESolution?

Returns the resulting digital resolution in high definition mode.

Return values:

<Resolution>	Range:	0 to 16
	Increment:	0.1
	*RST:	0

Usage: Query only
Asynchronous command

FORMat:BORDer <ByteOrder>

Sets the endianess.

The command is only relevant for raw data export in high definition mode (16 bit word length).

Parameters:

<ByteOrder> LSBFirst | MSBFirst

LSB first: little endian, least significant byte first
MSB first: big endian, most significant byte first

*RST: LSBFirst

Usage: Asynchronous command

17.8.13 Reference Clock

You can select an internal or external reference clock. The external reference is connected to the external reference input on the rear panel.

SENSe[:ROSCillator]:SOURce... 1126
SENSe[:ROSCillator]:EXTernal:FREQuency...1126

SENSe[:ROSCillator]:SOURce <RefSource>

Enables the use of the external reference signal instead of the internal OCXO reference.

Parameters:

<RefSource> INTernal | EXTernal

*RST: INTernal

Usage: Asynchronous command

SENSe[:ROSCillator]:EXTernal:FREQuency <ExternalRef>

Sets the frequency of an external reference input signal that is connected to the external reference input on the rear panel.

Parameters:

<ExternalRef> *RST: 10E+6
Default unit: Hz
Constant frequency of 10E+6 Hz.

Usage: Asynchronous command

17.9 Trigger

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- Basic Trigger Settings........ 1128
- Edge Trigger........ 1131
- Glitch Trigger........ 1134
- Width Trigger........ 1135
- Runt Trigger........ 1136
- Window Trigger........ 1139
- Timeout Trigger........ 1141
- Interval Trigger........ 1142
- Slew Rate Trigger........ 1144
- Data2Clock Trigger........ 1146
- State Trigger........ 1148
- Pattern Trigger........ 1150
- Serial Pattern Trigger........ 1153
- TV/Video Trigger........ 1155
- Line Trigger........ 1159
- Holdoff........ 1160
- Noise Reject........ 1163
- Trigger Sequence........ 1166
- Trigger Control........ 1169
- Actions on Trigger........ 1171
- External Trigger Input........ 1172
- Acquisition Info........ 1173

17.9.1 Basic Trigger Settings

DISPlay:TRIGger:LINes ... 1128
TRIGger<m>:SOURce[:SELect] ... 1128
TRIGger<m>:TYPE ... 1129
TRIGger<m>:LEVel<n>[:VALue] ... 1130
TRIGger<m>:FINDlevel ... 1130

DISPlay:TRIGger:LINes <State>

Hides or shows the trigger levels in the diagrams.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:SOURce[:SELect] <SourceDetailed>

Selects the source of the trigger signal.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
Available values depend on the selected trigger source. For input channels CHAN1...4, a trigger sequence can be configured.
For all other trigger sources, only suffix 1 is allowed.
See also: `TRIGger<m>:SEQuence:MODE`

Parameters:

<SourceDetailed> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4 | EXTernanalog | SBUS | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | LOGIC | MSOB1 | MSOB2 | MSOB3 | MSOB4 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | LINE

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channels

EXTernanalog
External analog signal connected to the External Trigger Input.
For this source, only the analog edge trigger is available.

LINE
The instrument generates the trigger from the AC power input and synchronizes the signal to the AC power frequency.

SBUS
Serial bus

D0...D15
Digital channels (option R&S RTE-B1)
See also: Chapter 17.18.4, "Trigger Settings for Digital Signals and Parallel Buses", on page 1874

LOGIc
Logic combination of digital channels, used as trigger source (option R&S RTE-B1)

MSOB1 | MSOB2 | MSOB3 | MSOB4
Parallel bus (option R&S RTE-B1)

Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4
Input channels of R&S RT-ZVCmulti-channel power probe. Only available in the A-trigger with trigger type EDGE.

DIFF1 | DIFF2 | COMMON1 | COMMON2
Differential signals

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:TYPE <Type>

Selects the trigger type to trigger on analog channels or the external trigger input.

See also: Chapter 5.3, "Trigger Types", on page 203.

To trigger on digital channels and parallel buses, use `TRIGger<m>:PARallel:TYPE`.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
For suffixes 2 and 3, the following trigger types are availabe: EDGE, GLITch, WIDTh, RUNT, WINDow, TIMeout, INTerval, SLEWrate.

Parameters:

<Type> EDGE | GLITch | WIDTh | RUNT | WINDow | TIMeout | INTerval | SLEWrate | DATatoclock | STATe | PATTern | ANEDge | SERPattern | TV

Most of the type values are self-explanatory.

DATatoclock
Data2Clock: analyzes the relative timimg between a data signal and the synchronous clock signal. For trigger settings, see Chapter 17.9.10, "Data2Clock Trigger", on page 1146.

ANEDge
Edge trigger for external trigger input. Only available if the trigger source is the external trigger input. This trigger type uses the analog input signal. For trigger settings, see Chapter 5.8.2, "External Trigger Setup", on page 237.

SERPattern
Serial pattern for signals with serial data patterns in relation to a clock signal. For trigger settings, see Chapter 17.9.13, "Serial Pattern Trigger", on page 1153

*RST: EDGE

Usage: Asynchronous command

TRIGger<m>:LEVel<n>[:VALue] <Level>

Sets the trigger level for the specified event and source.

If the trigger source is serial bus, the trigger level is set by the thresholds in the protocol configuration.

If the trigger source is LINE (AC power input), no trigger level can be set.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1 to 4
5 = external trigger input
6...11 = not available
12...27 = R&S RT-ZVC input channels
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:

<Level> Voltage for the trigger level.

Range: Depends on vertical scale, channel offset and other settings. The trigger level must be within the current display range.
Increment: 1E-3
*RST: 0
Default unit: V

Example:

```
TRIG:LEV5 0.01
```
Sets the trigger level for the external trigger signal to 10 mV.
```
TRIG2:LEV3 0.2
```
Sets the trigger level for the B-event and B-trigger source channel 3 to 200 mV.

Usage: Asynchronous command

TRIGger<m>:FINDlevel

Sets the trigger level automatically. The command is only relevant if the trigger source is an analog channel CHAN1...4.

Suffix:

<m> 1..3

1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Usage: Event

Asynchronous command

17.9.2 Edge Trigger

TRIGger<m>:EDGE:SLOPe....1131
TRIGger<m>:ANEDge:COUPling....1131
TRIGger<m>:ANEDge:FILTer....1132
TRIGger<m>:ANEDge:CUToff:HIGHpass....1132
TRIGger<m>:ANEDge:CUToff:LOWPass....1133
TRIGger<m>:ANEDge:GND....1133
TRIGger<m>:ANEDge:SLOPe....1133

TRIGger<m>:EDGE:SLOPe <Slope>

Defines the edge for the edge trigger event.

Suffix:

<m> 1..3

1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:ANEDge:COUPling <Coupling>

Sets the coupling for the external trigger signal.

Suffix:

<m> 1..3

Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Coupling> DC | DCLimit | AC

DC

Direct connection with 50 Ω termination, passes both DC and AC components of the trigger signal.

DCLimit

Direct connection with 1 MΩ termination, passes both DC and AC components of the trigger signal.

AC
Connection through DC capacitor, removes DC and very low-frequency components.

Usage: Asynchronous command

TRIGger<m>:ANEDge:FILTer <Filter>

Sets a filter for the external trigger signal to reject high or low frequencies.

Suffix:
<m>
1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Filter>
OFF | LFReject | RFReject

OFF
The trigger signal is not filtered.

LFReject
Frequencies higher than the "Cut-off" frequency are rejected, lower frequencies pass the filter.
You can adjust the "Cut-off" frequency using the `TRIGger<m>:ANEDge:CUToff:LOWPass` command, the default is 50 kHz.

RFReject
Frequencies below the "Cut-off" frequency are rejected, higher frequencies pass the filter.
You can adjust the "Cut-off" frequency using the `TRIGger<m>:ANEDge:CUToff:HIGHpass` command, the default is 50 kHz.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:ANEDge:CUToff:HIGHpass <AnalogCutOffHP>

Frequencies below the "Cut-off" frequency are rejected, higher frequencies pass the filter.

Suffix:
<m>
1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<AnalogCutOffHP>
KHZ5 | KHZ50 | MHZ50

Cut-off frequency

KHZ5
5 kHz

KHZ50
50 kHz

MHZ50
50 MHz

*RST: KHZ50

Usage: Asynchronous command

TRIGger<m>:ANEDge:CUToff:LOWPass <AnalogCutOffLP>

Frequencies higher than the "Cut-off" frequency are rejected, lower frequencies pass the filter.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<AnalogCutOffLP> KHZ5 | KHZ50 | MHZ50

KHZ5
5 kHz

KHZ50
50 kHz

MHZ50
50 MHz

*RST: KHZ50

Usage: Asynchronous command

TRIGger<m>:ANEDge:GND <Ground>

Connects the analog signal to the ground.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Ground> ON | OFF

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:ANEDge:SLOPe <Slope>

Sets the edge for the trigger event.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Slope> POSitive | NEGative

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

17.9.3 Glitch Trigger

TRIGger<m>:GLITch:POLarity....1134
TRIGger<m>:GLITch:RANGe....1134
TRIGger<m>:GLITch:WIDTh....1134

TRIGger<m>:GLITch:POLarity <Polarity>

Defines the polarity of a pulse, that is the direction of the first pulse slope.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 17.4.4, "Polarity Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:GLITch:RANGe <RangeMode>

Selects which glitches are identified: shorter or longer than the width specified using `TRIGger<m>:GLITch:WIDTh`.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode> SHORter | LONGer

SHORter
Glitches shorter than the specified width are identified.

LONGer
Glitches longer than the specified width are identified.

*RST: SHORter

Usage: Asynchronous command

TRIGger<m>:GLITch:WIDTh <Width>

Sets the length of a glitch. The instrument triggers on pulses shorter or longer than this value, depnding on the `TRIGger<m>:GLITch:RANGe` command.

You need to know the expected pulse widths of the circuit to set the glitch width correctly.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-6
*RST: 1E-9
Default unit: s

Usage: Asynchronous command

17.9.4 Width Trigger

TRIGger<m>:WIDTh:POLarity 1135
TRIGger<m>:WIDTh:RANGe 1135
TRIGger<m>:WIDTh:WIDTh 1136
TRIGger<m>:WIDTh:DELTa 1136

TRIGger<m>:WIDTh:POLarity <Polarity>

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Polarity> POSitive | NEGative
See Chapter 17.4.4, "Polarity Parameter", on page 1038.
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:WIDTh:RANGe <RangeMode>

Defines how the range of a pulse width is defined in relation to the width and delta specified using `TRIGger<m>:WIDTh:WIDTh` and `TRIGger<m>:WIDTh:DELTa`, respectively.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range is defined by the width ±delta.

OUTSide
Triggers on pulses outside a given range. The range is defined by the width ±delta.

SHORter
Triggers on pulses shorter than the given width.

LONGer
Triggers on pulses longer than the given width.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:WIDTh:WIDTh <Width>

For the ranges "Within" and "Outside" (defined using `TRIGger<m>:WIDTh:RANGe`), the width defines the center of a range which is defined by the limits "±Delta" (see `TRIGger<m>:WIDTh:DELTa` on page 1136).

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:WIDTh:DELTa <WidthDelta>

Defines a range around the width value specified using `TRIGger<m>:WIDTh:WIDTh`.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

17.9.5 Runt Trigger

TRIGger<m>:RUNT:POLarity....1137
TRIGger<m>:LEVel<n>:RUNT:LOWer....1137
TRIGger<m>:LEVel<n>:RUNT:UPPer....1137

TRIGger<m>:RUNT:RANGe....1137
TRIGger<m>:RUNT:WIDTh....1138
TRIGger<m>:RUNT:DELTa....1138

TRIGger<m>:RUNT:POLarity <Polarity>

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 17.4.4, "Polarity Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:RUNT:LOWer <Level>
TRIGger<m>:LEVel<n>:RUNT:UPPer <Level>

Set the lower and upper voltage thresholds.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1...4
5...31 = not available

Parameters:

<Level> Range: -10 to 10
Increment: 1E-3
*RST: Lower = -0.1, upper =0.1
Default unit: V

Usage: Asynchronous command

TRIGger<m>:RUNT:RANGe <Mode>

Defines the time limit of the runt pulse in relation to the `TRIGger<m>:RUNT:WIDTh` and `TRIGger<m>:RUNT:DELTa` settings.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Mode> ANY | LONGer | SHORter | WITHin | OUTSide

ANY
Triggers on all runts fulfilling the level condition, without time limitation.

LONGer
Triggers on runts longer than the given "Runt width".

SHORter
Triggers on runts shorter than the given "Runt width".

WITHin
Triggers if the runt length is inside a given time range. The range is defined by "Runt width" and "±Delta".

OUTSide
Triggers if the runt length is outside a given time range. The range is defined by "Runt width" and "±Delta".

*RST: ANY

Usage: Asynchronous command

TRIGger<m>:RUNT:WIDTh <Width>

Defines the upper or lower voltage threshold. This command is not available if `TRIGger<m>:RUNT:RANGe` is set to "ANY".

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:RUNT:DELTa <WidthDelta>

Defines a range around the runt width specified using `TRIGger<m>:RUNT:WIDTh`. This command is only available if `TRIGger<m>:RUNT:RANGe` is set to "WITHin" or "OUTSide".

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<WidthDelta> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-12
Default unit: s

Usage: Asynchronous command

17.9.6 Window Trigger

TRIGger<m>:LEVel<n>:WINDow:LOWer....1139
TRIGger<m>:LEVel<n>:WINDow:UPPer....1139
TRIGger<m>:WINDow:RANGe....1139
TRIGger<m>:WINDow:TIME....1140
TRIGger<m>:WINDow:WIDTh....1140
TRIGger<m>:WINDow:DELTa....1141

TRIGger<m>:LEVel<n>:WINDow:LOWer <Level>
TRIGger<m>:LEVel<n>:WINDow:UPPer <Level>

Set the lower and upper voltage limits for the window.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1...4
5...31 = not available

Parameters:

<Level> Range: -10 to 10
Increment: 1E-3
*RST: Lower = -0.1, upper =0.1
Default unit: V

Usage: Asynchronous command

TRIGger<m>:WINDow:RANGe <RangeMode>

Defines the signal run in relation to the window:

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode> ENTer | EXIT | WITHin | OUTSide

ENTer
Triggers when the signal crosses the upper or lower level and thus enters the window made up of these two levels.

EXIT
Triggers when the signal leaves the window.

WITHin
Triggers if the signal stays between the upper and lower level for a specified time. The time is defined using the `TRIGger<m>:WINDow:TIME` command.

OUTSide
Triggers if the signal stays above the upper level or below the lower level for a specified time. The time is defined using the `TRIGger<m>:WINDow:TIME` command.

*RST: ENTer

Usage: Asynchronous command

TRIGger<m>:WINDow:TIME <TimeRangeMode>

Defines the limit of the window in relation to the time specified using `TRIGger<m>:WINDow:WIDTh` and `TRIGger<m>:WINDow:DELTa`. Time conditioning is available for `TRIGger<m>:WINDow:RANGe`= "WITHin" and "OUTSide".

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<TimeRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers if the signal stays inside or outside the vertical window limits at least for the time *Width* - *Delta* and for *Width* + *Delta* at the most.

OUTSide
"Outside" is the opposite definition of "Within". The instrument triggers if the signal stays inside or outside the vertical window limits for a time shorter than *Width* - *Delta* or longer than *Width* + *Delta*.

SHORter
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

LONGer
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:WINDow:WIDTh <Width>

For the ranges "Within" and "Outside" (defined using `TRIGger<m>:WINDow:RANGe`), the width defines the center of a time range which is defined by the limits "±Delta" (see `TRIGger<m>:WINDow:DELTa` on page 1141).

For the ranges "Shorter" and "Longer", it defines the maximum and minimum time lapse, respectively.

Suffix:

<m>	1..3 1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Width>	Range: 100E-12 to 10000 Increment: 100E-9 *RST: 5E-9 Default unit: s

Usage: Asynchronous command

TRIGger<m>:WINDow:DELTa <WidthDelta>

Defines a range around the "Width" value specified using `TRIGger<m>:WINDow:WIDTh`.

Suffix:

<m>	1..3 1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<WidthDelta>	Range: 0 to 432 Increment: 500E-12 *RST: 0 Default unit: s

Usage: Asynchronous command

17.9.7 Timeout Trigger

TRIGger<m>:TIMeout:RANGe....................1141
TRIGger<m>:TIMeout:TIME....................1142

TRIGger<m>:TIMeout:RANGe <TimeoutMode>

Defines the relation of the signal level to the trigger level.

Suffix:

<m>	1..3 1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH
The signal level stays above the trigger level.

LOW
The signal level stays below the trigger level.

EITHer
The signal level stays above or below the trigger level.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:TIMeout:TIME <Time>

Defines the time limit for the timeout at which the instrument triggers.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

17.9.8 Interval Trigger

TRIGger<m>:INTerval:SLOPe....... 1142
TRIGger<m>:INTerval:RANGe....... 1142
TRIGger<m>:INTerval:WIDTh....... 1143
TRIGger<m>:INTerval:DELTa....... 1143

TRIGger<m>:INTerval:SLOPe <Slope>

Sets the edge for the trigger. You can analyze the interval between positive edges or between negative edges.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Slope> POSitive | NEGative

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:INTerval:RANGe <RangeMode>

Defines the range of an interval in relation to the interval width specified using `TRIGger<m>:INTerval:WIDTh` and `TRIGger<m>:INTerval:DELTa`.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range is defined by the interval width ±delta.

OUTSide
Triggers on pulses outside a given range. The range is defined by the interval width ±delta.

SHORter
Triggers on pulses shorter than the given interval width.

LONGer
Triggers on pulses longer than the given interval width.

*RST: OUTSide

Usage: Asynchronous command

TRIGger<m>:INTerval:WIDTh <Width>

Defines the time between two pulses.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Width> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:INTerval:DELTa <WidthDelta>

Defines a range around the "Interval width" value specified using `TRIGger<m>:INTerval:WIDTh` on page 1143.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<WidthDelta> Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

17.9.9 Slew Rate Trigger

TRIGger<m>:SLEW:SLOPe 1144
TRIGger<m>:LEVel<n>:SLEW:LOWer 1144
TRIGger<m>:LEVel<n>:SLEW:UPPer 1144
TRIGger<m>:SLEW:RANGe 1145
TRIGger<m>:SLEW:RATE 1145
TRIGger<m>:SLEW:DELTa 1146

TRIGger<m>:SLEW:SLOPe <Slope>

Selects the edge type for the trigger event.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:SLEW:LOWer <Level>
TRIGger<m>:LEVel<n>:SLEW:UPPer <Level>

Set the lower and upper voltage thresholds. When the signal crosses a level, the slew rate measurement starts or stops depending on the selected slope.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1...4
5...31 = not available

Parameters:

<Level> Range: -10 to 10
Increment: 1E-3
*RST: Lower = -0.1, upper =0.1
Default unit: V

Usage: Asynchronous command

TRIGger<m>:SLEW:RANGe <RangeMode>

Defines the time limit for the slew rate in relation to the upper or lower trigger level (see `TRIGger<m>:SLEW:RATE` on page 1145 and `TRIGger<m>:SLEW:DELTa` on page 1146). The time measurement starts when the signal crosses the first trigger level - the upper or lower level depending on the selected slope - and stops when the signal crosses the second level.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode> INSRange | OUTRange | LTHan | GTHan

INSRange
Triggers on pulses inside a given range. The range is defined by the slew rate ±delta.

OUTRange
Triggers on pulses outside a given range. The range is defined by the slew rate ±delta.

LTHan
Triggers on pulses shorter than the given slew rate.

GTHan
Triggers on pulses longer than the given slew rate.

*RST: GTHan

Usage: Asynchronous command

TRIGger<m>:SLEW:RATE <Time>

For the ranges "Within" and "Outside", the slew rate defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the slew rate defines the maximum and minimum slew rate limits, respectively. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope (see `TRIGger<m>:SLEW:SLOPe` on page 1144).

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-12
Default unit: s

Usage: Asynchronous command

TRIGger<m>:SLEW:DELTa <TimeDelta>

Defines a time range around the slew rate specified using `TRIGger<m>:SLEW:RATE`.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<TimeDelta> Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

17.9.10 Data2Clock Trigger

The Data2Clock trigger is only available for the A-event (Suffix = 1).

TRIGger<m>:DATatoclock:CSOurce[:VALue]....1146
TRIGger<m>:DATatoclock:CSOurce:EDGE....1146
TRIGger<m>:DATatoclock:CSOurce:LEVel....1147
TRIGger<m>:SCOupling....1147
TRIGger<m>:DATatoclock:HTIMe....1147
TRIGger<m>:DATatoclock:STIMe....1148

TRIGger<m>:DATatoclock:CSOurce[:VALue] <ClockSource>

Selects the source of the clock signal.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockSource> CHAN1 | CHANNEL1 | CHAN2 | CHANNEL2 | CHAN3 | CHANNEL3 | CHAN4 | CHANNEL4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channel of the clock signal

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:DATatoclock:CSOurce:EDGE <ClockEdge>

Sets the edge of the clock signal to define the time reference point for the setup and hold time.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockEdge> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:DATatoclock:CSOurce:LEVel <ClockLevel>

Sets the voltage level for the clock signal. Both this command and `TRIGger<m>:DATatoclock:CSOurce:EDGE` define the starting point for calculation of the setup and hold time.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockLevel> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRIGger<m>:SCOupling <TrigLevSrcCpl>

Sets the trigger levels of all used channels to the level of the data line (Data2clock and serial pattern trigger), clock line (state trigger) or source line (pattern trigger).

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<TrigLevSrcCpl> ON | OFF

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:DATatoclock:HTIMe <HoldTime>

Sets the minimum time **after** the clock edge while the data signal must stay steady above or below the data level.

The hold time can be negative. In this case, the setup time is always positive. The setup/hold interval starts before the clock edge (setup time) and ends before the clock edge (hold time). If you change the negative hold time, the setup time is adjusted by the intrument.

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<HoldTime>	Range: -99.999E-9 to 100E-9 Increment: 1E-9 *RST: 0 Default unit: s

Usage: Asynchronous command

TRIGger<m>:DATatoclock:STIMe <SetupTime>

Sets the minimum time **before** the clock edge while the data signal must stay steady above or below the data level.

The setup time can be negative. In this case, the hold time is always positive. The setup/hold interval starts after the clock edge (setup time) and ends after the clock edge (hold time). If you change the negative setup time, the hold time is adjusted by the instrument.

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<SetupTime>	Range: -99.999E-9 to 100E-9 Increment: 1E-9 *RST: 0 Default unit: s

Usage: Asynchronous command

17.9.11 State Trigger

The state trigger combines the edge trigger settings with trigger qualification. It is only available for the A-event (Suffix = 1).

Use the following commands:

- `TRIGger<m>:EDGE:SLOPe` on page 1131
- `TRIGger<m>:LEVel<n>[:VALue]` on page 1130
- `TRIGger<m>:SCOupling` on page 1147

TRIGger<m>:QUALify<n>:A[:ENABle] <State>
TRIGger<m>:QUALify<n>:B[:ENABle] <State>

TRIGger<m>:QUALify<n>:C[:ENABle] <State>
TRIGger<m>:QUALify<n>:D[:ENABle] <State>

The command is relevant for pattern trigger and for state trigger.

Select the channels to be considered:

- A[:ENABle]: CH1
- B[:ENABle]: CH3
- C[:ENABle]: CH2
- D[:ENABle]: CH4

The trigger source cannot be enabled.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Trigger type: 10 = State, 11 = Pattern, all other suffixes are not available

Parameters:

<State> ON | OFF

ON
The qualification expression is considered.

OFF
The qualification expression is ignored.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:QUALify<n>:A:LOGic <Operator>
TRIGger<m>:QUALify<n>:B:LOGic <Operator>
TRIGger<m>:QUALify<n>:C:LOGic <Operator>
TRIGger<m>:QUALify<n>:D:LOGic <Operator>

Defines the logic for the indicated channel:

- A: CH1
- B: CH3
- C: CH2
- D: CH4

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Trigger type: 10 = State, 11 = Pattern, all other suffixes are not available

Parameters:

<Operator> DIRect | NOT

DIRect
Input value remains unchanged

NOT
Input value is inverted

*RST: DIRect

Usage: Asynchronous command

TRIGger<m>:QUALify<n>:AB:LOGic <Operator>
TRIGger<m>:QUALify<n>:CD:LOGic <Operator>
TRIGger<m>:QUALify<n>:ABCD:LOGic <Operator>

Defines the logical combination of the indicated channels after evaluating the previous logical operations:

- AB: CH1 and CH3
- CD: CH2 and CH4
- ABCD: result of AB and CD

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Trigger type: 10 = State, 11 = Pattern, all other suffixes are not available

Parameters:

<Operator> AND | NAND | OR | NOR

AND
logical AND, conjunctive combination

NAND
logical NOT AND

OR
logical OR, disjunctive combination

NOR
logical NOT OR

*RST: AND

Usage: Asynchronous command

17.9.12 Pattern Trigger

The pattern trigger is only available for the A-event (Suffix = 1).

The pattern is defined using the commands:

- `TRIGger<m>:QUALify<n>:A[:ENABle]` on page 1148
- `TRIGger<m>:QUALify<n>:A:LOGic` on page 1149
- `TRIGger<m>:QUALify<n>:AB:LOGic` on page 1150

These are the commands for channel 1, use the similar commands for channels 2, 3, and 4.

TRIGger<m>:PATTern:MODE 1151
TRIGger<m>:PATTern:TIMeout:MODE 1151
TRIGger<m>:PATTern:TIMeout[:TIME] 1152
TRIGger<m>:PATTern:WIDTh:RANGe 1152
TRIGger<m>:PATTern:WIDTh[:WIDTh] 1152
TRIGger<m>:PATTern:WIDTh:DELTa 1153

TRIGger<m>:PATTern:MODE <Mode>

Adds additional time limitation to the pattern definition.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Mode> OFF | TIMeout | WIDTh

OFF
No time limitation. The event occurs if the pattern condition is fulfilled.

TIMeout
Defines how long the result of the pattern condition must be true or false. The duration of the timeout is defined using `TRIGger<m>:PATTern:TIMeout[:TIME]`.

WIDTh
Defines a time range for keeping up the true result of the pattern condition. The range is defined using `TRIGger<m>:PATTern:WIDTh:RANGe`.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:PATTern:TIMeout:MODE <TimeoutMode>

Defines the condition for the timeout.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<TimeoutMode> HIGH | LOW

HIGH
The result stays high.

LOW
The result stays low.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:PATTern:TIMeout[:TIME] <Time>

Defines how long the result of the pattern condition must be true or false.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PATTern:WIDTh:RANGe <WidthRangeMode>

Defines how the range of a pulse width is defined for keeping up the true result of the pattern condition. The width and delta are specified using `TRIGger<m>:PATTern:WIDTh[:WIDTh]` and `TRIGger<m>:PATTern:WIDTh:DELTa`, respectively.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<WidthRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range is defined by the width ±delta.

OUTSide
Triggers on pulses outside a given range. The range is defined by the width ±delta.

SHORter
Triggers on pulses shorter than the given width.

LONGer
Triggers on pulses longer than the given width.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:PATTern:WIDTh[:WIDTh] <Width>

For the ranges "Within" and "Outside" (defined using `TRIGger<m>:PATTern:WIDTh:RANGe`), the width defines the center of a range which is defined by the limits "±Delta" (see `TRIGger<m>:PATTern:WIDTh:DELTa` on page 1153).

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PATTern:WIDTh:DELTa <WidthDelta>

Defines a range around the width value specified using `TRIGger<m>:PATTern:WIDTh[:WIDTh]`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

17.9.13 Serial Pattern Trigger

The serial pattern trigger is only available for the A-event (Suffix = 1).

TRIGger<m>:SPATtern:CSOurce[:VALue]........1153
TRIGger<m>:SPATtern:CSOurce:EDGE........1154
TRIGger<m>:SPATtern:CSOurce:LEVel........1154
TRIGger<m>:SPATtern:PATTern........1154

TRIGger<m>:SPATtern:CSOurce[:VALue] <ClockSource>

Defines the source of the clock signal.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockSource> CHAN1 | CHANNEL1 | CHAN2 | CHANNEL2 | CHAN3 | CHANNEL3 | CHAN4 | CHANNEL4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channel of the clock signal

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:SPATtern:CSOurce:EDGE <ClockEdge>

Together with the clock level (see `TRIGger<m>:SPATtern:CSOurce:LEVel` on page 1154), the clock edge defines the point in time when the state of the data signal is checked.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockEdge> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:SPATtern:CSOurce:LEVel <ClockLevel>

Defines the voltage level for the clock signal.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockLevel> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRIGger<m>:SPATtern:PATTern <Pattern>

The pattern contains the bits of the serial data to be found in the data stream. The maximum length of the pattern is 128 bit.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Pattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

17.9.14 TV/Video Trigger

The TV or video trigger requires a channel input as trigger source (`TRIGger<m>:SOURce[:SELect]`). It is only available for the A-event (Suffix = 1).

Make sure to set the trigger level - the threshold of the sync pulse - with `TRIGger<m>:LEVel<n>[:VALue]`.

TRIGger<m>:TV:STANdard....1155
TRIGger<m>:TV:MODE....1156
TRIGger<m>:TV:POLarity....1156
TRIGger<m>:TV:LINE....1157
TRIGger<m>:TV:LFIeld....1158
TRIGger<m>:TV:CUSTom:SCANmode....1158
TRIGger<m>:TV:CUSTom:LDURation....1158
TRIGger<m>:TV:CUSTom:STYPe....1159
TRIGger<m>:TV:CUSTom:SDURation....1159

TRIGger<m>:TV:STANdard <Standard>

Sets the TV standard.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Standard> CUSTom | PAL | PALM | NTSC | SECam | P480L60HZ | P720L30HZ | P720L50HZ | P720L60HZ | I1080L50HZ | I1080L60HZ | P1080L24HZ | P1080L24HZSF | P1080L25HZ | P1080L30HZ | P1080L50HZ | P1080L60HZ

CUSTom
User-defined signal. Configure the signal using:
`TRIGger<m>:TV:CUSTom:SCANmode`
`TRIGger<m>:TV:CUSTom:STYPe`
`TRIGger<m>:TV:CUSTom:LDURation`
`TRIGger<m>:TV:CUSTom:SDURation`

PAL | PALM | NTSC | SECam
SDTV standards. PALM = PAL-M

PxxxxLyyHZ
HDTV standards using progressive scanning (P). xxxx indicates the number of active lines, yy is the frame rate.

IxxxxLxxHZ
HDTV standards using interlaced scanning (I). xxxx indicates the number of active lines, yy is the field rate.

P1080L24HZSF
1080p/24sF is a HDTV standard using progressive segmented frame scanning.

*RST: PAL

Usage: Asynchronous command

TRIGger<m>:TV:MODE <Mode>

Selects the lines or fields on which the instrument can trigger. Available modes depend on the scanning system.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Mode> ALL | ODD | EVEN | ALINe | LINE

ALL
All fields, triggers on the frame start (progressive scanning) or field start (interlaced and progressive segmented frame scanning)

ODD | EVEN
Only available for interlaced scanning and progressive segmented frame scanning. Triggers on the field start of the odd or even field.

ALINe
All lines, triggers on all line starts.

LINE
Triggers on a specified line. To set the line number, use `TRIGger<m>:TV:LINE`. For NTSC signals, set also the field with `TRIGger<m>:TV:LFIeld`.

*RST: ALL

Usage: Asynchronous command

TRIGger<m>:TV:POLarity <Polarity>

Sets the polarity of the *signal*. Note that the sync pulse has the opposite polarity, for example, a positive signal has a negative sync pulse.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Polarity> POSitive | NEGative
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:TV:LINE <LineNumber>

Specifies the line number to trigger on. The command is relevant if TRIGger<m>:TV:MODE is set to LINe.

Usually the lines of the frame are counted beginning from the frame start. For NTSC signals, the lines are counted per field, not per frame. For these signals, set also the field with TRIGger<m>:TV:LFIeld.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<LineNumber> Range: Depends on the standard, see table below
Increment: 1
*RST: 1

Usage: Asynchronous command

Standard	Minimum value	Maximum value
PAL	1	625
PAL-M	1	525
NTSC	1	263 in odd field 262 in even field
SECAM	1	625
480p/60 (P480L60HZ)	1	525
720p/30 (P720L30HZ) 720p/50 (P720L50HZ) 720p/60 (P720L60HZ)	1	750
1080i/50 (I1080L50HZ) 1080i/60 (I1080L60HZ) 1080p/24 (P1080L24HZ) 1080p/24sF (P1080L24HZSF) 1080p/25 (P1080L25HZ) 1080p/30 (P1080L30HZ) 1080p/50 (P1080L50HZ) 1080p/60 (P1080L60HZ)	1	1125

TRIGger<m>:TV:LFIeld <LineField>

The commands is only relevant for NTSC signals and sets the field in which the line number is counted.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<LineField> FIELD1 | FIELD2

FIELD1 = odd field
FIELD2 = even field

*RST: FIELD1

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:SCANmode <ScanMode>

Sets the scanning system. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

See also: "Scan" on page 225.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<ScanMode> INTerlaced | PROGressive | SEGMented

SEGMented: Progressive segmented frame uses progressive scanning to capture the frame, and interlaced scanning for transmission and display.

*RST: INTerlaced

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:LDURation <LinePeriod>

Sets the duration of a line, the time between two successive sync pulses. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<LinePeriod> Range: 1E-6 to 500E-6
Increment: 100E-9
*RST: 64E-6
Default unit: s

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:STYPe <SyncPulseType>

Sets the type of the sync pulse. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<SyncPulseType> BILevel | TRILevel

BILevel
Bi-level sync pulse, usually used in SDTV signals

TRILevel
Tri-level sync pulse, used in HDTV signals

*RST: BILevel

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:SDURation <SyncPulseDurat>

Sets the width of the sync pulse. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<SyncPulseDurat> Range: 100E-9 to 100E-6
Increment: 100E-9
*RST: 4.7E-6
Default unit: s

Usage: Asynchronous command

17.9.15 Line Trigger

To select the line trigger, set `TRIGger<m>:SOURce[:SELect]` to `LINE`.

TRIGger<m>:POWerline:SLOPe <Slope>

Selects the rising or falling edges of the AC power input for the trigger condition.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Slope> POSitive | NEGative

*RST: POSitive

Usage: Asynchronous command

17.9.16 Holdoff

TRIGger<m>:HOLDoff:MODE....1160
TRIGger<m>:HOLDoff:TIME....1161
TRIGger<m>:HOLDoff:EVENts....1161
TRIGger<m>:HOLDoff:MIN....1161
TRIGger<m>:HOLDoff:MAX....1162
TRIGger<m>:HOLDoff:AUTotime?....1162
TRIGger<m>:HOLDoff:SCALing....1163

TRIGger<m>:HOLDoff:MODE <Mode>

Selects the method to define the holdoff condition.

The trigger holdoff defines when the next trigger after the current will be recognized. Thus, it affects the next trigger to occur after the current one. Holdoff helps to obtain stable triggering when the oscilloscope is triggering on undesired events.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Mode> TIME | EVENts | RANDom | AUTO | OFF

TIME
Defines the holdoff directly as a time period. The next trigger occurs only after the "Holdoff time" has passed (defined using `TRIGger<m>:HOLDoff:TIME`).

EVENts
Defines the holdoff as a number of trigger events. The next trigger occurs only when this number of events is reached. The number of triggers to be skipped is defined using `TRIGger<m>:HOLDoff:EVENts`.

RANDom
Defines the holdoff as a random time limited by `TRIGger<m>:HOLDoff:MIN` on page 1161 and `TRIGger<m>:HOLDoff:MAX` on page 1162. For each acquisition cycle, the instrument selects a new random holdoff time from the specified range.

AUTO
The holdoff time is calculated automatically based on the current horizontal scale.

OFF
No holdoff

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:HOLDoff:TIME <Time>

Defines the holdoff time period. The next trigger occurs only after this time has passed. The setting is relevant if the holdoff mode is set to TIME.

See also:

- `TRIGger<m>:HOLDoff:MODE`

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Time> Range: 100E-9 to 10
Increment: 200E-6
*RST: 1E-3
Default unit: s

Example:
```
TRIGger:HOLDoff:MODE TIME
TRIGger:HOLDoff:TIME 1ms
```
The holdoff time is set to 1 ms.

Usage: Asynchronous command

TRIGger<m>:HOLDoff:EVENts <Events>

Defines the number of triggers to be skipped. The next trigger only occurs when this number of events is reached. The setting is relevant if the holdoff mode is set to EVENts.

See also:

- `TRIGger<m>:HOLDoff:MODE`

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Events> Range: 1 to 2147483647
Increment: 1
*RST: 1

Example:
```
TRIGger:HOLDoff:MODE EVENts
TRIGger:HOLDoff:EVENts 5
```

Usage: Asynchronous command

TRIGger<m>:HOLDoff:MIN <RandomMinTime>

Defines the lower limit for the random time holdoff. The setting is relevant if the holdoff mode is set to RANDom.

See also:

- `TRIGger<m>:HOLDoff:MODE`

- TRIGger<m>:HOLDoff:MAX

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<RandomMinTime>	Range: 100E-9 to 5 Increment: 200E-6 *RST: 1E-3 Default unit: s

Example:

```
TRIGger:HOLDoff:MODE RANDom
TRIGger:HOLDoff:MIN 1ms
TRIGger:HOLDoff:MAX 2ms
```

The holdoff time is set randomly between 1 ms and 2 ms.

Usage: Asynchronous command

TRIGger<m>:HOLDoff:MAX <RandomMaxTime>

Defines the upper limit for the random time holdoff. The setting is relevant if the holdoff mode is set to RANDom.

See also:

- TRIGger<m>:HOLDoff:MODE
- TRIGger<m>:HOLDoff:MIN

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<RandomMaxTime>	Range: 100E-9 to 10 Increment: 200E-6 *RST: 2E-3 Default unit: s

Usage: Asynchronous command

TRIGger<m>:HOLDoff:AUTotime?

Returns the resulting holdoff time if the holdoff mode is set to AUTO: *Auto time = Auto time scaling * Horizontal scale*. The auto time scaling factor is defined with TRIGger<m>:HOLDoff:SCALing.

See also: TRIGger<m>:HOLDoff:MODE

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Return values:

<AutoTime> Holdoff time

Range: 100E-9 to 10
*RST: 1E-3
Default unit: s

Example:
```
TRIGger:HOLDoff:MODE AUTO
TRIGger:HOLDoff:SCALing 0.5
TRIGger:HOLDoff:AUTotime?
1ms
```
Result if the horizontal scale is 1 ns/div

Usage: Query only
Asynchronous command

TRIGger<m>:HOLDoff:SCALing <AutoTimeScaling>

Sets the auto time scaling factor the horizontal scale is multipied with: *Auto time = Auto time scaling * Horizontal scale*. The setting is relevant if the holdoff mode is set to AUTO.

See also:
- `TRIGger<m>:HOLDoff:MODE`
- `TRIGger<m>:HOLDoff:AUTotime?` on page 1162

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<AutoTimeScaling> Range: 1E-3 to 1000
Increment: 1
*RST: 0.5

Usage: Asynchronous command

17.9.17 Noise Reject

TRIGger<m>:LEVel<n>:NOISe[:STATe] ... 1163
TRIGger<m>:LEVel<n>:NOISe:MODE ... 1164
TRIGger<m>:LEVel<n>:NOISe:ABSolute ... 1165
TRIGger<m>:LEVel<n>:NOISe:PERDivision ... 1165
TRIGger<m>:LEVel<n>:NOISe:RELative ... 1165
TRIGger<m>:ANEDge:NREJect ... 1166

TRIGger<m>:LEVel<n>:NOISe[:STATe] <HysteresisMode>

Selects how the hysteresis is set.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source:
1...4 = channel 1 to 4
5 = external trigger input
6...9 and 11 = not available
10 = line trigger
12...27 = R&S RT-ZVC input channels
28...31 = not available

Parameters:

<HysteresisMode> AUTO | MANual

AUTO
This is the recommended mode. The hysteresis is set by the instrument to reject at least the internal noise of the instrument. You can define a higher minimum value using `TRIGger<m>:LEVel<n>:NOISe:ABSolute`.

MANual
The hysteresis is defined directly with `TRIGger<m>:LEVel<n>:NOISe:ABSolute`.

*RST: AUTO

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:MODE <HystMode>

Selects how the hysteresis is set.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1163.

Parameters:

<HystMode> ABS | REL

ABS
The hysteresis is set in absolute values (voltage).

REL
The hysteresis is defined in relative values (div).

*RST: ABS

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:ABSolute <HystAbs>

Defines a range in absolute values around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1163.

Parameters:

<HystAbs>
Range: 0 to The value corresponding to full division range. The exact maximum value depends on the selected vertical scale.
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:PERDivision <HystInDivs>

Defines a range in divisions around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1163.

Parameters:

<HystInDivs> Hysteresis size in divisions
Range: 0 to 5
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:RELative <HystRel>

Defines a range in percent around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n>	1..31 Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1163.

Parameters:

<HystRel>	Hysteresis in %. 10% = 1 div Range: 0 to 50 Increment: 1 *RST: 0 Default unit: %

Usage: Asynchronous command

TRIGger<m>:ANEDge:NREJect <NoiseReject>

Enables the noise reject for the external trigger input.

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<NoiseReject>	ON | OFF *RST: OFF

Usage: Asynchronous command

Firmware/Software: FW 2.25

17.9.18 Trigger Sequence

TRIGger<m>:SEQuence:MODE 1166
TRIGger<m>:ECOupling 1167
TRIGger<m>:SEQuence:DELay 1167
TRIGger<m>:SEQuence:COUNt 1167
TRIGger<m>:SEQuence:RESet:EVENt 1168
TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle] 1168
TRIGger<m>:SEQuence:RESet:TIMeout:TIME 1168

TRIGger<m>:SEQuence:MODE <Type>

Selects the type of the sequence.

See also: Chapter 5.7, "Sequence", on page 232.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Parameters:

<Type>	AONLy | AB

AONLy
Triggers only on A-events. Additionally, a holdoff condition can be set. if AONLy sequence is set, all inputs (input channels, serial and parallel buses, digital channels etc.) can be used as trigger source.

AB
Triggers if all conditions of A- and B-events, as well as additional delay and waiting conditions are fulfilled. This trigger sequence requires that analog input channels CHAN1...4 are set as trigger sources for all events.

*RST: AONLy

Usage: Asynchronous command

TRIGger<m>:ECOupling <TrigLevEvtCpl>

Event coupling of trigger levels: Sets the trigger levels to the values of the indicated event. Thus, channel 1 has one trigger level for all events, channel 2 has one trigger level and so on.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<TrigLevEvtCpl> ON | OFF
*RST: ON

Usage: Asynchronous command

TRIGger<m>:SEQuence:DELay <Delay>

Sets the time the instrument waits after an A-event until it recognizes B-events.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Delay> Range: 0 to 50
Increment: 1E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

TRIGger<m>:SEQuence:COUNt <Events>

Sets the number of B-events to be fulfilled after an A-event. The last B-event causes the trigger.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

Parameters:

<Events> Range: 1 to 2147483647
Increment: 1
*RST: 1

Usage: Asynchronous command

TRIGger<m>:SEQuence:RESet:EVENt <EnabRstEvt>

If set to ON, the trigger sequence is restarted by the R-event if the specified number of B-event does not occur.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

Parameters:

<EnabRstEvt> ON | OFF
*RST: OFF

Usage: Asynchronous command

TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle] <State>

If set to ON, the instrument waits for the time defined using `TRIGger<m>:SEQuence:RESet:TIMeout:TIME` for the specified number of B-events. If no trigger occurs during that time, the sequence is restarted with the A-event.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

TRIGger<m>:SEQuence:RESet:TIMeout:TIME <ResetTimeout>

The time the instrument waits for the number of B-events specified using `TRIGger<m>:SEQuence:COUNt` before the sequence is restarted with the A-event.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

Parameters:

<ResetTimeout> Range: 0 to 50
Increment: 1E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

17.9.19 Trigger Control

TRIGger<m>:MODE....1169
TRIGger<m>:FORCe....1169
TRIGger<m>:OUT:STATe....1170
TRIGger<m>:OUT:POLarity....1170
TRIGger<m>:OUT:PLENgth....1170
TRIGger<m>:OUT:DELay....1171

TRIGger<m>:MODE <TriggerMode>

Sets the trigger mode which determines the behaviour of the instrument if no trigger occurs.

See also: "Trigger mode" on page 229

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

Parameters:

<TriggerMode> AUTO | NORMal | FREerun

AUTO
The instrument triggers repeatedly after a time interval if the trigger conditions are not fulfilled. If a real trigger occurs, it takes precedence. The time interval depends on the time base.

NORMal
The instrument acquires a waveform only if a trigger occurs.

FREerun
The instrument triggers after a very short time interval - faster than in AUTO mode. Real triggers are ignored

*RST: AUTO

Usage: Asynchronous command

TRIGger<m>:FORCe

If the acquisition is running in normal mode and no valid trigger occurs, forcing the trigger provokes an immediate single acquisition. Thus you can confirm that a signal is available and use the waveform display to determine how to trigger on it.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Usage: Event
Asynchronous command

TRIGger<m>:OUT:STATe <State>

Enables/disbales the trigger out signal that is provided to the [TRIGGER OUTPUT] connector on the rear panel when a trigger occurs.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Parameters:

<State>	ON | OFF *RST: OFF

Usage: Asynchronous command

TRIGger<m>:OUT:POLarity <Polarity>

Sets the polarity of the trigger out pulse.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Parameters:

<Polarity>	POSitive | NEGative *RST: POSitive

Usage: Asynchronous command

TRIGger<m>:OUT:PLENgth <PulseLength>

Sets the length of the trigger out pulse.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Parameters:

<PulseLength>	Range: 4E-9 to 0.06 Increment: 20E-9 *RST: 100E-9 Default unit: s

Usage: Asynchronous command

TRIGger<m>:OUT:DELay <Delay>

Sets the delay of the first pulse edge to the trigger point.

The settin gis not available if a mask test or measurement is running and the on-violation event is set to trigger out pulse.

Suffix:

<m>	1..3 Irrelevant, omit the suffix.

Parameters:

<Delay>	Range: 800E-9 to 1 Increment: 1E-9 *RST: 800E-9 Default unit: s

Usage: Asynchronous command

17.9.20 Actions on Trigger

TRIGger<m>:EVENt:BEEP 1171
TRIGger<m>:EVENt:PRINt 1171
TRIGger<m>:EVENt:WFMSave 1172
TRIGger<m>:EVENt:RUNexec 1172

TRIGger<m>:EVENt:BEEP <Beep>

Generates a beep sound if the command is set to `TRIGger`.

Suffix:

<m>	1..3 The suffix is irrelevant.

Parameters:

<Beep>	NOAction | TRIGger *RST: NOAction

Usage: Asynchronous command

TRIGger<m>:EVENt:PRINt <Print>

Saves a screenshot at each trigger if the command is set to `TRIGger`.

For screenshot settings, see Chapter 17.16.10, "Screenshots", on page 1374.

Suffix:

<m>	1..3 The suffix is irrelevant.

Parameters:

<Print>	NOAction | TRIGger *RST: NOAction

Usage: Asynchronous command

TRIGger<m>:EVENt:WFMSave <SaveWfm>

Saves the waveform data to file at each trigger if the command is set to `TRIGger`.

For data export settings, see Chapter 17.16.5, "Waveform Data Export to File", on page 1362.

Suffix:
<m> 1..3
The suffix is irrelevant.

Parameters:
<SaveWfm> NOACtion | TRIGger
*RST: NOACtion

Usage: Asynchronous command

TRIGger<m>:EVENt:RUNexec <RunExecutable>

Starts an external application if the command is set to `TRIGger`.

Use the following commands to set up the application:

- `EXECutable:NAME` on page 1069
- `EXECutable:PARameter` on page 1069
- `EXECutable:WDIRectory` on page 1069

Suffix:
<m> 1..3
The suffix is irrelevant.

Parameters:
<RunExecutable> NOACtion | TRIGger
*RST: NOACtion

Usage: Asynchronous command

17.9.21 External Trigger Input

To control the external trigger signal, the TRPRobe commands are used. The required commands depend on the used probe type. They work in the same way as the PROBe commands. For details, see Chapter 17.8.7, "Probes", on page 1086

TRIGger<m>:EXTern:OVERload <Overload>

The query returns the overload status of the external trigger input.

`:TRIGger:EXTern:OVERload 0` confirms the information in the message box, it has same effect as OK.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Overload> ON | OFF

ON | 1
Iindicates an overload of the probe

OFF | 0
In a query: no overlaod.
Ssetting: confirms the information in the message box

*RST: OFF

Usage: Asynchronous command

17.9.22 Acquisition Info

ACQuire:CURRent?

Shows the current number of acquisitions that have been acquired.

Return values:
<CurrAcqCnt> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

17.10 Waveform Analysis

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- Zoom....................1174
- Reference Waveforms....................1183
- Mathematics....................1189
- History....................1194
- XY-Diagram....................1198

17.10.1 Zoom

LAYout:ZOOM:ADD.......... 1174
LAYout:ZOOM:ADDCoupled.......... 1175
LAYout:ZOOM:ONEDiagram.......... 1175
LAYout:ZOOM:POSCoupling.......... 1175
LAYout:ZOOM:HORZ:MODE.......... 1176
LAYout:ZOOM:HORZ:ABSolute:POSition.......... 1176
LAYout:ZOOM:HORZ:ABSolute:SPAN.......... 1176
LAYout:ZOOM:HORZ:ABSolute:STARt.......... 1177
LAYout:ZOOM:HORZ:ABSolute:STOP.......... 1177
LAYout:ZOOM:HORZ:RELative:POSition.......... 1178
LAYout:ZOOM:HORZ:RELative:SPAN.......... 1178
LAYout:ZOOM:HORZ:RELative:STARt.......... 1178
LAYout:ZOOM:HORZ:RELative:STOP.......... 1179
LAYout:ZOOM:VERTical:MODE.......... 1179
LAYout:ZOOM:VERTical:ABSolute:POSition.......... 1179
LAYout:ZOOM:VERTical:ABSolute:SPAN.......... 1180
LAYout:ZOOM:VERTical:ABSolute:STARt.......... 1180
LAYout:ZOOM:VERTical:ABSolute:STOP.......... 1181
LAYout:ZOOM:VERTical:RELative:POSition.......... 1181
LAYout:ZOOM:VERTical:RELative:SPAN.......... 1181
LAYout:ZOOM:VERTical:RELative:STARt.......... 1182
LAYout:ZOOM:VERTical:RELative:STOP.......... 1182
LAYout:ZOOM:REMove.......... 1182

LAYout:ZOOM:ADD <NodeName>, <ParentType>, <InsertBefore>, <XStart>, <XStop>, <YStart>, <YStop>, <NewZoomName>

Adds a new zoom diagram based on the specified waveform.

For numerical values, the input of a unit is not supported for this command.

Setting parameters:

<NodeName> String with the name of diagram to be zoomed

<ParentType> VERTical, OFF
The new zoom diagram is displayed below the original one.

<InsertBefore> OFF
Position of the zoom diagram, depending on `ParentType`

<XStart> Defines the x-value at the beginning of the zoom area.

<XStop> Defines the x-value at the end of the zoom area.

<YStart> Defines the y-value at the beginning of the zoom area.

<YStop> Defines the y-value at the end of the zoom area.

<NewZoomName> String with the name of the new zoom diagram.

Example:
```
LAYout:ZOOM:ADD 'Diagram1', VERT, OFF, -10e-9,
20e-9, -0.1, 0.05, 'MyZoom1'
```
Creates the zoom diagram 'MyZoom1' for 'Diagram1'.

Example: See Chapter 17.3.1.2, "Creating Zoom Diagrams", on page 1019

Usage: Setting only
Asynchronous command

LAYout:ZOOM:ADDCoupled <ZoomName>, <XOffset>, <YOffset>, <NewZoomName>

Creates a new zoom diagram based on the settings of an existing zoom area for the same source.

For numerical values, the input of a unit is not supported for this command.

Parameters:
<NewZoomName> Defines the name of the new zoom diagram.

Setting parameters:
<ZoomName> Defines the name of the zoom diagram to be copied.

<XOffset> Defines an offset to the existing zoom area in x direction.

<YOffset> Defines an offset to the existing zoom area in y direction.

Usage: Asynchronous command

LAYout:ZOOM:ONEDiagram <ShowInOne>

Shows all zooms of a diagram in one zoom window. The zoomed areas are overlaid for better comparison of the zoomed waveforms.

The command takes effect on all zoom diagrams.

Parameters:
<ShowInOne> ON | OFF
*RST: OFF

Usage: Asynchronous command

LAYout:ZOOM:POSCoupling <DiagramName>, <ZoomName>, <PositionCoupl>
LAYout:ZOOM:POSCoupling? <DiagramName>, <ZoomName>

Enables or disables the position coupling of coupled zooms. If position coupling is enabled and one zoom area is moved, the other coupled zoom areas are moved, too, and keep their distance.

Parameters:
<PositionCoupl> ON | OFF
*RST: OFF

Parameters for setting and query:

<DiagramName>	String with the name of the diagram on which the zoom is based
<ZoomName>	String with the name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:MODE <DiagramName>, <ZoomName>,<Mode>
LAYout:ZOOM:HORZ:MODE? <DiagramName>, <ZoomName>

Defines whether absolute or relative values are used to specify the x-axis values. Since the zoom area refers to the active signal, relative values ensure that the zoom area remains the same.

Parameters:

<Mode> ABS | REL

Mode used to specify the x-axis values of the zoom area.

*RST: ABS

Parameters for setting and query:

<DiagramName>	Name of the diagram on which the zoom area is based.
<ZoomName>	Name of the zoom diagram
Example:	See Chapter 17.3.1.2, "Creating Zoom Diagrams", on page 1019
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:POSition <DiagramName>, <ZoomName>,<Position>
LAYout:ZOOM:HORZ:ABSolute:POSition? <DiagramName>, <ZoomName>

Defines the x-value of the centerpoint of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Position>
Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName>	Name of the diagram on which the zoom area is based.
<ZoomName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:SPAN <DiagramName>, <ZoomName>,<Span>
LAYout:ZOOM:HORZ:ABSolute:SPAN? <DiagramName>, <ZoomName>

Defines the width of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Span>	Range: 0 to 100E+24 Increment: 0.01 *RST: 0.01

Parameters for setting and query:

<DiagramName>	Name of the diagram on which the zoom area is based.
<ZoomName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:STARt <DiagGroupNme>, <ZoomDiagramName>,<Start>
LAYout:ZOOM:HORZ:ABSolute:STARt? <DiagGroupNme>, <ZoomDiagramName>

Defines the lower limit of the zoom area on the x-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start>	Range: -100E+24 to 100E+24 Increment: 0.01 *RST: 0

Parameters for setting and query:

<DiagGroupNme>	Name of the diagram on which the zoom area is based.
<ZoomDiagramName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:STOP <DiagramName>, <ZoomName>,<Stop>
LAYout:ZOOM:HORZ:ABSolute:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the x-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop>	Range: -100E+24 to 100E+24 Increment: 0.01 *RST: 0.01

Parameters for setting and query:

<DiagramName>	Name of the diagram on which the zoom area is based.
<ZoomName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:RELative:POSition <DiagramName>, <ZoomName>,<RelPosi>
LAYout:ZOOM:HORZ:RELative:POSition? <DiagramName>, <ZoomName>

Defines the x-value of the centerpoint of the zoom area.

Parameters:

<RelPosi> Relative position of the centerpoint (x-value)

Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 17.3.1.2, "Creating Zoom Diagrams", on page 1019

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:SPAN <DiagramName>, <ZoomName>,<RelativeSpan>
LAYout:ZOOM:HORZ:RELative:SPAN? <DiagramName>, <ZoomName>

Defines the width of the zoom area.

Parameters:

<RelativeSpan> Range: 1E-15 to 100
Increment: 0.1
*RST: 1
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 17.3.1.2, "Creating Zoom Diagrams", on page 1019

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:STARt <DiagramName>, <ZoomName>,<RelativeStart>
LAYout:ZOOM:HORZ:RELative:STARt? <DiagramName>, <ZoomName>

Defines the lower limit of the zoom area on the x-axis.

Parameters:

<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:STOP <DiagramName>, <ZoomName>,<RelativeStop>
LAYout:ZOOM:HORZ:RELative:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the x-axis.

Parameters:

<RelativeStop>
Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:MODE <DiagramName>, <ZoomName>,<Mode>
LAYout:ZOOM:VERTical:MODE? <DiagramName>, <ZoomName>

Defines whether absolute or relative values are used to specify the y-axis values. Since the zoom area refers to the active signal, relative values ensure that the zoom area remains the same.

Parameters:

<Mode> ABS | REL

Mode used to specify the y-axis values of the zoom area.

*RST: ABS

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:POSition <DiagramName>, <ZoomName>,<Position>
LAYout:ZOOM:VERTical:ABSolute:POSition? <DiagramName>, <ZoomName>

Defines the y-value of the centerpoint of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Position> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:SPAN <DiagramName>, <ZoomName>,<Span>
LAYout:ZOOM:VERTical:ABSolute:SPAN? <DiagramName>, <ZoomName>

Defines the height of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Span> Range: 0 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:STARt <DiagGroupNme>, <ZoomDiagramName>,<Start>
LAYout:ZOOM:VERTical:ABSolute:STARt? <DiagGroupNme>, <ZoomDiagramName>

Defines the lower limit of the zoom area on the y-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Parameters for setting and query:

<DiagGroupNme> Name of the diagram on which the zoom area is based.

<ZoomDiagramName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:STOP <DiagramName>, <ZoomName>,<Stop>
LAYout:ZOOM:VERTical:ABSolute:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the y-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:POSition <DiagramName>, <ZoomName>,<RelPosi>
LAYout:ZOOM:VERTical:RELative:POSition? <DiagramName>, <ZoomName>

Defines the y-value of the centerpoint of the zoom area.

Parameters:

<RelPosi> Relative position of the centerpoint (y-value)

Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:SPAN <DiagramName>, <ZoomName>,<RelativeSpan>
LAYout:ZOOM:VERTical:RELative:SPAN? <DiagramName>, <ZoomName>

Defines the height of the zoom area.

Parameters:

<RelativeSpan> Range: 1E-15 to 100
Increment: 0.1
*RST: 1
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:STARt <DiagramName>, <ZoomName>,<RelativeStart>
LAYout:ZOOM:VERTical:RELative:STARt? <DiagramName>, <ZoomName>

Defines the lower limit of the zoom area on the y-axis.

Parameters:

<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:STOP <DiagramName>, <ZoomName>,<RelativeStop>
LAYout:ZOOM:VERTical:RELative:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the y-axis.

Parameters:

<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:REMove <DiagramName>, <ZoomName>

Removes the specified zoom diagram.

Setting parameters:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 17.3.1.2, "Creating Zoom Diagrams", on page 1019

Usage: Setting only
Asynchronous command

17.10.2 Reference Waveforms

- Reference....1183
- Scaling....1185
- Waveform Data Export....1186
- Import of Multichannel Waveform Data....1188

17.10.2.1 Reference

REFCurve<m>:SOURce....1183
REFCurve<m>:STATe....1183
REFCurve<m>:NAME....1184
REFCurve<m>:OPEN....1184
REFCurve<m>:UPDate....1184
REFCurve<m>:SAVE....1184
REFCurve<m>:DELete....1184
REFCurve<m>:CLEar....1185

REFCurve<m>:SOURce <Source>

Selects the source waveform to be used as a reference.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | R1 | R2 | R3 | R4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of the reference waveform, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

REFCurve<m>:STATe <State>

If enabled, the reference waveform is displayed in the diagram.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

REFCurve<m>:NAME <Name>

Defines the name of the reference waveform file to be loaded, saved or deleted.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<Name> Path and name of the file that contains the reference waveform or to which the reference waveform is to be stored (`.xml` or `.bin` format), enclosed in single quotes.

Usage: Asynchronous command

REFCurve<m>:OPEN

Loads the reference waveform file selected by `REFCurve<m>:NAME` on page 1184.

Suffix:
<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:UPDate

Copies the selected source waveform (see `REFCurve<m>:SOURce` on page 1183) with all its settings to the memory of the reference waveform.

Suffix:
<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:SAVE

Saves the reference waveform to the file selected by `REFCurve<m>:NAME` on page 1184.

Suffix:
<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:DELete

Deletes the reference waveform file selected by `REFCurve<m>:NAME` on page 1184.

Suffix:

<m>	1..4 Reference waveform

Usage: Event

REFCurve<m>:CLEar

The selected reference waveform is no longer displayed, its memory is deleted.

Suffix:

<m>	1..4 Reference waveform

Usage: Event
Asynchronous command

17.10.2.2 Scaling

REFCurve<m>:RESCale:VERTical:STATe........ 1185
REFCurve<m>:RESCale:VERTical:OFFSet........ 1185
REFCurve<m>:RESCale:HORizontal:STATe........ 1186
REFCurve<m>:RESCale:HORizontal:OFFSet........ 1186

REFCurve<m>:RESCale:VERTical:STATe <State>

Enables and disables the vertical stretching. Stretching changes the display of the waveform independent of the vertical scale and position.

Suffix:

<m>	1..4 Reference waveform

Parameters:

<State>	ON | OFF *RST: OFF

Usage: Asynchronous command

REFCurve<m>:RESCale:VERTical:OFFSet <Offset>

Moves the reference waveform vertically. Like vertical offset of channel waveforms, the offset of a reference waveform is subtracted from the measured value.

Suffix:

<m>	1..4 Reference waveform

Parameters:

<Offset> Negative values shift the waveform up, positive values shift it down.

Range: -100E+24 to 100E+24
Increment: 1E-6
*RST: 0
Default unit: V

Usage: Asynchronous command

REFCurve<m>:RESCale:HORizontal:STATe <State>

Enables and disables the horizontal stretching.

Stretching changes the display of the waveform independent of the horizontal settings of the source waveform and of the horizontal diagram settings.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

REFCurve<m>:RESCale:HORizontal:OFFSet <Offset>

Moves the waveform horizontally. Positive values shift the waveform to the right, negative values shift it to the left.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<Offset> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0
Default unit: s

Usage: Asynchronous command

17.10.2.3 Waveform Data Export

Commands for saving waveform data to file are described in Chapter 17.16.5, "Waveform Data Export to File", on page 1362.Commands for reference waveforms are listed below.

REFCurve<m>:DATA:STYPe?..1187
REFCurve<m>:DATA:HEADer?...1187
REFCurve<m>:DATA[:VALues]?...1187

REFCurve<m>:DATA:STYPe?

Returns the signal type of the source of the reference waveform.

Suffix:
<m> 1..4
Reference waveform

Return values:
<SignalType> SOUR | SPEC | CORR | NONE
SOURce = normal signal
SPECtrum = FFT spectrum, specific math signal
CORRelation = correlated signal, specific math signal
NONE = undefined

Usage: Query only

REFCurve<m>:DATA:HEADer?

Returns information on the reference waveform.

Table 17-3: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of the waveform in Samples	200000
4	Number of values per sample interval. For reference waveforms the number is always 1.	1

Suffix:
<m> 1..4
Reference waveform

Example:
```
REFC:DATA:HEAD?
-9.477E-008,9.477E-008,200000,1
```

Usage: Query only

REFCurve<m>:DATA[:VALues]?

Returns the data of the channel waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values, or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define this.

Suffix:
<m> 1..4
Reference waveform

Return values:
<Data> List of values according to the format and content settings.

Usage: Query only

17.10.2.4 Import of Multichannel Waveform Data

Commands for saving waveform data to file are described in Chapter 17.16.5, "Waveform Data Export to File", on page 1362.Commands for reference waveforms are listed below.

REFCurve<m>:MULTichannel:NAME ... 1188
REFCurve<m>:MULTichannel:IMPort ... 1188
REFCurve<m>:MULTichannel:OPEN ... 1189

REFCurve<m>:MULTichannel:NAME <MltChImptPath>

Defines the path and the file to be imported. If not path is given, the default path `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms` is used.

Suffix:
<m> 1..4
The suffix is irrelevant.

Parameters:
<MltChImptPath> String with path and filename

Example: See `REFCurve<m>:MULTichannel:IMPort` on page 1188.

Usage: Asynchronous command

REFCurve<m>:MULTichannel:IMPort <MltChImptWfmSel>

Assigns a waveform from the multichannel file to a reference waveform. To import all waveforms that are in the file, you must assign each waveform to another reference waveform.

Suffix:
<m> 1..4
Selects the reference waveform.

Parameters:
<MltChImptWfmSel> NONE | NONE | WF1 | WAVeform1 | WF2 | WAVeform2 | WF3 | WAVeform3 | WF4 | WAVeform4

WF1 = WAVeform1, WF2 = WAVeform2, WF3 = WAVeform3, WF4 = WAVeform4
Selects the waveform in the export file.

*RST: NONE

Example: Prerequisite: file `RefCurve_2016-02-16_01.bin` is a multi-channel export with at least WF1, WF3 and WF4.

```
REFCurve:MULTichannel:NAME 'RefCurve_2016-02-16_01.bin'
REFCurve1:MULTichannel:IMPort WF1
REFCurve2:MULTichannel:IMPort WF3
REFCurve3:MULTichannel:IMPort WF4
REFCurve:MULTichannel:OPEN
```

Usage: Asynchronous command

Firmware/Software: FW 3.20

REFCurve<m>:MULTichannel:OPEN

Loads the waveform data to the reference waveforms.

Suffix:

<m> 1..4
The suffix is irrelevant.

Example: See `REFCurve<m>:MULTichannel:IMPort` on page 1188.

Usage: Event
Asynchronous command

17.10.3 Mathematics

CALCulate:MATH<m>[:EXPRession][:DEFine] ... 1189
CALCulate:MATH<m>:STATe ... 1190
CALCulate:MATH<m>:ENVSelection ... 1190
CALCulate:MATH<m>:ARIThmetics ... 1190
CALCulate:MATH<m>:VERTical:OFFSet ... 1191
CALCulate:MATH<m>:VERTical:RANGe ... 1192
CALCulate:MATH<m>:VERTical:SCALe ... 1192
CALCulate:MATH<m>:DATA:STYPe? ... 1192
CALCulate:MATH<m>:DATA:HEADer? ... 1193
CALCulate:MATH<m>:DATA[:VALues]? ... 1193

CALCulate:MATH<m>[:EXPRession][:DEFine] <RemComplExpr>

Defines the math expression to be calculated for the specified math channel.

For an overview of corresponding expressions for the available keys in the formula editor, see Chapter 6.3.4, "Advanced Expressions", on page 263.

Suffix:

<m> 1..8
Selects the math waveform.

Parameters:

<RemComplExpr> String with regular expression for calculation

Example: `CALC:MATH 'Ch1Wfm1*Ch2Wfm1'`
Defines the multiplication of waveforms Ch1Wfm1 and Ch2Wfm1.

Usage: Asynchronous command

CALCulate:MATH<m>:STATe <State>

Enables the math waveform display.

Suffix:
<m> 1..8
math waveform

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:ENVSelection <EnvelopeCurve>

Selects the upper or lower part of the waveform for mathematic calculation, or a combination of both.

The setting is relevant for waveforms with waveform arithmetic mode "Envelope" or with "Peak detect" decimation. All mathematic operations - except for derivation - can be applied to envelope waveforms and waveforms with "Peak detect" decimation.

Suffix:
<m> 1..8
Selects the math waveform.

Parameters:
<EnvelopeCurve> MIN | MAX | BOTH
*RST: BOTH

Usage: Asynchronous command

Firmware/Software: FW 2.25

CALCulate:MATH<m>:ARIThmetics <Arithmetics>

Selects the method to build the resulting math waveform from consecutive acquisitions. The processing is similar to the waveform arithmetics - instead of the acquired waveforms the results of the mathematic formula are used to create envelope, average and RMS.

Suffix:

<m> 1..8
Selects the math waveform.

Parameters:

<Arithmetics> OFF | ENVelope | AVERage | RMS | MINHold | MAXHold

Waveform arithmetic mode

OFF
The math waveform is built according to the mathematic formula.

ENVelope
Detects the minimum and maximum math values in a sample interval over a number of acquisitions.

AVERage
Calculates the average from the math data of the current acquisition and a number of acquisitions before. To define the number of acquisitions, use `ACQuire:COUNt`.

RMS
The resulting math waveform is the root mean square of the current acquisition and a number of acquisitions before. The result is the average power spectrum. Number of acquisitions: `ACQuire:COUNt`

MAXHold
Determines the maximum result for each input value from the math data of the current acquisition and a number of acquisitions before. To define the number of acquisitions, use `ACQuire:COUNt`.

MINHold
Determines the minimum result for each input value from the math data of the current acquisition and a number of acquisitions before. To define the number of acquisitions, use `ACQuire:COUNt`.

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:OFFSet <VerticalOffset>

Sets a voltage offset to adjust the vertical position of the math function on the screen.

Suffix:

<m> 1..8
Math waveform

Parameters:

<VerticalOffset> Negative values move the waveform au, positive values move it down.

Range: -100E+12 to 100E+12
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:RANGe <VerticalRange>

Defines the range of FFT values to be displayed.

Suffix:

<m> 1..8
Math waveform

Parameters:

<VerticalRange> Range: 0 to 1E+15
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:SCALe <VerticalScale>

Defines the scale of the y-axis in the math function diagram. The value is defined as "V per division", e.g. *50V/div*. In this case, the horizontal grid lines are displayed in intervals of 50 V.

Suffix:

<m> 1..8
Math waveform

Parameters:

<VerticalScale> Range: 1E-12 to 100E+12
Increment: 10E-6
*RST: 0.5
Default unit: V/div

Usage: Asynchronous command

CALCulate:MATH<m>:DATA:STYPe?

Returns the signal type of the source of the math waveform.

Suffix:

<m> 1..8
Selects the math waveform.

Return values:

<SignalType> SOUR | SPEC | CORR | MEAS | NONE

SOURce = normal signal
SPECtrum = FFT spectrum, specific math signal
CORRelation = correlated signal, specific math signal
MEAsurement = result of a measurement
NONE = undefined

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:DATA:HEADer?

Returns the header of math waveform data. The header contains attributes of the waveform.

Table 17-4: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of the waveform in Samples	200000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of samples (record length).	1

Suffix:

<m> 1..8
Selects the math waveform.

Example:

```
CALC:MATH1:DATA:HEAD?
-9.477E-008,9.477E-008,200000,1
```

Usage: Query only

CALCulate:MATH<m>:DATA[:VALues]?

Returns the data of the math waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values, or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define this.

Suffix:

<m> 1..8
Selects the math waveform.

Return values:

<Data> List of values according to the format and content settings.

Usage: Query only

17.10.4 History

CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]....1194
ACQuire:AVAilable?....1194
CHANnel<m>[:WAVeform<n>]:HISTory:CURRent....1195
CHANnel<m>[:WAVeform<n>]:HISTory:STARt....1195
CHANnel<m>[:WAVeform<n>]:HISTory:STOP....1195
CHANnel<m>[:WAVeform<n>]:HISTory:TPACq....1196
CHANnel<m>[:WAVeform<n>]:HISTory:PLAY....1196
CHANnel<m>[:WAVeform<n>]:HISTory:REPLay....1196
CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?....1197
CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?....1197
CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?....1197
CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?....1198

CHANnel<m>[:WAVeform<n>]:HISTory[:STATe] <State>

Enables or disables the history display.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

ACQuire:AVAilable?

Returns the number of acquisitions that is saved in the memory. This number of acquisitions is available for history viewing. It is also the number of acquisitions in a fast segmentation acquisition series.

Return values:

<AcquisitionCount> Range: 0 to 4294967295

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:CURRent <CurrAcqIdx>

Accesses a particular acquisition in the memory to display it. The query returns the index of the segment that is shown.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:

<CurrAcqIdx> History index: the newest segment has the index "0", older segments have a negative index: -(n-1), ... -1 , 0
where n is the number of acquired segments.
Range: 0 to -(n-1)
Increment: 1

Example:

```
CHAN2:HIST:STAT ON
CHAN2:HIST:CURR -1
*OPC
```

Displays the acquisition before last from the history.

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:STARt <StrtAcqIdx>

Sets the index of the oldest history acquisition for the history viewing.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:

<StrtAcqIdx> The start index is always negative.
Range: 0 to -(n-1)
Increment: 1

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:STOP <StpAcqIdx>

Sets the index of the latest segment to be displayed in the history viewer.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:

<StpAcqIdx> Index of the stop acquisition. The newest acquisition always has the index "0".

Range: 0 to -(n-1)
Increment: 1

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TPACq <TimePerAcq>

Sets the display time for one acquisition. The shorter the time, the faster is the replay.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:

<TimePerAcq> Range: 40E-6 to 10
Increment: 1
*RST: 0.05
Default unit: s

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:PLAY

Starts and stops the replay of the history waveforms.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Usage: Event
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:REPLay <AutoRepeat>

If ON, the replay of the history waveform sequence repeats automatically. Otherwise, the replay stops at the stop index set with `CHANnel<m>[:WAVeform<n>]:HISTory:STOP`.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Parameters:
<AutoRepeat> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?

Returns the date of the current acquisition that is shown in the history viewer (`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent`).

Suffix:
<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Return values:
<DateAbsString> String with date of the current acquisition (absolute time)

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?

Returns the abolsute daytime of the current acquisition that is shown in the history viewer (`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent`).

Suffix:
<m> 1..4
Selects the input channel.

<n> 1..3
[:WAVeform<n>] is irrelevant, omit it.

Return values:
<TimeAbsString> String containing the time and unit

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?

Returns the relative time of the current acquisition - the time difference to the newest acquisition (index = 0).

See also: (`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent`).

Suffix:
<m> 1..4
Selects the input channel.

<n>	1..3 [:WAVeform<n>] is irrelevant, omit it.

Return values:

<TimeRelativ>	Range: -100E+24 to 100E+24 Default unit: s

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?

Returns the relative time of the currently selected acquisition and the internal reference time (horizontal alignment) in history view with respect to the acquisition with index 0.

Suffix:

<m>	1..4 Selects the input channel.
<n>	1..3 [:WAVeform<n>] is irrelevant, omit it.

Return values:

<TimeRelIntRef>	Range: -100E+24 to 100E+24 Increment: 1 *RST: 0 Default unit: s

Usage: Query only
Asynchronous command

Firmware/Software: Version 2.70

17.10.5 XY-Diagram

WAVeform<m>:XYCurve:RATio....1198
WAVeform<m>:XYCurve:STATe....1199
WAVeform<m>:XYCurve:SWAP....1199
LAYout:SIGNal:AXIS....1199
WAVeform<m>:XYCurve:XSOurce....1200
WAVeform<m>:XYCurve:YSOurce....1200

WAVeform<m>:XYCurve:RATio <ConstantXYRatio>

If enabled, the x- and y-axes maintain a constant ratio in the diagram.

Suffix:

<m>	1..4 XY-diagram

Parameters:

<ConstantXYRatio>	ON | OFF *RST: ON

Usage: Asynchronous command

WAVeform<m>:XYCurve:STATe <State>

Activates an XY-waveform.

Suffix:

<m> 1..4
XY-diagram

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

WAVeform<m>:XYCurve:SWAP

Replaces the source of the x-axis with the source of the y-axis and vice versa.

Suffix:

<m> 1..4
XY-diagram

Usage: Event
Asynchronous command

LAYout:SIGNal:AXIS <DiagramName>, <Source>, <XSource>

Creates an XY-diagram by adding a second waveform to a diagram.

Setting parameters:

<DiagramName> String with the name of the diagram where the waveform is added.

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to be added, see Chapter 17.4.2, "Waveform Parameter", on page 1037

<XSource> ON | OFF

If on, the added waveform is assigned to the x-axis.
If off, it is assigned to the y-axis.

Usage: Setting only
Asynchronous command

WAVeform<m>:XYCurve:XSOurce <XSource>

Defines the signal source that supplies the x-values of the XY-diagram.

Suffix:

<m> 1..4
XY-diagram

Parameters:

<XSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | SG4TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of x-values, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

WAVeform<m>:XYCurve:YSOurce <YSource>

Defines the signal source that supplies the y-values of the XY-diagram.

Suffix:

<m> 1..4
XY-diagram

Parameters:

<YSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of y-values, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C2W1

Usage: Asynchronous command

17.11 Cursor Measurements

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

17.11.1 Cursor Setup....................1201
17.11.2 Cursor Results....................1208
17.11.3 Peak Search Using Cursors....................1210

17.11.1 Cursor Setup

CURSor<m>:STATe....................1201
CURSor<m>:AOFF....................1202
CURSor<m>:FUNCtion....................1202
CURSor<m>:TRACking[:STATe]....................1202
CURSor<m>:SOURce....................1203
CURSor<m>:SSOurce....................1203
CURSor<m>:USSource....................1204
CURSor<m>:X1Position....................1204
CURSor<m>:X2Position....................1204
CURSor<m>:XCOupling....................1205
CURSor<m>:Y1Position....................1205
CURSor<m>:Y2Position....................1205
CURSor<m>:YCOupling....................1206
CURSor<m>:X1ENvelope....................1206
CURSor<m>:X2ENvelope....................1207
CURSor<m>:STYLe....................1207
CURSor<m>:LABel....................1208
CURSor<m>:SIAD....................1208

CURSor<m>:STATe <State>

Switches the indicated cursor on or off.

Suffix:

<m>	1 | 2 Selects the cursor set.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

CURSor<m>:AOFF

This command switches all cursors off.

Suffix:

<m> The numeric suffix is irrelevant.

Usage: Event
Asynchronous command

CURSor<m>:FUNCtion <Type>

Defines the type of the indicated cursor set.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<Type> HORizontal | VERTical | PAIRed

HORizontal
A pair of horizontal cursor lines.

VERTical
A pair of vertical cursor lines.

PAIRed
Both vertical and horizontal cursor line pairs.

*RST: PAIRed

Usage: Asynchronous command

CURSor<m>:TRACking[:STATe] <TrackCurve>

If set to `ON`, the horizontal cursor lines follow the waveform.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<TrackCurve> ON | OFF

*RST: OFF

Usage: Asynchronous command

CURSor<m>:SOURce <Source>

Defines the source of the cursor measurement.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1 | SG2 | SG3 | SG4 | SG5 | SG6 | SG7 | SG8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of the cursor measurement, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

CURSor<m>:SSOurce <Source2>

Selects the second cursor source.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<Source2> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1 | SG2 | SG3 | SG4 | SG5 | SG6 | SG7 | SG8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Second source of the cursor measurement, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C2W1

Usage: Asynchronous command

Firmware/Software: FW 3.60

CURSor<m>:USSource <UseSource2>

Enables the second cursor source. To select the second source, use `CURSor<m>:SSOurce` on page 1203.

If enabled, the second corsor lines Cx.2 measure on the second source. Using a second source, you can measure differences between two channels with cursors.

Suffix:
<m> 1 | 2
Selects the cursor set.

Parameters:
<UseSource2> ON | OFF
*RST: OFF

Usage: Asynchronous command

Firmware/Software: FW 3.60

CURSor<m>:X1Position <XPosition1>

Defines the position of the left vertical cursor line.

Suffix:
<m> 1 | 2
Selects the cursor set.

Parameters:
<XPosition1> Range: 0 to 500
Increment: 0.1
*RST: 0
Default unit: s

Usage: Asynchronous command

CURSor<m>:X2Position <XPosition2>

Defines the position of the right vertical cursor line.

Suffix:
<m> 1 | 2
Selects the cursor set.

Parameters:
<XPosition2> Range: 0 to 500
Increment: 0.1
*RST: 0
Default unit: s

Usage: Asynchronous command

CURSor<m>:XCOupling <Coupling>

Defines the positioning mode of the vertical cursor.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<Coupling> ON | OFF

ON
Moving one cursor line moves the other cursor line too. The cursor lines always remain a fixed distance.

OFF
Each cursor line is positioned independently.

*RST: OFF

Usage: Asynchronous command

CURSor<m>:Y1Position <YPosition1>

Defines the position of the lower horizontal cursor line.

If `CURSor<m>:TRACking[:STATe]` is enabled, the query returns the measurement result - the lower vertical value of the waveform.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<YPosition1>
Range: -50 to 50
Increment: 0.01
*RST: 0
Default unit: The unit depends on the type of the waveform.

Usage: Asynchronous command

CURSor<m>:Y2Position <YPosition2>

Defines the position of the upper horizontal cursor line.

If `CURSor<m>:TRACking[:STATe]` is enabled, the query returns the measurement result - the upper vertical value of the waveform.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<YPosition2> Range: -50 to 50
Increment: 0.01
*RST: 0
Default unit: The unit depends on the type of the waveform.

Usage: Asynchronous command

CURSor<m>:YCOupling <Coupling>

Defines the positioning mode of the horizontal cursor. If the horizontal cursor lines track the waveform, the y-coupling is irrelevant (`CURSor<m>:MODE TRACk`).

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<Coupling> ON | OFF

ON
Moving one cursor line moves the other cursor line too. The cursor lines always remain a fixed distance.

OFF
Each cursor line is positioned independently.

*RST: OFF

Usage: Asynchronous command

CURSor<m>:X1ENvelope <EnvelopeCurve1>

If the waveform arithmetics are set to envelope curve (see `CHANnel<m>[:WAVeform<n>]:ARIThmetics` on page 1077) and `CURSor<m>:TRACking[:STATe]` is set to "ON", this setting defines how the first horizontal cursor is positioned.

Suffix:

<m> 1..*
Selects the cursor set. 2 cursor sets are available.

Parameters:

<EnvelopeCurve1> MIN | MAX

MIN
The horizontal cursor is set to the crossing point of the vertical cursor with the minimum waveform envelope.

MAX
The horizontal cursor is set to the crossing point of the vertical cursor with the maximum waveform envelope.

*RST: MIN

Usage: Asynchronous command

CURSor<m>:X2ENvelope <EnvelopeCurve2>

If the waveform arithmetics are set to envelope curve (see `CHANnel<m>[:WAVeform<n>]:ARIThmetics` on page 1077) and `CURSor<m>:TRACking[:STATe]` is set to "ON", this setting defines how the second horizontal cursor is positioned.

Suffix:

<m> 1..*
Selects the cursor set. 2 cursor sets are available.

Parameters:

<EnvelopeCurve2> MIN | MAX

MIN
The horizontal cursor is set to the crossing point of the vertical cursor with the minimum waveform envelope.

MAX
The horizontal cursor is set to the crossing point of the vertical cursor with the maximum waveform envelope.

*RST: MAX

Usage: Asynchronous command

CURSor<m>:STYLe <Style>

Defines how the cursor is displayed in the diagram.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<Style> LINes | LRHombus | VLRHombus | RHOMbus

LINes
The cursors are displayed as lines.

LRHombus
The cursors are displayed as lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

VLRHombus
The cursors are displayed only as vertical lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

RHOMbus
The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

*RST: LINes

Usage: Asynchronous command

CURSor<m>:LABel <ShowLabel>

Shows the cursor labels in the diagram.

Suffix:

<m> 1 | 2
Selects the cursor set.

Parameters:

<ShowLabel> ON | OFF
*RST: ON

Usage: Asynchronous command

CURSor<m>:SIAD <ShwInAllDiags>

Shows the enabled cursor measurements in all active diagrams of the same domain (time/ spectrum).

Suffix:

<m> 1..*
The suffix is irrelevant. All cursor measurements are affected.

Parameters:

<ShowInAllDiagr> ON | OFF
*RST: ON

Usage: Asynchronous command

17.11.2 Cursor Results

DISPlay:RESultboxes:CUPosition....1208
CURSor<m>:XDELta[:VALue]?....1209
CURSor<m>:XDELta:INVerse?....1209
CURSor<m>:YDELta[:VALue]?....1209
CURSor<m>:YDELta:SLOPe....1210

DISPlay:RESultboxes:CUPosition <Position>

Defines the position of the cursor measurement results on the screen.

Parameters:

<Position> PREV | FLOA | DOCK

PREV
Preview: result icon on the signal bar.

FLOA
Floating result box in front of the diagrams.

DOCK
Docked: fixed tab below the diagrams.

*RST: DOCK

Usage: Asynchronous command

CURSor<m>:XDELta[:VALue]?

Queries the delta value (distance) of two vertical cursor lines.

Suffix:

<m> 1 | 2
Selects the cursor set.

Return values:

<Delta> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only
Asynchronous command

CURSor<m>:XDELta:INVerse?

Queries the inverse value of the delta value (distance) of the two vertical cursor lines.

Suffix:

<m> 1 | 2
Selects the cursor set.

Return values:

<DeltaInverse> Range: -100E+24 to 100E+24
*RST: 0
Default unit: Hz

Usage: Query only
Asynchronous command

CURSor<m>:YDELta[:VALue]?

Queries the delta value (distance) of the two horizontal cursor lines.

Suffix:

<m> 1 | 2
Selects the cursor set.

Return values:

<Delta> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

CURSor<m>:YDELta:SLOPe <DeltaSlope>

Returns the inverse value of the voltage difference - the reciprocal of the vertical distance of two horizontal cursor lines: *1/ΔV*.

Suffix:

<m>	1 | 2 Selects the cursor set.	

Parameters:

<DeltaSlope>	Range:	-100E+24 to 100E+24
	Increment:	0
	*RST:	0

Usage: Asynchronous command

17.11.3 Peak Search Using Cursors

CURSor<m>:FFT:SETCenter....................1210
CURSor<m>:FFT:TOCenter....................1210
CURSor<m>:MAXimum[:PEAK]....................1210
CURSor<m>:MAXimum:LEFT....................1211
CURSor<m>:MAXimum:RIGHt....................1211
CURSor<m>:MAXimum:NEXT....................1211
CURSor<m>:THReshold....................1211
CURSor<m>:PEXCursion....................1212

CURSor<m>:FFT:SETCenter

Sets the center frequency to the frequency value that is measured at cursor line c1.

Suffix:
<m> The suffix is irrelevant.

Usage: Event
Asynchronous command

CURSor<m>:FFT:TOCenter

Sets the vertical cursor line c1 to the center frequency.

Suffix:
<m> The suffix is irrelevant.

Usage: Event
Asynchronous command

CURSor<m>:MAXimum[:PEAK]

Sets both cursors to the absolute peak value.

Suffix:
<m> 1 | 2
Selects the cursor set.

Usage: Event

CURSor<m>:MAXimum:LEFT

Sets cursor 2 to the next maximum to the left of the current position.

Suffix:
<m> 1 | 2
Selects the cursor set.

Usage: Event

CURSor<m>:MAXimum:RIGHt

Sets cursor 2 to the next peak to the right (from the current position).

Suffix:
<m> 1 | 2
Selects the cursor set.

Usage: Event

CURSor<m>:MAXimum:NEXT

Sets cursor 2 to the next smaller peak (from the current position).

Suffix:
<m> 1 | 2
Selects the cursor set.

Usage: Event

CURSor<m>:THReshold <Value>

Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

Suffix:
<m> The suffix is irrelevant.

Parameters:
<Value> Threshold in dBm

Usage: Asynchronous command

Firmware/Software: Version 2.70

CURSor<m>:PEXCursion <Value>

Defines the minimum level by which the waveform must rise or fall so that it will be identified as a maximum or a minimum by the search functions.

Suffix:
<m> The suffix is irrelevant

Parameters:
<Value> Range: 0 to 100
Increment: 1
*RST: 5
Default unit: dB

Usage: Asynchronous command

17.12 Automatic Measurements

This chapter contains all remote commands to set up automatic measurements and to analyze the measurement results.

Selection of the measurement: MEASurement<m>

With R&S RTE you can configure up to eight simultaneous measurements. In manual operation, these eight measurements are represented by subtabs "MG1" to "MG8" in the "Measurements" dialog box. For remote operation, the measurement is indicated by the suffix MEAS<m>, containing the number of the measurement.

Remote control: measurement suffix <m>	Manual operation: "MG" subtab
1 to 8 correspond to	"MG1" to "MG8"
9, 10: do not use	Not available

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- General Settings....1213
- Results....1217
- Amplitude/Time Measurements....1220
- Eye Diagram Measurements....1229
- Spectrum....1232
- Histograms....1240

- Display..1250
- Statistics and Long-term Measurements...1252
- Protocol..1258
- Track and Trend...1264
- Gating..1265
- Limit check and Event Actions..1268
- Reference Levels..1271

17.12.1 General Settings

MEASurement<m>[:ENABle]..1213
MEASurement<m>:SOURce...1213
MEASurement<m>:FSRC...1214
MEASurement<m>:SSRC...1215
MEASurement<m>:CATegory...1215
MEASurement<m>:MAIN...1215

MEASurement<m>[:ENABle] <State>

Switches the indicated measurement on or off.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<State> ON | OFF

*RST: OFF

Example: See "Simple Frequency and Amplitude Measurement" on page 1020.

Usage: Asynchronous command

MEASurement<m>:SOURce <SignalSource>, [<SignalSource2>]

Defines the source of the measurement. Availability of sources depends on the selected category and installed options.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

First source of the measurement, see Chapter 17.4.2, "Waveform Parameter", on page 1037

<SignalSource2> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Second measurement source. The 2nd source is required for amplitude/time measurements that are performed on two waveforms (e.g. delay, phase).
Digital channels are only available if <SignalSource> is a digital channel.
Step response, advanced jitter track components and synthetic eye sources are available for amplitude/time measurements only.

*RST: C1W1,C2W1

Example: See "Simple Frequency and Amplitude Measurement" on page 1020.

MEASurement<m>:FSRC <Source>

Defines the first measurement source.

The command is an alternative to `MEASurement<m>:SOURce`.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Source> Source of the measurement, see `MEASurement<m>:SOURce` on page 1213.

*RST: C1W1

Usage: Asynchronous command

MEASurement<m>:SSRC <Source2>

Defines the second measurement source.

The command is an alternative to `MEASurement<m>:SOURce`.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Source2> Second source of the measurement, see `MEASurement<m>:SOURce` on page 1213.
Digital channels are only available if a digital channel is set as first measurement source using `MEASurement<m>:FSRC`.

*RST: C2W1

Usage: Asynchronous command

MEASurement<m>:CATegory <Category>

Defines the measurement category.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Category> AMPTime | EYEJitter | SPECtrum | HISTogram | PROTocol

AMPTime
Amplitude and time measurements

EYEJitter
Eye diagram mesurements

SPECtrum
Spectrum measurements

HISTogram
Histogram measurements

PROTocol
Protocol measurements

*RST: AMPTime

Example: See "Creating and Reading Histograms" on page 1021

Usage: Asynchronous command

MEASurement<m>:MAIN <MeasType>

Defines the measurement type of the indicated measurement.

Suffix:

<m>
1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<MeasType>

Amplitude/time measurements
HIGH | LOW | AMPLitude | MAXimum | MINimum | PDELta | MEAN | RMS | STDDev | POVershoot | NOVershoot | AREA | RTIMe | FTIMe | PPULse | NPULse | PERiod | FREQuency | PDCYcle | NDCYcle | CYCarea | CYCMean | CYCRms | CYCStddev | PULCnt | DELay | PHASe | BWIDth | PSWitching | NSWitching | PULSetrain | EDGecount | SHT | SHR | DTOTrigger | PROBemeter | SLERising | SLEFalling
See Chapter 17.12.3, "Amplitude/Time Measurements", on page 1220.
*RST value for amplitude/time measurements: `AMPLitude`.

Eye diagram measurements
ERPercent | ERDB | EHEight | EWIDth | ETOP | EBASe | QFACtor | RMSNoise | SNRatio | DCDistortion | ERTime | EFTime | EBRate | EAMPlitude | PPJitter | STDJitter | RMSJitter
See Chapter 17.12.4, "Eye Diagram Measurements", on page 1229
*RST value for eye/jitter measurements: `ERPercent`.

Spectrum measurements
CPOWer | OBWidth | SBWidth | THD | THDPCT | THDA | THDU | THDR | PLISt | HAR | THDF
See Chapter 17.12.5, "Spectrum", on page 1232.
*RST value for spectrum measurements: `CPOWer`.

Histogram measurements
WCOunt | WSAMples | HSAMples | HPEak | PEAK | UPEakvalue | LPEakvalue | HMAXimum | HMINimum | MEDian | MAXMin | HMEan | HSTDdev | M1STddev | M2STddev | M3STddev | MKPositive | MKNegative
See Chapter 17.12.6.2, "Histogram Measurement", on page 1246.
*RST value for histogram measurements: `WCOunt`.

Protocol measurements
NONE | AUDio | F2F | F2A | FLDValue | GAP | MBITrate | SBITrate | BIDLe | FCNT | FEC | FER | CFER
Requires option R&S RTE-K35
See Chapter 17.12.9, "Protocol", on page 1258

Example:
See "Simple Frequency and Amplitude Measurement" on page 1020.
See "Creating and Reading Histograms" on page 1021

Usage:
Asynchronous command

17.12.2 Results

MEASurement<m>:ARES?......1217
MEASurement<m>:ARNames......1218
MEASurement<m>:RESult[:ACTual]?......1218
MEASurement<m>:RESult:AVG?......1218
MEASurement<m>:RESult:EVTCount?......1218
MEASurement<m>:RESult:NPEak?......1218
MEASurement<m>:RESult:PPEak?......1218
MEASurement<m>:RESult:RELiability?......1218
MEASurement<m>:RESult:RMS?......1218
MEASurement<m>:RESult:WFMCount?......1218
MEASurement<m>:RESult:STDDev?......1218
MEASurement<m>:RESult:STARt?......1219
MEASurement<m>:RESult:STOP?......1219
MEASurement<m>:RESult:COUNt?......1219

MEASurement<m>:ARES?

Returns the results of the selected measurement. If repeated measurements and/or statistics are enabled, the instrument returns all results.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Return values:

<sData> Result string

Example:

```
:MEASurement:ARES?
1.225296442688e-001
```

Example:

```
:MEASurement:ARNames 1
:MEASurement:ARES?
High: 1.225296442688e-001
```

Example:

```
:MEASurement:MULTiple 1
:MEASurement:ARNames 1
:MEASurement:ARES?
High:
1.225296442688e-001,1.225296442688e-001,
1.225296442688e-001,1.225296442688e-001,
1.225296442688e-001,3.785715534183e-015,3893,
3893
```

Example:
```
:MEASurement:ARNames 1
:MEASurement:STATistics:ENABle 1
:MEASurement:ARES?
High:
1.225296442688e-001,1.225296442688e-001,
1.225296442688e-001,1.225296442688e-001,
1.225296442688e-001,1.673196471220e-014,176656,
176656
Results: current, peak+, peak-, average, RMS,
standard deviation, event count, waveform count
```

Usage: Query only
Asynchronous command

MEASurement<m>:ARNames <Identifier>

Enables a prefix that indicates the measurement in the result string of the `MEASurement<m>:ARES?` command.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<Identifier> ON | OFF
*RST: OFF

Example:
```
:MEASurement:ARNames 1
:MEASurement:ARES?
Amplitude: 2.371541501976e-001
```

Usage: Asynchronous command

MEASurement<m>:RESult[:ACTual]? [<MeasType>]
MEASurement<m>:RESult:AVG? [<MeasType>]
MEASurement<m>:RESult:EVTCount? [<MeasType>]
MEASurement<m>:RESult:NPEak? [<MeasType>]
MEASurement<m>:RESult:PPEak? [<MeasType>]
MEASurement<m>:RESult:RELiability? [<MeasType>]
MEASurement<m>:RESult:RMS? [<MeasType>]
MEASurement<m>:RESult:WFMCount? [<MeasType>]
MEASurement<m>:RESult:STDDev? [<MeasType>]

Return the statistic results of the selected measurement.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results

- RELiability: reliability of the measurement result
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

For a detailed description of the results see "Selection of the measurement: MEASurement<m>" on page 1212.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Query parameters:
<MeasType> | Not relevant, omit it

Example: See "Simple Frequency and Amplitude Measurement" on page 1020.
See "Creating and Reading Histograms" on page 1021

Usage: Query only

MEASurement<m>:RESult:STARt? [<MeasType>]
MEASurement<m>:RESult:STOP? [<MeasType>]

Return the start and stop times of the selected measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Query parameters:
<MeasType> Not relevant, omit it

Usage: Query only

MEASurement<m>:RESult:COUNt? <MeasType>

Returns the number of result groups that are returned by `MEASurement:RESult:ACTual? HAR`.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Query parameters:
<MeasType> HAR
The command is only relevant for harmonic search.

Return values:
<Count> Number of result groups

Example:
```
:MEASurement:RESult:ACTual? HAR
99.9;-6.000139236;199.8;-80.701713562;299.7;-15.528377533;
:MEASurement:RESult:COUNt? HAR
4
```

Usage: Query only

Firmware/Software: Version 2.70

17.12.3 Amplitude/Time Measurements

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit check remote commands are listed.

For a detailed description, see Chapter 7.2.5.1, "Overview of Amplitude/Time Measurements", on page 310.

Table 17-5: Amplitude and time measurement types

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description, result	Suffix <n>
HIGH	High	High signal level	1
LOW	Low	Low signal level	2
AMPLitude	Amplitude	Amplitude of the signal	3
MAXimum	Max	Maximum value of the waveform	4
MINimum	Min	Minimum value of the waveform	5
PDELta	Peak to peak	Peak-to-peak value of the waveform	6
MEAN	Mean	Mean value of the waveform	7
RMS	RMS	RMS (Root Mean Square) value of the voltage	8
STDDev	σ (S-dev)	Standard deviation of the waveform	9
POVershoot	Pos. overshoot	Positive overshoot of a square wave	10
NOVershoot	Neg. overshoot	Negative overshoot of a square wave	11
AREA	Area	Area beneath the waveform (integral)	12
RTIMe	Rise time	Rise time of the left-most rising edge of the waveform.	13
FTIMe	Fall time	Falling time of the left-most falling edge of the waveform.	14
PPULse	Pos. pulse	Width of a positive pulse – a rising edge followed by a falling edge. The measurement requires at least one complete period of a triggered signal.	15
NPULse	Neg. pulse	Width of a negative pulse – a falling edge followed by a rising edge. The measurement requires at least one complete period of a triggered signal.	16
PERiod	Period	Length of the left-most signal period of the waveform	17

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description, result	Suffix <n>
FREQuency	Frequency	Frequency of the signal. The result is based on the period measurement.	18
PDCYcle	Pos. duty cycle	Positive duty cycle. The measurement requires at least one complete period of a triggered signal.	19
NDCYcle	Neg. duty cycle	Negative duty cycle. The measurement requires at least one complete period of a triggered signal.	20
CYCarea	Cycle area	Area (integral) beneath one cycle	21
CYCMean	Cycle mean	Mean value of one cycle	22
CYCRms	Cycle RMS	The RMS (Root Mean Square) value of one cycle	23
CYCStddev	Cycle σ (S-dev)	Standard deviation of one cycle	24
PULCnt	Pulse count	Number of positive or negative pulses of the waveform, or both	25
DELay	Delay	Time difference between the any edges of two measurement sources at any reference level. The measurement result is negative if the edge of the second source comes before the edge of the first source.	26
PHASe	Phase	Phase difference between two waveforms	27
BWIDth	Burst width	Duration of one burst, measured from the first edge to the last	28
PSWitching	Pos. switching	Settling time at rising edges	29
NSWitching	Neg. switching	Settling time at falling edges	30
PULSetrain	Pulse train	Duration of N positive pulses, measured from the rising edge of the first pulse to the falling edge of the N-th pulse. N has to be configured.	31
EDGecount	Edge count	Number of positive or negative edges of the waveform, or both	32
SETup	Setup time	Parameters to query the setup and hold times. Use these parameters only in following queries: • `MEASurement<m>:ARES?` • `MEASurement<n>:RESult:..` commands	33
HOLD	Hold time		
SHT	Setup/Hold time	Setting parameter to enable Setup/Hold time measurements. Use this parameter only as setting in: • `MEASurement<m>:MAIN` on page 1215	35
SHR	Setup/Hold ratio	Setup/Hold ratio measurement. Setup/Hold ratio is the ratio of the setup time to the sum of hold and setup time: $T_{Setup} / (T_{Setup} + T_{Hold})$ Use this parameter as setting to activate the Setup/Hold ratio measurement in: • `MEASurement<m>:MAIN` on page 1215 It is also used in the following queries: • `MEASurement<m>:ARES?` • `MEASurement<n>:RESult:..` Used also in queries with and commands.	36

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description, result	Suffix <n>
DTOTrigger	Delay to trigger	Time between the trigger event and a following signal slope. High accuracy even if the trigger event is outside the acquisition data.	46
PROBemeter	Trig. ProbeMeter	DC voltage measured by the connected active R&S probe	47
SLERising	Slew rate on rising edge	Steepness of the rising edge: voltage difference between the lower and higher reference level, divided by the rise time.	48
SLEFalling	Slew rate on falling edge	Steepness of the falling edge: voltage difference between the higher and lower reference level, divided by the fall time.	49

MEASurement<m>:ENVSelect....1222
MEASurement<m>:DETThreshold....1223
MEASurement<m>:AMPTime:ALEVel....1223
MEASurement<m>:AMPTime:PFSLope....1223
MEASurement<m>:AMPTime:PSLope....1224
MEASurement<m>:AMPTime:DELay<n>:DIRection....1224
MEASurement<m>:AMPTime:DELay<n>:ECOunt....1225
MEASurement<m>:AMPTime:DELay<n>:SLOPe....1225
MEASurement<m>:AMPTime:PTCount....1226
MEASurement<m>:AMPTime:ESLope....1226
MEASurement<m>:AMPTime:CSLope....1226
MEASurement<m>:AMPTime:CLCK<n>:LSELect....1227
MEASurement<m>:AMPTime:DATA<n>:LSELect....1227
MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe....1227
MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect....1228
MEASurement<m>:AMPTime:LCHeck<n>:VALid....1228
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit....1228
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit....1228
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin....1229
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin....1229

MEASurement<m>:ENVSelect <EnvelopeCurve>

The command is only relevant for measurements on envelope waveforms. It selects the envelope to be used for measurement.

Suffix:

<m> 1..10

See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<EnvelopeCurve> MIN | MAX | BOTH

MIN: measures on the lower envelope
MAX: measures on the upper envelope
BOTH: the envelope is ignored and the waveform measured as usual

*RST: BOTH

Usage: Asynchronous command

MEASurement<m>:DETThreshold <SignDetectThres>

Defines the value above which measurement results are displayed. Values beneath the threshold are considered to be noise and they are ignored.

Suffix:

<m> 1..10
irrelevant

Parameters:

<SignDetectThres> Range: 0 to 50
Increment: 1
*RST: 5
Default unit: %

Usage: Asynchronous command

MEASurement<m>:AMPTime:ALEVel <AreaLevel>

Defines the reference level used to integrate the waveform.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<AreaLevel> Range: -100E+24 to 100E+24
Increment: 0
*RST: 0
Default unit: V

Usage: Asynchronous command

MEASurement<m>:AMPTime:PFSLope <PeriodSlope>

Selects the slope direction for frequency and period measurements.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<PeriodSlope> FIRSt | POSitive | NEGative | EITHer

POSitive | NEGative
Measures the time between rising or falling edges, respectively.

EITHer
In multiple measurements, the time is measured both between rising edges and between falling edges.
In single measurements. the first edge is taken for the measurement.

FIRSt
Time is measured either between rising edges or between falling edges. The first edge is taken for the measurement. In single measurements, it works the same way as "Either".
Only available for analog measurement sources.

*RST: FIRSt (analog source), POSitive (digital source)

Usage: Asynchronous command

MEASurement<m>:AMPTime:PSLope <PulsesSlope>

Sets the first slope of the pulses to be counted. The setting is only relevant for pulse count measurement (`MEASurement<m>:MAIN PULCnt` or `MEASurement<m>:ADDitional PULCnt,ON`).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<PulsesSlope> POSitive | NEGative | EITHer

Count either positive or negative pulses, or both.

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:DIRection <EdgeCntDirct>

Selects the direction for counting slopes for each source: from the beginning of the waveform, or from the end.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
Selects the source number.

Parameters:

<EdgeCntDirct> FRFI | FRLA

FRFI - FRom FIrst, counting starts with the first edge of the waveform.
FRLA - FRom LAst, counting starts with the last edge of the waveform.

*RST: FRFI

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:ECOunt <EdgeIndex>

Sets the number of the edge that is relevant for delay measurement for each source.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
Selects the source number.

Parameters:

<EdgeIndex> Edge number

Range: 1 to 100000
Increment: 1
*RST: 1

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:SLOPe <Slope>

Sets the edge of each source, between which the delay is measured.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
Selects the source number.

Parameters:

<Slope> POSitive | NEGative | EITHer

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:PTCount <PulseCount>

Sets the number of positive pulses for the pulse train measurement. It measures the duration of N positive pulses from the rising edge of the first pulse to the falling edge of the N-th pulse.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<PulseCount> Range: 1 to 2147483647
Increment: 1
*RST: 1

Usage: Asynchronous command

MEASurement<m>:AMPTime:ESLope <EdgesSlope>

Sets the edge direction to be counted: rising edges, falling edges, or both. The setting is only relevant for edge count measurement (`MEASurement<m>:MAIN EDGecount` or `MEASurement<m>:ADDitional EDGecount,ON`).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<EdgesSlope> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:CSLope <SetHoldClkSlp>

Sets the edge of the clock from which the setup and hold times are measured.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<SetHoldClkSlp> POSitive | NEGative | EITHer

EITHer
The clock edges next to the data edge are considered regardless of the clock slope.
*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:CLCK<n>:LSELect <ClkLevSel>

Selects the reference level of the clock on which the time is measured. Reference level and clock slope define the time point for setup and hold measurements.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
The suffix is irrelevant, omit it.

Parameters:

<ClkLevSel> UPPer | MIDDle | LOWer
*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:DATA<n>:LSELect <DatLevSel>

Selects the reference level of the data on which the setup and hold time are measured.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
The suffix is irrelevant, omit it.

Parameters:

<DatLevSel> UPPer | MIDDle | LOWer
*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe <DlyTrigSlp>

Sets the edge direction to be used for delay measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
The suffix is irrelevant.

Parameters:

<DlyTrigSlp> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect <DlyTrigLevSel>

Selects the reference level of the measurement source on which the delay is measured.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..2
The suffix is irrelevant.

Parameters:

<DlyTrigLevSel> UPPer | MIDDle | LOWer
*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for amplitude vs. time measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..49
Number of the amplitude/time measurement, see Table 17-5.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limit for limit checking, respectiveley. The valid range is defined using the `MEASurement<m>:AMPTime:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..49
Number of the amplitude/time measurement, see Table 17-5.

Parameters:

<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower and upper margins for the limit check, respectively. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:AMPTime:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..49
Number of the amplitude/time measurement type, see Table 17-5.

Parameters:

<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

17.12.4 Eye Diagram Measurements

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit check remote commands are listed.

For a detailed description, see Chapter 7.2.6.1, "Overview of Eye Diagram Measurements", on page 322.

Table 17-6: Eye diagram measurement types

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
		not used	1
ERPercent	Extinction ratio (%)	Eye base / Eye top *100 Prerequisite: Eye base > 0 and Eye top > 0	2
ERDB	Extinction ratio (dB)	10*log (Eye top / Eye base)	3
EHEight	Eye height	Vertical eye opening	4
EWIDth	Eye width	Horizontal eye opening	5
ETOP	Eye top	Mean of the upper vertical histogram	6
EBASe	Eye base	Mean of the lower vertical histogram	7
		not used	8...9
QFACtor	Q factor	(Eye top – Eye base) / (σtop + σbase)	10
		not used	11...13
RMSNoise	Noise (RMS)	Quadratic mean of the noise of eye top and eye base	14
SNRatio	S/N ratio	Signal-to-noise ratio 10 * log (Eye amplitude / Noise RMS)	15
DCDistortion	Duty cycle distortion	20 * log (Eye amplitude / Noise RMS)	16
ERTime	Eye rise time	Duration for signal to rise from 10% to 90% of the high signal level	17
EFTime	Eye fall time	Duration for signal to fall from 90% to 10% of the high signal level	18
EBRate	Eye bit rate	Frequency between two crossings	19
EAMPlitude	Eye amplitude	Eye top - Eye base	20
		not used	21...27
PPJitter	Jitter (peak to peak)	Average of the jitter for both crossing points (σcrossing1 + σcrossing2) / 2	28
STDJitter	Jitter (6*σ)	Jitter *6	29
RMSJitter	Jitter (RMS)	Quadratic mean of the jitter at both crossing points	30

MEASurement<m>:EYEJitter:AUToscale....................1231
MEASurement<m>:EYEJitter:LCHeck<n>:VALid....................1231
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit....................1231
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit....................1231
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin....................1232
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin....................1232

MEASurement<m>:EYEJitter:AUToscale

Defines optimized settings to perform an eye diagram measurement for the selected source.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Event
Asynchronous command

MEASurement<m>:EYEJitter:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for eye/jitter measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..30
Number of the eye/jitter measurement, see Table 17-6.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limit for the limit check, respectively. The valid range is defined using the `MEASurement<m>:EYEJitter:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..30
Number of the eye/jitter measurement, see Table 17-6.

Parameters:

<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin <UpperMargin>

Defines the upper margin for the limit check. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:EYEJitter:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..30
Number of the eye/jitter measurement, see Table 17-6.

Parameters:

<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

17.12.5 Spectrum

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit check remote commands are listed.

For a detailed description, see Chapter 7.2.7.1, "Overview of Spectrum Measurements", on page 324.

Table 17-7: Spectrum measurements

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
CPOWer	Channel power	Power integrated over the sample values defined by a center frequency and a bandwidth; based on a defined impedance; the result is given in mW	1
		not used	2
OBWidth	Occupied bandwidth	From the defined center frequency, symmetric sample value pairs to the left and right are integrated until a user-defined percentage of the total power is reached	3

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
SBWidth	Bandwidth	n dB down bandwidth; the samples to the left and right of the peak value are analyzed until the n dB threshold is exceeded; the frequencies at which the threshold is exceeded define the limits of the requested bandwidth	4
		not used	5, 6
THD	THD[dB]	Total harmonic distortion in dB	7
THDPCT	THD[%]	Total harmonic distortion in %	8
THDA	THD_a	Is equivalent to THD. Requires option R&S RTE-K18.	9
THDU	THD_u	Requires option R&S RTE-K18.	10
THDR	THD_r	Distorsion factor. Requires option R&S RTE-K18.	11
PLISt	Peak list	List of frequency and peak power value pairs. Requires option R&S RTE-K18.	12
HAR	Harmonic search	Returns the measured harmonics. For each harmonic, the frequency and the value is listed. To get the number of result pairs (= harmonics), use MEASurement<m>:RESult:COUNt?.	13
THDF	THD_f	Root mean square of the sum of all amplitudes of the harmonic waves in relation to the amplitude at the fundamental frequency (first harmonic). Requires option R&S RTE-K18.	14

MEASurement<m>:SPECtrum:CPOWer:BANDwidth....................1233
MEASurement<m>:SPECtrum:OBANdwidth....................1234
MEASurement<m>:SPECtrum:CPOWer:CFRequency....................1234
MEASurement<m>:SPECtrum:NDBDown....................1234
MEASurement<m>:SPECtrum:PEXCursion....................1235
MEASurement<m>:SPECtrum:ATHReshold....................1235
MEASurement<m>:SPECtrum:RESult<n>:COUNt....................1235
MEASurement<m>:SPECtrum:RESult<n>:MODE....................1236
MEASurement<m>:RESult:MAXCount....................1236
MEASurement<m>:RESult:INVerse....................1237
MEASurement<m>:RESult:LABorder....................1237
MEASurement<m>:RESult:SHFRequency....................1238
MEASurement<m>:RESult:SHLabels....................1238
MEASurement<m>:SPECtrum:LCHeck<n>:VALid....................1239
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit....................1239
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit....................1239
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin....................1240
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin....................1240

MEASurement<m>:SPECtrum:CPOWer:BANDwidth <ChPowBw>

Defines the bandwidth over which the channel power is calculated.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>"
on page 1212.

Parameters:

<ChPowBw> Range: 0 to 4E+9
Increment: 1
*RST: 0
Default unit: Hz

Usage: Asynchronous command

MEASurement<m>:SPECtrum:OBANdwidth <OccupiedBW>

Defines the percentage of the total power used to determine the occupied bandwidth.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>"
on page 1212.

Parameters:

<OccupiedBW> Range: 0.1 to 99.9
Increment: 1
*RST: 20
Default unit: %

Usage: Asynchronous command

MEASurement<m>:SPECtrum:CPOWer:CFRequency <ChPowCtrFreq>

Defines the center frequency from which the channel power is calculated over the specified bandwidth.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>"
on page 1212.

Parameters:

<ChPowCtrFreq> Range: 0 to 6E+9
Increment: 1
*RST: 0
Default unit: Hz

Usage: Asynchronous command

MEASurement<m>:SPECtrum:NDBDown <NDbDown>

Defines the threshold until which the samples to the left and right of the peak value are analyzed in order to determine the "N dB down bandwidth".

Suffix:

<m> 1..10

See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<NDbDown> Range: 0 to 100

Increment: 1

*RST: 20

Default unit: dB

Usage: Asynchronous command

MEASurement<m>:SPECtrum:PEXCursion <Value>

Defines a relative threshold, the minimum level value by which the waveform must rise or fall to be considered as a peak. To avoid identifying noise peaks, enter a peak excursion value that is higher than the noise levels.

Suffix:

<m> 1..10

The suffix is irrelevant.

Parameters:

<Value>

Usage: Asynchronous command

Firmware/Software: Version 2.70

MEASurement<m>:SPECtrum:ATHReshold <Value>

Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

Suffix:

<m> 1..10

See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Value> numeric value

Default unit: dBm

Usage: Asynchronous command

MEASurement<m>:SPECtrum:RESult<n>:COUNt <MaxNoOfResults>

Sets the maximum number of measurement results that are listed in the result table. Available for peak list and harmonic search mesurements.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..14

Parameters:

<MaxNoOfResults> Range: 1 to 1000
Increment: 1
*RST: 10

Usage: Asynchronous command

MEASurement<m>:SPECtrum:RESult<n>:MODE <ResultMode>

Sets the the way the measurement results are displayed. Available only if `MEASurement:MAIN` is set to `PLISt` or `HAR`.

Suffix:

<m> 1..10

<n> 1..14
1...11, 14: not available
12: Peak list measurements
13: Harmonic search measurements

Parameters:

<ResultMode> ABS | REL

ABS
The harmonics/peaks are shown in absolute value, dBm.

REL
The level of the carrier is shown in absolute value dBm. The values the harmonics/peaks are shown relatively to the carrier in dBc.

*RST: ABS

Example: :MEASurement1:SPECtrum:RESult13:MODE ABS
The harmonic measurements are shown in absolute value, dBm.

Usage: Asynchronous command

MEASurement<m>:RESult:MAXCount <MeasType>,<Number>
MEASurement<m>:RESult:MAXCount? <MeasType>

Defines the maximum number of peaks that are labeled in the diagram.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Number> Maximum number of peaks

Parameters for setting and query:

<MeasType> PLISt

Only for peaklist measurements. For other measurement type, this command returns an error.

MEASurement<m>:RESult:INVerse <MeasType>, <State>
MEASurement<m>:RESult:INVerse? <MeasType>

Displays labels with black font on white background using the "Full frame" label type (if `MEASurement<m>:RESult:LABorder=ON`).).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<MeasType> PLISt

Only for peaklist measurements. For other measurement type, this command returns an error.

Example:

Display labels.
```
MEAS:RES:SHL PLIS,ON
```
Select inverted labels.
```
MEAS:RES:INV PLIS,ON
```
Query the type of labels for peak lists.
```
MEAS:RES:INV? PLIS
//Result: ON
```

MEASurement<m>:RESult:LABorder <MeasType>, <FrameType>
MEASurement<m>:RESult:LABorder? <MeasType>

Defines the layout of the labels.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<FrameType> NOBorder | UNDERLINE | FULL

NOBorder
Label without border or lines.

UNDERLINE
The label is underlined.

FULL
The label is surrounded by a frame.

Parameters for setting and query:

<MeasType> PLISt
Only for peaklist measurements. For other measurement types, this command returns an error.

MEASurement<m>:RESult:SHFRequency <MeasType>, <State>
MEASurement<m>:RESult:SHFRequency? <MeasType>

Includes the frequency of the detected peak in the diagram labels (if `MEASurement<m>:RESult:LABorder=ON`).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<MeasType> PLISt
Only for peaklist measurements. For other measurement types, this command returns an error.

Example:
Display labels.
`MEAS:RES:SHL PLIS,ON`
Show frequency in labels.
`MEAS:RES:SHFR PLIS,ON`
Query the type of labels for peak lists.
`MEAS:RES:SHFR? PLIS`
`//Result: ON`

MEASurement<m>:RESult:SHLabels <MeasType>, <State>
MEASurement<m>:RESult:SHLabels? <MeasType>

Defines whether a description (label) is displayed for each detected peak in the spectrum diagram.

The layout of the label is defined by `MEASurement<m>:RESult:LABorder`.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<MeasType> PLISt

Only for peaklist measurements. For other measurement types, this command returns an error.

Example: Display labels.
`MEAS:RES:SHL PLIS,ON`

MEASurement<m>:SPECtrum:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for spectrum measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..14
Number of the spectrum measurement type, see Table 17-7.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limits for the limit check, respectively. The valid range is defined using the `MEASurement<m>:SPECtrum:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..14
Number of the spectrum measurement type, see Table 17-7.

Parameters:

<UpperLimit>
Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower and upper margins for the limit check, respectively. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:SPECtrum:LCHeck<n>:VALid` command.

Suffix:

<m>
1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n>
1..14
Number of the spectrummeasurement type, see Table 17-7.

Parameters:

<UpperMargin>
Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

17.12.6 Histograms

See also: Chapter 17.16.6, "Waveform Histogram Export to File", on page 1369

● Histogram Display....................1240
● Histogram Measurement....................1246

17.12.6.1 Histogram Display

LAYout:HISTogram:ADD....................1241
LAYout:HISTogram:SOURce....................1241
LAYout:HISTogram:MODE....................1242
LAYout:HISTogram:HORZ:MODE....................1242
LAYout:HISTogram:HORZ:ABSolute:STARt....................1243
LAYout:HISTogram:HORZ:ABSolute:STOP....................1243
LAYout:HISTogram:HORZ:RELative:STARt....................1243
LAYout:HISTogram:HORZ:RELative:STOP....................1243
LAYout:HISTogram:VERTical:MODE....................1244
LAYout:HISTogram:VERTical:ABSolute:STARt....................1244
LAYout:HISTogram:VERTical:ABSolute:STOP....................1244
LAYout:HISTogram:VERTical:RELative:STARt....................1245

LAYout:HISTogram:VERTical:RELative:STOP....1245
LAYout:HISTogram:RESet....1245
LAYout:HISTogram:REMove....1245

LAYout:HISTogram:ADD <HistogramName>, <Source>, <XStart>, <XStop>, <YStart>, <YStop>, <Relative>, <Orientation>

Defines and displays a new histogram for the specified source.

Note: To define the mode of the histogram (vertical or horizontal), use the `LAYout:HISTogram:MODE` command.

For numerical values, the input of a unit is not supported for this command.

Setting parameters:

<HistogramName> String defining the histogram name which is used to refer to the histogram by other functions.

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Data source of the histogram, see Chapter 17.4.2, "Waveform Parameter", on page 1037

<XStart> Defines the start value of the x-value range.

<XStop> Defines the stop value of the x-value range.

<YStart> Defines the start value of the y-value range.

<YStop> Defines the stop value of the y-value range.

<Relative> ON | OFF

Defines whether relative or absolute values are used for the value range definition.

<Orientation> VERTical | HORizontal

Example: See "Creating and Reading Histograms" on page 1021

Usage: Setting only
Asynchronous command

LAYout:HISTogram:SOURce <HistogramName>,<Source>
LAYout:HISTogram:SOURce? <HistogramName>

Defines the waveform which is the source of the histogram.

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform source of the histogram, see Chapter 17.4.2, "Waveform Parameter", on page 1037.

Parameters for setting and query:

<HistogramName> String parameter

Usage: Asynchronous command

LAYout:HISTogram:MODE <HistogramName>,<Mode>
LAYout:HISTogram:MODE? <HistogramName>

Defines or queries the type of histogram.

Parameters:

<Mode> VERTical | HORizontal

VERTical
Amplitude histogram (horizontal bars across amplitude)

HORizontal
Time or frequency histogram (vertical bars over time/frequencies)

*RST: VERTical

Parameters for setting and query:

<HistogramName> The name of the histogram as defined using `LAYout:HISTogram:ADD` on page 1241.

Usage: Asynchronous command

LAYout:HISTogram:HORZ:MODE <HistogramName>,<Mode>
LAYout:HISTogram:HORZ:MODE? <HistogramName>

Defines or queries whether the value range limits are entered as absolute or relative values.

Parameters:

<Mode> ABS | REL

*RST: ABS

Parameters for setting and query:

<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:ABSolute:STARt <HistogramName>,<Start>
LAYout:HISTogram:HORZ:ABSolute:STARt? <HistogramName>

Defines the horizontal start value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:ABSolute:STOP <HistogramName>,<Stop>
LAYout:HISTogram:HORZ:ABSolute:STOP? <HistogramName>

Defines the horizontal stop value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:RELative:STARt <HistogramName>,<RelativeStart>
LAYout:HISTogram:HORZ:RELative:STARt? <HistogramName>

Defines the horizontal start value of the histogram.

Parameters:

<RelativeStart>	Range:	0 to 100
	Increment:	0.1
	*RST:	0
	Default unit:	%

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:RELative:STOP <HistogramName>,<RelativeStop>
LAYout:HISTogram:HORZ:RELative:STOP? <HistogramName>

Defines the horizontal stop value of the histogram.

Parameters:

<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:MODE <HistogramName>,<Mode>
LAYout:HISTogram:VERTical:MODE? <HistogramName>

Defines or queries whether the value range limits are entered as absolute or relative values.

Parameters:

<Mode> ABS | REL
*RST: ABS

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:ABSolute:STARt <HistogramName>,<Start>
LAYout:HISTogram:VERTical:ABSolute:STARt? <HistogramName>

Defines the vertical start value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:ABSolute:STOP <HistogramName>,<Stop>
LAYout:HISTogram:VERTical:ABSolute:STOP? <HistogramName>

Defines the vertical stop value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:RELative:STARt <HistogramName>,<RelativeStart>
LAYout:HISTogram:VERTical:RELative:STARt? <HistogramName>

Defines the vertical start value of the histogram.

Parameters:
<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:RELative:STOP <HistogramName>,<RelativeStop>
LAYout:HISTogram:VERTical:RELative:STOP? <HistogramName>

Defines the vertical stop value of the histogram.

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:RESet <HistogramName>

Resets the values to begin a new histogram.

Setting parameters:
<HistogramName>

Usage: Setting only
Asynchronous command

LAYout:HISTogram:REMove <Name>

Removes the specified histogram.

Setting parameters:
<Name>

Usage: Setting only
Asynchronous command

17.12.6.2 Histogram Measurement

This chapter lists commands to set up measurements on histograms.

See also: "Creating and Reading Histograms" on page 1021.

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit check remote commands are listed.

For a detailed description, see Table 7-7.

Table 17-8: Histogram measurement types

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
WCOunt	Waveform count	Number of acquisitions (waveforms) the histogram is based on	1
WSAMples	Waveform samples	Number of samples from the most recent acquisition included in the current histogram	2
HSAMples	Histogram samples	Number of samples from all acquisitions included in the current histogram	3
HPEak	Histogram peak	Maximum sample value in the histogram	4
PEAK	Peak value	Signal value at the histogram peak	5
UPEakvalue	Upper peak value	Signal value at the maximum sample value in the upper half of the histogram	6
LPEakvalue	Lower peak value	Signal value at the maximum sample value in the lower half of the histogram	7
HMAXimum	Maximum	Highest signal value with a probability > 0	8
HMINimum	Minimum	Lowest signal value with a probability > 0	9
MEDian	Median	Signal value for which half the samples lie above, the other half below in the histogram	10
MAXMin	Max - Min	Range of signal values with a probability > 0	11
HMEan	Mean	Weighted arithmetic average of the histogram	12
HSTDdev	σ (S-dev)	Standard deviation of the sample numbers	13
M1STddev	Mean ±σ	Range between (mean value + standard deviation) and (mean value - standard deviation)	14
M2STddev	Mean ±2*σ	Range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)	15
M3STddev	Mean ±3*σ	Range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)	16

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
MKPositive	Marker + Probability %	Marker value (according to the selected probability domain marker type) plus the defined limit. Note that the value is restricted to the histogram range.	17
MKNegative	Marker - Probability %	Marker value (according to the selected probability domain marker type) minus the defined limit. Note that the value is restricted to the histogram range.	18

MEASurement<m>:HISTogram:SELect.. 1247
MEASurement<m>:HISTogram:PROBability:TYPE.. 1247
MEASurement<m>:HISTogram:PROBability:LIMit.. 1248
MEASurement<m>:HISTogram:LCHeck<n>:VALid.. 1248
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit...1249
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit... 1249
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin... 1249
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin... 1249

MEASurement<m>:HISTogram:SELect <HistgRef>

Selects the histogram on which the measurement is based.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<HistgRef> String with the name of the histogram

Example: See "Creating and Reading Histograms" on page 1021

Usage: Asynchronous command

MEASurement<m>:HISTogram:PROBability:TYPE <HistgPrbDomMark>

Defines the marker reference in the probability domain.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<HistgPrbDomMark> PEAK | UPPK | LWPK | MAXimum | MINimum | MEDian | MEAN

PEAK
The y-value with the maximum sample value in the histogram

UPPK
The y-value at the maximum sample value in the upper half of the histogram

LWPK
The y-value at the maximum sample value in the lower half of the histogram

MAXimum
The highest y-value with a probability > 0

MINimum
The lowest y-value with a probability > 0

MEDian
The y-value for which half the samples lie above, the other half below in the histogram

MEAN
The weighted arithmetic average of the histogram

*RST: PEAK

Usage: Asynchronous command

MEASurement<m>:HISTogram:PROBability:LIMit <HistgPrbDomLim>

Defines a range around the probability marker.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<HistgPrbDomLim> Range: 0 to 100
Increment: 10
*RST: 10
Default unit: %

Usage: Asynchronous command

MEASurement<m>:HISTogram:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for histogram measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..18
Number of the histogram measurement type, see Table 17-8.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limits for the limit check, respectively. The valid range is defined using the `MEASurement<m>:HISTogram:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..18
Number of the histogram measurement type, see Table 17-8.

Parameters:

<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower and upper margins for the limit check, respectively. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:HISTogram:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

<n> 1..18
Number of the histogram measurement type, see Table 17-8.

Parameters:

<UpperMargin>	Range:	-100 to 100
	Increment:	10E-6
	*RST:	0

Usage: Asynchronous command

17.12.7 Display

MEASurement<m>:DISPlay:LEVels....1250
MEASurement<m>:DISPlay:RESults....1250
MEASurement<m>:DISPlay:STYLe....1251
MEASurement<m>:DISPlay:HISTogram....1251
DISPlay:RESultboxes:MEPosition....1251

MEASurement<m>:DISPlay:LEVels <DisplayLevels>

If enabled, the reference levels used for the measurement are displayed in the diagram.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<DisplayLevels> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:DISPlay:RESults <DisplayResult>

If enabled, the intermediate result linesare displayed in the measurement diagram.

Result lines mark the samples in the waveform that are used to obtain the measurement result. These are, for example, maximum and minimum values, mean, cycle start and cycle end, and others.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<DisplayResult> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:DISPlay:STYLe <DisplayStyle>

Selects the style in which the measurement waveform is displayed.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<DisplayStyle> LINE | MARKer

LINE
The individual data points are connected by a line.

MARKer
Only the individual data points are displayed as markers.

*RST: LINE

Usage: Asynchronous command

MEASurement<m>:DISPlay:HISTogram <DispHistg>

Displays a histogram for the source of the selected measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<DispHistg> ON | OFF

*RST: OFF

Usage: Asynchronous command

DISPlay:RESultboxes:MEPosition <Position>

Defines the position of the measurement results on the display.

Parameters:
<Position> PREV | FLOA | DOCK

PREV
Preview: result icon on the signal bar

FLOA
Floating: floating result box in front of the diagrams

DOCK
Docked: fixed tab below the diagrams

*RST: DOCK

Usage: Asynchronous command

17.12.8 Statistics and Long-term Measurements

See also: Chapter 17.16.9, "Long Term Measurement Results and Measurement Histogram Export to File", on page 1372.

MEASurement<m>:STATistics[:ENABle]....1252
MEASurement<m>:CLEar....1252
MEASurement<m>:MULTiple....1253
MEASurement<m>:MNOMeas....1253
MEASurement<m>:STATistics:HISTogram....1253
MEASurement<m>:STATistics:HBINs....1254
MEASurement<m>:STATistics:MODE....1254
MEASurement<m>:STATistics:RCOunt....1254
MEASurement<m>:STATistics:RMEascount....1255
MEASurement<m>:STATistics:RTIMe....1255
MEASurement<m>:STATistics:RESet....1255
MEASurement<m>:VERTical:CONT....1256
MEASurement<m>:VERTical:AUTO....1256
MEASurement<m>:VERTical:OFFSet....1256
MEASurement<m>:VERTical:SCALe....1257
MEASurement<m>:LTMeas[:STATe]....1257
MEASurement<m>:LTMeas:COUNt....1257
MEASurement<m>:LTMeas:TIME....1258

MEASurement<m>:STATistics[:ENABle] <StatisticsState>

Enables statistics calculation for the measurement.

For details on the statistics results, see Chapter 7.2.10.1, "Statistics", on page 343.

Suffix:

<m> — The suffix is irrelevant, the command enables statistics for all measurements.

Parameters:

<StatisticsState> — ON | OFF

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:CLEar

Deletes the statistic results of the indicated mesurement.

Suffix:

<m> — 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Event
Asynchronous command

MEASurement<m>:MULTiple <MultiMeas>

The measurement is performed repeatedly if the measured parameter occurs several times inside the acquisition or defined gate.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<MultiMeas> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:MNOMeas <MaxMeasPerAcq>

Sets the maximum number of measurements per acquisition if repeated measurement is enabled (`MEASurement<m>:MULTiple` is `ON`).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<MaxMeasPerAcq> Range: 2 to 1000000
Increment: 1
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:HISTogram <ShowHistogram>

Displays a histogram of the statistical results. Enabling the histogram enables also the calculation and display of statistics for the measurement results if statistics were disabled. the histogram shows the cumulative occurence distribution of mean measurement results in a graphic.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ShowHistogram> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:STATistics:HBINs <HistogramBins>

Sets the number of bins - the number of vertical bars that build the histogram.

If `MEASurement<m>:VERTical:CONT` is ON, the instrument determines the number of bins automatically based on the time base, the current measurements, and other settings.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<HistogramBins> Range: 2 to 1000
Increment: 10
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:MODE <ResetMode>

Defines when the statistics for long term measurements are reset.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ResetMode> TIME | WFMS | MEAS

TIME
Sets one long term measurement point after the time defined using `MEASurement<m>:STATistics:RTIMe`.

WFMS - Waveforms
Sets one long term measurement point after a number of acquired waveforms defined using `MEASurement<m>:STATistics:RCOunt`.

MEAS
Sets one long term measurement point after a number of measurement results.

*RST: TIME

Usage: Asynchronous command

MEASurement<m>:STATistics:RCOunt <RstWfmCnt>

Defines the number of measured waveforms from which one point of the long term measurement is created (reset of statistics).

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<RstWfmCnt>	Range: 1 to 65535 Increment: 10 *RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:RMEascount <RstMeasCnt>

Defines the number of measurement results from which one point of the long term measurement is created.

This setting is only available if `MEASurement<m>:STATistics:MODE` is set to `MEAS`.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<RstMeasCnt>	Range: 1 to 65535 Increment: 10 *RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:RTIMe <ResetTime>

Defines the time or period after which the statistics are reset.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ResetTime>	Range: 0.1 to 2.14748E+9 Increment: 1E-3 *RST: 0.2 Default unit: s

Usage: Asynchronous command

MEASurement<m>:STATistics:RESet

Resets the histogram, the long term measurement and the statistics.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Event
Asynchronous command

MEASurement<m>:VERTical:CONT <ContAutoScl>

If enabled, automatic vertical scaling is performed whenever the waveform does not fit in the diagram during the long term measurement period.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ContAutoScl> ON | OFF
*RST: ON

Usage: Asynchronous command

MEASurement<m>:VERTical:AUTO

Performs an automatic scaling once so that the scaling is adapted to the current measurement results. Available only for long term measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Event
Asynchronous command

MEASurement<m>:VERTical:OFFSet <VerticalOffset>

Defines a vertical offset for the long term measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<VerticalOffset> Range: -100E+12 to 100E+12
Increment: 1E-6
*RST: 0
Default unit: div

Usage: Asynchronous command

MEASurement<m>:VERTical:SCALe <VerticalScale>

Defines the vertical scaling per division, so that the scaling can be adapted automatically during the long term measurement period.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<VerticalScale>
Range: 1E-12 to 100E+12
Increment: 10E-6
*RST: 0.5
Default unit: V/div

Usage: Asynchronous command

MEASurement<m>:LTMeas[:STATe] <ShowDiagram>

Enables long term measurement for a defined number of measurement points (see `MEASurement<m>:LTMeas:COUNt` on page 1257) or a specified time (see`MEASurement<m>:LTMeas:TIME` on page 1258).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ShowDiagram> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:LTMeas:COUNt <MeasCnt>

Defines the total number of points to be measured during the long term measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<MeasCnt>
Range: 1000 to 200000
Increment: 10
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:LTMeas:TIME <MeasurementTime>

Defines the total duration of the long term measurement.

This setting is only available if `MEASurement<m>:STATistics:MODE` is set to "Time".

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<MeasurementTime> Range: 0.01 to 2.14748E+9
Increment: 1
*RST: 200
Default unit: s

Usage: Asynchronous command

17.12.9 Protocol

Before starting the protocol measurement a bus type has to be selected (`BUS<m>:TYPE`) and the bus has to be activated (`BUS<m>[:STATe]`).

The following table lists the <MeasType> parameter values with a short description.

Table 17-9: Protocol measurement types

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result
AUDio	Audio	
FLDValue	Field value	Value of a field over time
F2F	Frame to frame	The distance between two frame types
	Trigger to frame	
GAP	Gap	Measures a gap, periods at which the bus is idle. The distance of a gap can only be measured between two identified frames.
MBITrate	Main bit rate	Bit rate of the protocol
SBITrate	2nd bit rate	Additional bit rate, protocol dependent.
BIDLe	Bus idle	Calculates the bus idle time
FCNT	Frame count	Number of all frames within the acquisition window
FEC	Frame error count	Sum of all frames with errors within the acquisition window

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result
FER	Frame error rate	Sum of all frames with errors divided by all frames within the acquisition window
CFER	Consecutive frame error rate	Measures the frame error rate

MEASurement<m>:PROTocol:F2FRame:FLDFrom... 1259
MEASurement<m>:PROTocol:F2FRame:FLDTo.. 1259
MEASurement<m>:PROTocol:F2FRame:FRMFrom.. 1260
MEASurement<m>:PROTocol:F2FRame:FRMTo...1260
MEASurement<m>:PROTocol:F2FRame:VALFrom...1260
MEASurement<m>:PROTocol:F2FRame:VALTo.. 1260
MEASurement<m>:PROTocol:FLDValue:FLD.. 1261
MEASurement<m>:PROTocol:FLDValue:FRM... 1261
MEASurement<m>:PROTocol:FLDValue:TRCK.. 1261
MEASurement<m>:PROTocol:FLDValue:VAL.. 1261
MEASurement<m>:PROTocol:MBITrate:FLD.. 1262
MEASurement<m>:PROTocol:MBITrate:FRM... 1262
MEASurement<m>:PROTocol:MBITrate:VAL.. 1262
MEASurement<m>:PROTocol:SBITrate:FLD...1262
MEASurement<m>:PROTocol:SBITrate:FRM..1262
MEASurement<m>:PROTocol:SBITrate:VAL...1263
MEASurement<m>:PROTocol:T2FRame:DIRection..1263
MEASurement<m>:PROTocol:T2FRame:FLD...1263
MEASurement<m>:PROTocol:T2FRame:FRM.. 1263
MEASurement<m>:PROTocol:T2FRame:VALue...1264

MEASurement<m>:PROTocol:F2FRame:FLDFrom <FieldName>

Sets the type of field for the start frame of the frame to frame protocol measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<FieldName>

MEASurement<m>:PROTocol:F2FRame:FLDTo <FieldName>

Sets the type of field for the end frame of the frame to frame protocol measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:F2FRame:FRMFrom <FrameName>

Sets the type of start frame for the frame to frame protocol measurement.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:F2FRame:FRMTo <FrameName>

Sets the type of end frame for the frame to frame protocol measurement.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:F2FRame:VALFrom <FrmFrmFromVal>

Sets the field value of the start frame for the frame to frame protocol measurement.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FrmFrmFromVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:F2FRame:VALTo <FrmFrmVal>

Sets the field value of the end frame for the frame to frame protocol measurement.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FrmFrmVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:FLDValue:FLD <FieldName>

Sets the type of field for the frame of the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:FLDValue:FRM <FrameName>

Sets the type frame for the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:FLDValue:TRCK <FieldName>

Sets the field to be tracked value of the frame for the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:FLDValue:VAL <FieldValIdVal>

Sets the field value of the frame for the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FieldValIdVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:MBITrate:FLD <FieldName>

Sets the type of field for the frame of the main bit rate protocol measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<FieldName>

MEASurement<m>:PROTocol:MBITrate:FRM <FrameName>

Sets the type frame for the main bit rate protocol measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<FrameName>

MEASurement<m>:PROTocol:MBITrate:VAL <MainBtRateIdVal>

Sets the field value of the frame for the main bit rate protocol measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<MainBtRateIdVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:SBITrate:FLD <FieldName>

Sets the type of field for the frame of the secondary bit rate protocol measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<FieldName>

MEASurement<m>:PROTocol:SBITrate:FRM <FrameName>

Sets the type frame for the secondary bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:SBITrate:VAL <SecondaryBtRateIdVal>

Sets the field value of the frame for the secondary bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<SecondaryBtRateIdVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:T2FRame:DIRection <TrigFrmDirct>

Sets the direction of the field for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<TrigFrmDirct> T2F | F2T

`T2F`: trigger to frame
`F2T`: frame to trigger

*RST: T2F

Usage: Asynchronous command

MEASurement<m>:PROTocol:T2FRame:FLD <FieldName>

Sets the type of the field for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<FieldName>

MEASurement<m>:PROTocol:T2FRame:FRM <FrameName>

Sets the value of the frame for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<FrameName>

MEASurement<m>:PROTocol:T2FRame:VALue <TrigFrmVal>

Sets the value of the field for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<TrigFrmVal>

Usage: Asynchronous command

17.12.10 Track and Trend

MEASurement<m>:TRACk[:STATe]... 1264
MEASurement<m>:TRACk:DATA:HEADer?...1264
MEASurement<m>:TRACk:DATA:STYPe?.. 1265
MEASurement<m>:TRACk:DATA[:VALues]?...1265

MEASurement<m>:TRACk[:STATe] <State>

Enables the track functionality and displays the track.

The track functionality requires at least one option, see "Enable (Track)" on page 350.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:TRACk:DATA:HEADer?

Returns the header of the track.

Suffix:
<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Query only

MEASurement<m>:TRACk:DATA:STYPe?

Returns the data type: TRK (track).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Query only

MEASurement<m>:TRACk:DATA[:VALues]?

Returns the data of track points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Query only

17.12.11 Gating

MEASurement<m>:GATE[:STATe].......... 1265
MEASurement<m>:GATE:MODE.......... 1266
MEASurement<m>:GATE:ABSolute:STARt.......... 1266
MEASurement<m>:GATE:ABSolute:STOP.......... 1266
MEASurement<m>:GATE:RELative:STARt.......... 1266
MEASurement<m>:GATE:RELative:STOP.......... 1266
MEASurement<m>:GATE:CURSor.......... 1267
MEASurement<m>:GATE:CCOupling.......... 1267
MEASurement<m>:GATE:ZCOupling.......... 1267
MEASurement<m>:GATE:ZDIagram.......... 1268
MEASurement<m>:GATE:GCOupling.......... 1268

MEASurement<m>:GATE[:STATe] <State>

Considers the gating settings of the source waveform for the measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:GATE:MODE <Mode>

Defines whether the gate settings are configured using absolute or relative values.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Mode> ABS | REL

*RST: ABS

Usage: Asynchronous command

MEASurement<m>:GATE:ABSolute:STARt <Start>
MEASurement<m>:GATE:ABSolute:STOP <Stop>

Define the absolute start and end values for the gate.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Usage: Asynchronous command

MEASurement<m>:GATE:RELative:STARt <RelativeStart>
MEASurement<m>:GATE:RELative:STOP <RelativeStop>

Define the relative start and end values for the gate, respectively.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Usage: Asynchronous command

MEASurement<m>:GATE:CURSor <Cursorset>

Selects the cursor set to be used for measurement gating. The gate area is defined by the cursor lines.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Cursorset>	CURSOR1 | CURSor1 | CURSOR2 | CURSor2 | CURSOR3 | CURSor3 | CURSOR4 | CURSor4 CURSOR1 = CURSor1, CURSOR2 = CURSor2, CURSOR3 = CURSor3, CURSOR4 = CURSor4 *RST: CURSOR1

Usage: Asynchronous command

MEASurement<m>:GATE:CCOupling <CursorCoupling>

Enables the cursor coupling for automatic measurements.

Select the cursor set to be used with `MEASurement<m>:GATE:CURSor`.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<CursorCoupling>	ON | OFF *RST: OFF

Usage: Asynchronous command

MEASurement<m>:GATE:ZCOupling <ZoomCoupling>

If enabled, the gate area is defined identically to the zoom area for the zoom diagram.

If enabled, define the zoom area to be used as gate with `MEASurement<m>:GATE:ZDIagram`.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ZoomCoupling>	ON | OFF *RST: OFF

Usage: Asynchronous command

MEASurement<m>:GATE:ZDIagram <DiagramName>, <ZoomName>

If `MEASurement<m>:GATE:ZCOupling` is enabled, the gate area is defined identically to the zoom area for the selected zoom diagram.

Suffix:

<m> 1..10

See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<ZoomDiagram> String with the name of the diagram on which the zoom is based

Example: Prerequisite: Zoom 1 is available

```
MEASurement:GATE:ZCOupling ON
MEASurement:GATE:ZDIagram "Diagram1","Zoom1"
```

Usage: Asynchronous command

MEASurement<m>:GATE:GCOupling <GateCoupling>

If you enable the gate coupling, the gate settings of the selected measurement are copied to all other measurements. If zoom or cursor coupling is active in a measurement, the zoom size and cursor positions are adjusted.

Suffix:

<m> 1..10

See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<GateCoupling> ON | OFF

*RST: OFF

Usage: Asynchronous command

17.12.12 Limit check and Event Actions

MEASurement<m>:LCHeck....................1268
MEASurement<m>:ONViolation:BEEP....................1269
MEASurement<m>:ONViolation:ACQStop....................1269
MEASurement<m>:ONViolation:PRINt....................1270
MEASurement<m>:ONViolation:WFMSave....................1270
MEASurement<m>:ONViolation:REPort....................1270
MEASurement<m>:ONViolation:TRIGgerout....................1271
MEASurement<m>:ONViolation:RUNexec....................1271

MEASurement<m>:LCHeck <LimitCheckState>

Defines the type of the limit check that can run together with the measurement.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<LimitCheckState> OFF | LIMit | LMARgin

OFF
No limit check.

LIMit
Only limits are checked.

LMARgin
Limits and margins are checked.

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:ONViolation:BEEP <Beep>

Generates a beep sound for the specified event.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Beep> NOAction | SUCCess | VIOLation

See Chapter 17.4.5, "Event Parameter", on page 1038

*RST: NOAction

Usage: Asynchronous command

MEASurement<m>:ONViolation:ACQStop <StopAcq>

Stops data acquisition for the specified event.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<StopAcq> NOAction | SUCCess | VIOLation

See Chapter 17.4.5, "Event Parameter", on page 1038

*RST: NOAction

Usage: Asynchronous command

MEASurement<m>:ONViolation:PRINt <Print>

Prints a screenshot including the measurement results to the printer defined using `SYSTem:COMMunicate:PRINter:SELect<1..2>` for the specified event.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Print>	NOAction | SUCCess | VIOLation See Chapter 17.4.5, "Event Parameter", on page 1038 *RST: NOAction

Usage: Asynchronous command

MEASurement<m>:ONViolation:WFMSave <SaveWfm>

Saves the waveform data.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<SaveWfm>	NOAction | SUCCess | VIOLation See Chapter 17.4.5, "Event Parameter", on page 1038 *RST: NOAction

Usage: Asynchronous command

MEASurement<m>:ONViolation:REPort <Report>

Creates and saves a report of the current settings and results.

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<Report>	NOAction | SUCCess | VIOLation See Chapter 17.4.5, "Event Parameter", on page 1038 *RST: NOAction

Usage: Asynchronous command

MEASurement<m>:ONViolation:TRIGgerout <TriggerOutPulse>

Creates a trigger out pulse on limit or margin violation or on successful completion of themeasurement.

Suffix:

<m>	1..10

Parameters:

<TriggerOutPulse>	NOACtion | SUCCess | VIOLation *RST: NOACtion

Usage: Asynchronous command

MEASurement<m>:ONViolation:RUNexec <RunExecutable>

Starts an external application if the command is set to `VIOLation` or `SUCCess`.

Use the following commands to set up the application:

- `EXECutable:NAME` on page 1069
- `EXECutable:PARameter` on page 1069
- `EXECutable:WDIRectory` on page 1069

Suffix:

<m>	1..10 See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<RunExecutable>	NOACtion | SUCCess | VIOLation See Chapter 17.4.5, "Event Parameter", on page 1038 *RST: NOACtion

Usage: Asynchronous command

17.12.13 Reference Levels

- General Reference Level Settings....1271
- Automatic Configuration....1273
- Manual Configuration....1274
- Results....1277

17.12.13.1 General Reference Level Settings

REFLevel<m>:LDETection....1272
REFLevel<m>:LMODe....1272
REFLevel<m>:RELative:MODE....1272

REFLevel<m>:LDETection <Mode>

Defines whether the reference level is configured manually or automatically.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Mode> AUTO | MANual

Example: `REFLevel2:LDETection MANual`
Sets manual level configuration for Ch1. C1W1 corresponds to suffix number 2.

Example: See: Chapter 17.3.2.2, "Setting Reference Levels", on page 1020

Usage: Asynchronous command

REFLevel<m>:LMODe <Mode>

Defines whether the reference is configured using absolute or relative values.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Mode> ABS | REL

Example: `REFLevel2:LMODe ABS`
Sets definition of reference levels to absolute values for Ch1. C1W1 corresponds to suffix number 2.

Example: See: "Manual Reference Level Definition Using Relative Values" on page 1020

Usage: Asynchronous command

REFLevel<m>:RELative:MODE <Mode>

The lower, middle and upper reference levels, defined as percentages of the high signal level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Mode> FIVE | TEN | TWENty | USER

FIVE
5/50/95

TEN
10/50/90

TWENty
20/50/80

USER
Set the reference levels to individual values with `REFLevel<m>:RELative:LOWer`, `REFLevel<m>:RELative:MIDDle`, and `REFLevel<m>:RELative:UPPer`.

Example: `REFL2:REL:MODE FIVE`
Reference levels for Ch1: Lower reference level = 5% of high signal level, middle reference level = 50% of high signal level, upper reference level = 95% of high signal level

Example: See: "Manual Reference Level Definition Using Relative Values" on page 1020

Usage: Asynchronous command

17.12.13.2 Automatic Configuration

REFLevel<m>:AUTO[:STATe]....................1273
REFLevel<m>:AUTO:COUNt....................1273

REFLevel<m>:AUTO[:STATe] <Value>

Enables averaging over several histograms to determine the reference levels. The number of histograms to consider is defined using `REFLevel<m>:AUTO:COUNt`.

This function is only available in automatic reference level mode (see `REFLevel<m>:LDETection` on page 1272).

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value> ON | OFF

Usage: Asynchronous command

REFLevel<m>:AUTO:COUNt <Value>

Defines the number of histograms from which the average is calculated.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value>

Usage: Asynchronous command

17.12.13.3 Manual Configuration

REFLevel<m>:ABSolute:HIGH....1274
REFLevel<m>:ABSolute:LOW....1274
REFLevel<m>:ABSolute:TDIStance....1275
REFLevel<m>:ABSolute:BDIStance....1275
REFLevel<m>:ABSolute:MLEVel....1275
REFLevel<m>:RELative:UPPer....1276
REFLevel<m>:RELative:MIDDle....1276
REFLevel<m>:RELative:LOWer....1277

REFLevel<m>:ABSolute:HIGH <Value>

The signal value that represents a high level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:HIGH 0.015`
Sets the high signal level for Ch1 to 15 mV. C1W1 corresponds to suffix number 2.

Usage: Asynchronous command

REFLevel<m>:ABSolute:LOW <Value>

The signal value that represents a low level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:Low 0.0015`
Sets the low signal level forCh1 to 1.5 mV. C1W1 corresponds to suffix number 2.

Usage: Asynchronous command

REFLevel<m>:ABSolute:TDIStance <Value>

The distance between the high signal level and the upper reference level.

Suffix:

<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:

<Value>

Example: `REFLevel2:ABSolute:TDIStance 0.0002`
Sets the top distance for Ch1 to 0.2 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual Reference Level Definition Using Absolute Values" on page 1021

Usage: Asynchronous command

REFLevel<m>:ABSolute:BDIStance <Value>

The distance between the lower reference level and the low signal value.

Suffix:

<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:

<Value>

Example: `REFLevel2:ABSolute:BDIStance 0.0002`
Sets the bottom distance for Ch1 to 0.2 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual Reference Level Definition Using Absolute Values" on page 1021

Usage: Asynchronous command

REFLevel<m>:ABSolute:MLEVel <Value>

The middle level between high and low signal level. The value is adjusted automatically if you change the high or low signal levels. Vice versa, if you change the middle level, the high and low signal levels are adjusted.

Suffix:

<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:

<Value>

Example: `REFLevel2:ABSolute:MLEVel 0.05`
Sets the middle signal level for Ch1 to 50 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual Reference Level Definition Using Absolute Values" on page 1021

Usage: Asynchronous command

REFLevel<m>:RELative:UPPer <Value>

Sets the upper relative reference level if `REFLevel<m>:RELative:MODE` is set to `USER`.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value> Percentage of the high signal level.

Example: `REFLevel8:RELative:LOWer 85`
Sets the upper reference level for Ch3 to 85 %. Ch3 corresponds to suffix number 8.

Example: See: "Manual Reference Level Definition Using Relative Values" on page 1020

Usage: Asynchronous command

REFLevel<m>:RELative:MIDDle <Value>

Sets the middle relative reference level if `REFLevel<m>:RELative:MODE` is set to `USER`.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:
<Value> Percentage of the high signal level.

Example: `REFLevel8:RELative:MIDDle 50`
Sets the middle reference level for Ch3 to 50 %. Ch3 corresponds to suffix number 8.

Example: See: "Manual Reference Level Definition Using Relative Values" on page 1020

Usage: Asynchronous command

REFLevel<m>:RELative:LOWer <Value>

Sets the lower relative reference level if `REFLevel<m>:RELative:MODE` is set to `USER`.

Suffix:

<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 111, 115 to 118
Source waveform of the measurement, see Chapter 17.4.1, "Waveform Suffix", on page 1035.

Parameters:

<Value> Percentage of the high signal level.

Example: `REFLevel8:RELative:LOWer 15`
Sets the lower reference level for Ch3 to 15 %. Ch3 corresponds to suffix number 8.

Example: See: "Manual Reference Level Definition Using Relative Values" on page 1020

Usage: Asynchronous command

17.12.13.4 Results

MEASurement<m>:REFLevel:RESult:LOWer?....................1277
MEASurement<m>:REFLevel:RESult:MIDDle?....................1277
MEASurement<m>:REFLevel:RESult:UPPer?....................1277
MEASurement<m>:REFLevel:RESult:SIGLow?....................1277
MEASurement<m>:REFLevel:RESult:SIGHigh?....................1277

MEASurement<m>:REFLevel:RESult:LOWer?
MEASurement<m>:REFLevel:RESult:MIDDle?
MEASurement<m>:REFLevel:RESult:UPPer?

Return the lower, middle, and upper reference level, respectively.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Return values:

<Upper> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MEASurement<m>:REFLevel:RESult:SIGLow?
MEASurement<m>:REFLevel:RESult:SIGHigh?

Return the signal value that represents a low or high level, respectively.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Return values:

<SignalHigh> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

17.13 Spectrum Analysis

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

17.13.1 Basic FFT

CALCulate:MATH<m>:FFT:LOGScale.....1279
CALCulate:MATH<m>:FFT:STARt.....1279
CALCulate:MATH<m>:FFT:STOP.....1280
CALCulate:MATH<m>:FFT:CFRequency.....1280
CALCulate:MATH<m>:FFT:FULLspan.....1280
CALCulate:MATH<m>:FFT:SPAN.....1280
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted?.....1281
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO.....1281
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio.....1281
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue].....1282
CALCulate:MATH<m>:FFT:WINDow:TYPE.....1282
CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics.....1283
CALCulate:MATH<m>:FFT:FRAMe:COVerage?.....1284
CALCulate:MATH<m>:FFT:FRAMe:MAXCount.....1284
CALCulate:MATH<m>:FFT:FRAMe:OFACtor.....1284
CALCulate:MATH<m>:FFT:GATE:COUPling.....1285
TIMebase:RACTime?.....1285
CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt.....1286
CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP.....1286
CALCulate:MATH<m>:FFT:GATE:MODE.....1286

CALCulate:MATH<m>:FFT:GATE:RELative:STARt..1286
CALCulate:MATH<m>:FFT:GATE:RELative:STOP..1287
CALCulate:MATH<m>:FFT:GATE:ZCOupling..1287
CALCulate:MATH<m>:FFT:GATE[:STATe]..1287
CALCulate:MATH<m>:FFT:MAGNitude:LEVel..1288
CALCulate:MATH<m>:FFT:MAGNitude:RANGe..1288
CALCulate:MATH<m>:FFT:MAGNitude:SCALe...1288
CALCulate:MATH<m>:FFT:PHASe:SCALe...1289
CALCulate:MATH<m>:FFT:PHASe:SUPPression..1289
CALCulate:MATH<m>:FFT:PHASe:THReshold..1290
CALCulate:MATH<m>:FFT:PHASe:UNWRap..1290
CALCulate:MATH<m>:FFT:COUPled:WITH<1..8>...1290

CALCulate:MATH<m>:FFT:LOGScale <XAxisMode>

Defines the scaling method for the frequency (x-)axis of the spectrogram.

This command is only available if option R&S RTE-K18 is installed.

Suffix:

<m> 1..8

Parameters:

<XAxisMode> LIN | LOG

LOG
Logarithmic scaling

LIN
Linear scaling

*RST: LIN

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:STARt <StartFreq>

Defines the start frequency of the displayed frequency span.

Suffix:

<m> 1..8
math waveform

Parameters:

<StartFreq> start frequency

Range: 0 to 5E+9
Increment: 1
*RST: 2E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:STOP <StopFreq>

Defines the stop frequency of the displayed frequency span.

Suffix:

<m> 1..8
math waveform

Parameters:

<StopFreq> stop frequency

Range: 0 to 5E+9
Increment: 1
*RST: 2E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:CFRequency <CenterFreq>

Defines the position of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The width of the range is defined using the `CALCulate:MATH<m>:FFT:SPAN` command.

Suffix:

<m> 1..8
math waveform

Parameters:

<CenterFreq> center frequency

Range: 0 to 2E+12
Increment: 1
*RST: 2.5E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:FULLspan

Performs FFT calculation for the full frequency span.

Suffix:

<m> 1..8
math waveform

Usage: Event
Asynchronous command

CALCulate:MATH<m>:FFT:SPAN <FreqSpan>

The span is specified in Hertz and defines the width of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The position of the span is defined using the `CALCulate:MATH<m>:FFT:CFRequency` command.

Suffix:

<m> 1..8
Math waveform

Parameters:

<FreqSpan> Frequency span

Range: 1 to 4E+12
Increment: 1
*RST: 5E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted?

Queries the effective resolution bandwidth.

Suffix:

<m> 1..8
Math waveform

Return values:

<AdjResBW> effective resolution bandwidth

Range: 0.01 to 2E+12
*RST: 0
Default unit: Hz

Usage: Query only

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO <State>

Couples the frequency span to the RBW.

Suffix:

<m> 1..8
math waveform

Parameters:

<State> ON | OFF

*RST: ON

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio <SpanRBWRatio>

This command defines the ratio of span (Hz) / resolution bandwidth (Hz).

Suffix:

<m> 1..8
math waveform

Parameters:

<SpanRBWRatio> ratio span / resolution bandwidth

Range: 1 to 1000
Increment: 1
*RST: 100

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue] <ResolutionBW>

This command defines the resolution bandwidth.

Suffix:

<m> 1..8
math waveform

Parameters:

<ResolutionBW> resolution bandwidth

Range: 0.01 to 160E+6
Increment: 0.01
*RST: 2E+6
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:WINDow:TYPE <WindowType>

Windowing helps minimize the discontinuities at the end of the measured signal interval and thus reduces the effect of spectral leakage, increasing the frequency resolution.

Various different window functions are provided in the R&S RTE to suit different input signals. Each of the window functions has specific characteristics, including some advantages and some trade-offs. These characteristics need to be considered carefully to find the optimum solution for the measurement task.

Suffix:

<m> 1..8
math waveform

Parameters:

<WindowType> RECTangular | HAMMing | HANN | BLACkharris | GAUSsian | FLATTOP2 | FLATtop2 | KAISerbessel

RECTangular
The rectangular window has the best frequency resolution, but a poor amplitude accuracy and is recommended for separating two tones with almost equal amplitudes and a small frequency distance.

HAMMing
The Hamming window is bell shaped and has a good frequency resolution and fair amplitude accuracy. It is recommended for frequency response measurements as well as sine waves, periodic signals and narrow-band noise

HANN
The Hann window is bell shaped and has a slightly worse frequency resolution but smaller sidelobe level than the Hamming window. The applications are the same.

BLACkharris
The Blackman window is bell shaped and has a poor frequency resolution, but very good amplitude accuracy. It is recommended mainly for signals with single frequencies to detect harmonics.

GAUSsian
Good frequiency resolution and best magnitude resolution, recommended for weak signals and short duration

FLATTOP2 = FLATtop2
The flattop window has a poor frequency resolution, but the best amplitude accuracy and the sharpest side lobe. It is recommended for accurate single-tone amplitude measurements.

KAISerbessel
The Kaiser-Bessel window has a fair frequency resolution and good amplitude accuracy, and a very high sidelobe level. It is recommended for separating two tones with differing amplitudes and a small frequency distance.

*RST: BLACkharris

Usage: Asynchronous command

Firmware/Software: Version 3.35 and higher: Use FLATTOP2 or FLATtop2 instead of FLAT2

CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics <Arithmetics>

The arithmetic mode defines how the final FFT result is calculated from the individual frame results.

Suffix:
<m> 1..8
Selects the math waveform.

Parameters:
<Arithmetics> OFF | ENVelope | AVERage | RMS | MINHold | MAXHold
See "FFT Segment Arithmetics" on page 371
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:FRAMe:COVerage?

Due to the restriction of the number of frames (see `CALCulate:MATH<m>:FFT:FRAMe:MAXCount` on page 1284), the waveform may only be analyzed partially. This command queries the percentage of the trace that was analyzed, i.e. which part of the trace was included in the frame calculation.

Suffix:

<m>	1..8 math waveform

Return values:

<FrameCoverage>	Range: 0 to 100 *RST: 100 Default unit: %

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:FFT:FRAMe:MAXCount <MaxFrameCount>

Restricts the maximum number of frames to be calculated. Due to the other parameter settings, the required number of frames may become very high, thus slowing performance. By restricting the number of frames, you can avoid performance loss without changing the other parameters.

Suffix:

<m>	1..8 math waveform

Parameters:

<MaxFrameCount>	Range: 1 to 10000 Increment: 10 *RST: 1000

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:FRAMe:OFACtor <OverlapFactor>

Defines the minimum factor by which two neighboring frames overlap. If the required number of frames to cover the input values allows for more overlap, the factor is increased.

The higher the overlap factor, the more frames are used. This leads to more individual results and improves detection of transient signal effects. However, it also extends the duration of the calculation.

Suffix:

<m>	1..8 math waveform

Parameters:

<OverlapFactor> Range: 0 to 90
Increment: 1
*RST: 50
Default unit: %

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:COUPling <GateRBWCoupling>

Defines the behaviour of the record length or RBW value in dependancy to the other FFT parameters.

See also:

- "Record Length/RBW Coupling" on page 374
- Chapter 8.1.1, "Fundamentals of FFT Analysis", on page 359

Suffix:

<m> 1..8
math waveform

Parameters:

<GateRBWCoupling> LENGth | RBW

LENGth
The record length remains constant. If not enough samples are available for the selected RBW, the RBW will be decreased.

RBW
The RBW is not adapted, i.e. remains as defined by the user. The required acquisition time for this RBW is indicated. If necessary and possible, the record length is extended to acquire the required number of samples.

*RST: RBW

Usage: Asynchronous command

TIMebase:RACTime?

Queries the required acquisition time. If FFT gating is used and the resolution BW is set to constant, record length can be extended to acquire the required number of samples. In this case, the required acquisition time differs from the adjusted acquisition time (`TIMebase:RANGe`).

Return values:

<RqrdAcqTime> Required acquisition time for FFT

Range: 125E-12 to 100E+3
*RST: 0.5
Default unit: s

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt <Start>

Defines the starting value for the gate.

Suffix:

<m> 1..8
math waveform

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP <Stop>

Defines the end value for the gate.

Suffix:

<m> 1..8
math waveform

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:MODE <Mode>

Defines whether the gate settings are configured using absolute or relative values.

Suffix:

<m> 1..8
math waveform

Parameters:

<Mode> ABS | REL
*RST: ABS

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:RELative:STARt <RelativeStart>

Defines the starting value for the gate in percent.

Suffix:

<m> 1..8
math waveform

Parameters:

<RelativeStart>	Range:	0 to 100
	Increment:	0.1
	*RST:	0
	Default unit:	%

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:RELative:STOP <RelativeStop>

Defines the end value for the gate in percent.

Suffix:

<m> 1..8
math waveform

Parameters:

<RelativeStop>	Range:	0 to 100
	Increment:	0.1
	*RST:	100
	Default unit:	%

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:ZCOupling <ZoomCoupling>

If enabled, the gate area is defined identically to the zoom area for the zoom diagram.

Suffix:

<m> 1..8
math waveform

Parameters:

<ZoomCoupling> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE[:STATe] <State>

Enables FFT gating.

Suffix:

<m> 1..8
math waveform

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:MAGNitude:LEVel <VerticalMax>

Defines the reference level for dB scaling.

Suffix:

<m> 1..8
Math waveform

Parameters:

<VerticalMax> Range: -1E+15 to 1E+15
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:MAGNitude:RANGe <Range>

Defines the vertical value range in spectrum mode.

Suffix:

<m> 1..8
math waveform

Parameters:

<Range> Range: 1 to 500
Increment: 1
*RST: 100
Default unit: dB

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:MAGNitude:SCALe <MagnitudeScale>

Defines the scaling of the y-axis. The display values are valid for 50Ω termination impendance.

For details on the available scaling modes, see "Magnitude unit" on page 375.

Suffix:

<m> 1..8
math waveform

Parameters:

<MagnitudeScale> LINear | DBM | DB | DBUV | DBMV | DBV | DBPS | DBNS | DBUS | DBMS | DBS | DBHZ | DBKHZ | DBKHz | DBMHZ | DBMHz | DBGHZ | DBGHz | DBA | DBMA | DBUA

LINear
Linear scaling; displays the RMS value of the voltage

*RST: DBM

Usage: Asynchronous command

Table 17-10: Logarithmic scaling values

DBM	dBm
DB	dB (related to reference level)
DBUV	dBµV
DBMV	dBmV
DBV	dBV
DBPS	dBps
DBNS	dBns
DBUS	dBµs
DBMS	dBms
DBS	dBs
DBHZ	dBHz
DBKHZ = DBKHz	dBkHz
DBMHZ = DBMHz	dBMHz
DBGHZ = DBGHz	dBGHz
DBA	dBA
DBMA	dBmA
DBUA	dBµA

CALCulate:MATH<m>:FFT:PHASe:SCALe <PhaseScale>

Defines the scaling unit for phase display.

Suffix:
<m> 1..8
math waveform

Parameters:
<PhaseScale> DEGRees | RADians
*RST: DEGRees

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:PHASe:SUPPression <Suppression>

Enables noise suppression. Phase calculation is restricted to frequencies with a minimum magnitude, the threshold value (see `CALCulate:MATH<m>:FFT:PHASe:THReshold` on page 1290).

Suffix:
<m> 1..8
math waveform

Parameters:

<Suppression> ON | OFF

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:PHASe:THReshold <SupprThres>

Defines the minimum frequency magnitude for which phases are calculated. This setting is only available if `CALCulate:MATH<m>:FFT:PHASe:SUPPression` is set to "ON".

Suffix:

<m> 1..8
math waveform

Parameters:

<SupprThres> Range: -180 to 180
Increment: 0.1
*RST: 0
Default unit: dBm

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:PHASe:UNWRap <Unwrap>

If enabled, phase shifts due to a limitation of the value range are eliminated.

Suffix:

<m> 1..8
math waveform

Parameters:

<Unwrap> ON | OFF

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:COUPled:WITH<1..8> <MathIndex>

Copies the current FFT settings of the selected math waveform (m) to the other selected math waveform (m2), and couples the two waveforms. This can be repeated for all math waveforms.

If any FFT setting for any of the coupled spectrums is changed, it is changed for all coupled spectrums.

Suffix:

<m> 1..8
FFT waveforms to be coupled. <m>, <m2> must be active and distinct.

Parameters:

<MathIndex> ON | OFF

*RST: OFF

Example: Prerequisite: FFt waveforms M1 and M2 are active.

`CALC:MATH1:FFT:COUP:WITH2 ON`

Couples the math waveforms M1 and M2.

Usage: Asynchronous command

17.13.2 Waveform Data

CALCulate:MATH<m>:DATA:STYPe?... 1291
CALCulate:MATH<m>:DATA:HEADer?..1291
CALCulate:MATH<m>:DATA[:VALues]?...1292

CALCulate:MATH<m>:DATA:STYPe?

Returns the signal type of the source of the math waveform.

Suffix:

<m> 1..8

Selects the math waveform.

Return values:

<SignalType> SOUR | SPEC | CORR | MEAS | NONE

SOURce = normal signal
SPECtrum = FFT spectrum, specific math signal
CORRelation = correlated signal, specific math signal
MEAsurement = result of a measurement
NONE = undefined

Usage: Query only

Asynchronous command

CALCulate:MATH<m>:DATA:HEADer?

Returns the header of math waveform data. The header contains attributes of the waveform.

Table 17-11: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of the waveform in Samples	200000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of samples (record length).	1

Suffix:

<m> 1..8
Selects the math waveform.

Example: `CALC:MATH1:DATA:HEAD?`
`-9.477E-008,9.477E-008,200000,1`

Usage: Query only

CALCulate:MATH<m>:DATA[:VALues]?

Returns the data of the math waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values, or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define this.

Suffix:

<m> 1..8
Selects the math waveform.

Return values:

<Data> List of values according to the format and content settings.

Usage: Query only

17.13.3 Spectrum Analysis (Option R&S RTE-K18)

In all `CALC:MATH<m>:FFT` commands, the suffix <m> selects the math waveform.

In all `CALC:MATH<m>:FFT:SPEC:TIM` commands, the suffix <m> selects the timeline.

CALCulate:MATH<m>:FFT:SPECtrogram:CMODe.. 1292
CALCulate:MATH<m>:FFT:USEColtab..1293
CALCulate:MATH<m>:FFT:SPECtrogram:STATe...1293
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition.................................... 1293
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe..1294

CALCulate:MATH<m>:FFT:SPECtrogram:CMODe <ColorTableMode>

Selects the color table mode for the frequency analysis display.

Suffix:

<m> 1..8

Parameters:

<ColorTableMode> INCI | AMPL

INCI
("Incidence") The display color is set depending on the frequency of occurrence of a value.

AMPL
("Amplitude") In the spectrogram and the frequency analysis display, the color is used to indicate the magnitude of the FFT signal. The higher the amplitude the higher the color in the assigned color table.

*RST: INCI

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:USEColtab <UseColorTable>

If enabled, the selected waveform is displayed according to its assigned color table. For information on the available color tables, see Chapter 3.4.2.2, "Color Tables", on page 108.

If this option is disabled, the preset color of the selected channel source is displayed, and the intensity of the specific signal color varies according to the cumulative occurrence of the values.

Suffix:
<m> 1..8

Parameters:
<UseColorTable> ON | OFF

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:SPECtrogram:STATe <State>

Enables the spectrogram display for a math waveform.

Suffix:
<m> 1..8

Parameters:
<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition <Position>

Defines the position of one of the two possible time lines in a spectrogram. The time line must be enabled first, using the `CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe` command.

Suffix:
<m> 1..8

<n> 1..2

Parameters:

<Position> The position of the time line is defined by the index of the data acquisition in the history. How many acquisitions are available depends on the history settings.

Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe <State>

Enables one of two possible time lines in a spectrogram diagram. A time line marks a single waveform in the spectrogram, that is: the power vs frequency results for the data acquired at a specific time. After enabling a time line, the results for that time are displayed in the spectrum diagram.

The position of the time line is defined using the `CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition` command.

Suffix:

<m> 1..8

<n> 1..2

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

17.14 Mask Testing

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- Mask Test Definition....1295
- Mask Definition: User Mask....1298
- Mask Definition: Waveform Mask....1304
- Event Actions....1306
- Mask Display....1309
- Results....1312

17.14.1 Mask Test Definition

MTESt:ADD....1295
MTESt:REMove....1295
MTESt[:STATe]....1295
MTESt:RST....1296
MTESt:SOURce....1296
MTESt:CONDition....1296
MTESt:TOLerance....1297
MTESt:CTYPe....1297
MTESt:FILE:NAME....1298
MTESt:FILE:SAVE....1298
MTESt:FILE:OPEN....1298
MTESt:FILE:DELete....1298

MTESt:ADD <MaskTestName>

Creates a new mask test definition with the specified name.

Setting parameters:

<MaskTestName> String with the name of the mask test

Example: See Chapter 17.3.3.1, "Creating a User Mask", on page 1023

Usage: Setting only
Asynchronous command

MTESt:REMove <MaskTestName>

Deletes the mask test definition with the specified name.

Setting parameters:

<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt[:STATe] <MaskTestName>,<State>
MTESt[:STATe]? <MaskTestName>

Activates and deactivates the mask test. If the acquisition is running, the test starts immediately. Otherwise, the test starts when acquisition is started.

The testing is stopped when acquisition is stopped, also due to the `MTESt:ONViolation:STOP` command, or if `MASK[:STATe]` is set to "OFF".

The coommand needs `*OPC` command synchronisation, see Chapter B.3, "Command Sequence and Synchronization", on page 1980.

Parameters:

<State> ON | OFF

*RST: OFF

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Example:
```
MTEST:STAT 'MyMask', ON; *OPC?
```
See Chapter 17.3.3.1, "Creating a User Mask", on page 1023

Usage: Asynchronous command

MTESt:RST

Clears all totals and results in all "Mask Test" result boxes.

Usage: Event
Asynchronous command

MTESt:SOURce <MaskTestName>,<Source>
MTESt:SOURce? <MaskTestName>

Selects the waveform to be tested against the mask.

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to be tested, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:CONDition <MaskTestName>,<PassFailMode>
MTESt:CONDition? <MaskTestName>

Sets the first criteria for a failed test, the kind of hits to be considered for test evaluation. A test has failed if the number of sample hits or acquisition hits exceeds the limit defined by `MTESt:TOLerance`.

Parameters:

<PassFailMode> SAMPles | ACQuisitions

SAMPles
Considers the number of samples that hit the mask.

ACQuisitions
Considers the number of acquisitions that contain at least one sample hit. How many samples hit the mask in that acquisition is not relevant.

*RST: SAMPles

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:TOLerance <MaskTestName>,<TolViolCnt>
MTESt:TOLerance? <MaskTestName>

Sets the second criteria for a failed test, the number of tolerable sample hits or acquisition hits. Use `MTESt:CONDition` to define which hits are considered for test evaluation.

Parameters:

<TolViolCnt> Range: 0 to 4000000000
Increment: 1
*RST: 0

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:CTYPe <MaskTestName>,<DefinitionType>
MTESt:CTYPe? <MaskTestName>

Sets the method of mask definition.

Parameters:

<DefinitionType> USER | WFML | EYEMask | PROTocol

USER
The mask segments are created by entering the numerical x- and y-values of the mask points.
See: Chapter 17.14.2, "Mask Definition: User Mask", on page 1298

WFML
The mask is created from the envelope of an existing waveform.
See: Chapter 17.14.3, "Mask Definition: Waveform Mask", on page 1304

*RST: USER

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:FILE:NAME <MaskTestName>, <Path>
MTESt:FILE:NAME? <MaskTestName>

Specifies a file to save the mask test.

Parameters:

<Path>	String containing path and file name, format .xml

Parameters for setting and query:

<MaskTestName>	String with the name of the mask test

MTESt:FILE:SAVE <MaskTestName>

Saves the specified mask test. It contains the mask definition, defined actions and fail conditions.

Setting parameters:

<MaskTestName>	String with the name of the mask test
Usage:	Setting only Asynchronous command

MTESt:FILE:OPEN <MaskTestName>

Loads the specified mask test to the instrument.

Setting parameters:

<MaskTestName>	String with the name of the mask test
Usage:	Setting only Asynchronous command

MTESt:FILE:DELete <MaskTestName>

Deletes the specified mask test.

Setting parameters:

<MaskTestName>	String with the name of the mask test
Usage:	Setting only Asynchronous command

17.14.2 Mask Definition: User Mask

The chapter contains commands required for the definition of user masks - `MTESt:CTYPe` is set to `USER`.

Segment and point indices

In remote control, the numbering of segments and points starts from 0. But in manual operation, the numbering starts from 1.

MTESt:SEGMent:STATe....................1299
MTESt:SEGMent:ADD....................1299
MTESt:SEGMent:COUNt?....................1300
MTESt:SEGMent:INSert....................1300
MTESt:SEGMent:REMove....................1300
MTESt:SEGMent:CLEar....................1300
MTESt:SEGMent:REGion....................1300
MTESt:SEGMent:POINt:ADD....................1301
MTESt:SEGMent:POINt:INSert....................1301
MTESt:SEGMent:POINt:REMove....................1301
MTESt:SEGMent:POINt:COUNt?....................1302
MTESt:SEGMent:POINt:X....................1302
MTESt:SEGMent:POINt:Y....................1302
MTESt:SEGMent:RESCale:RECalculate....................1303
MTESt:SEGMent:RESCale:XFACtor....................1303
MTESt:SEGMent:RESCale:YFACtor....................1303
MTESt:SEGMent:RESCale:XOFFset....................1303
MTESt:SEGMent:RESCale:YOFFset....................1304

MTESt:SEGMent:STATe <MaskTestName>, <MaskSegmIdx>,<State>
MTESt:SEGMent:STATe? <MaskTestName>, <MaskSegmIdx>

Enables and disables the mask segment. Disabled segments are not considered by running tests.

Parameters:

<State> ON | OFF

*RST: ON

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Asynchronous command

MTESt:SEGMent:ADD <MaskTestName>

Creates a new segment in the mask definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

Example: See Chapter 17.3.3.1, "Creating a User Mask", on page 1023

Usage: Setting only
Asynchronous command

MTESt:SEGMent:COUNt? <MaskTestName>

Returns the number of segments in the mask definition

Query parameters:

<MaskTestName> String with the name of the mask test

Return values:

<Count> Number of segments

Usage: Query only

MTESt:SEGMent:INSert <MaskTestName>, <MaskSegmIdx>

Inserts a new segment before the specified index in the mask definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:REMove <MaskTestName>, <MaskSegmIdx>

Removes the specified segment from the mask definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:CLEar <MaskTestName>

Deletes all mask segments of the indicated mask test.

Setting parameters:

<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:SEGMent:REGion <MaskTestName>, <MaskSegmIdx>,<Region>
MTESt:SEGMent:REGion? <MaskTestName>, <MaskSegmIdx>

Defines the region of the segment that builds the mask.

Parameters:

<Region> UPPer | LOWer | INNer

UPPer
the segment points are connected to a line, the display area above this line is the mask segment

LOWer
the segment points are connected to a line, the display area below this line is the mask segment

INNer
the segment points form a closed geometrical shape, which is the mask segment

*RST: INNer

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Example: See Chapter 17.3.3.1, "Creating a User Mask", on page 1023

Usage: Asynchronous command

MTESt:SEGMent:POINt:ADD <MaskTestName>, <MaskSegmIdx>

Adds a new point to the segment definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Example: See Chapter 17.3.3.1, "Creating a User Mask", on page 1023

Usage: Setting only
Asynchronous command

MTESt:SEGMent:POINt:INSert <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Inserts a new point before the specified mask segment point.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:POINt:REMove <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Removes the specified point from the mask segment.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:POINt:COUNt? <MaskTestName>, <MaskSegmIdx>

Returns the number of defined points for the specified mask segment.

Query parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Query only

MTESt:SEGMent:POINt:X <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>,<X>
MTESt:SEGMent:POINt:X? <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Defines the x-value of the mask segment point.

Parameters:

<X> Range: -100E+24 to 100E+24
Increment: 1E-6
*RST: 0
Default unit: s

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Example: See Chapter 17.3.3.1, "Creating a User Mask", on page 1023

Usage: Asynchronous command

MTESt:SEGMent:POINt:Y <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>,<Y>
MTESt:SEGMent:POINt:Y? <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Defines the y-value of the mask segment point.

Parameters:

<Y> Range: -100E+24 to 100E+24
Increment: 1E-6
*RST: 0
Default unit: V

Parameters for setting and query:

<MaskTestName>	String with the name of the mask test
<MaskSegmIdx>	Number of the segment. Counting starts from 0.
<MaskSegmPtIdx>	Number of the point. Counting starts from 0.
Example:	See Chapter 17.3.3.1, "Creating a User Mask", on page 1023
Usage:	Asynchronous command

MTESt:SEGMent:RESCale:RECalculate <MaskTestName>, <MaskSegmIdx>

Multiplies and adds the given x- and y-factors and offsets to the coordinates of all points of the selected mask segment.

Setting parameters:

<MaskTestName>	String with the name of the mask test
<MaskSegmIdx>	Number of the segment. Counting starts from 0.
Usage:	Setting only Asynchronous command

MTESt:SEGMent:RESCale:XFACtor <MaskTestName>, <MaskSegmIdx>,<ExpansFactX>
MTESt:SEGMent:RESCale:XFACtor? <MaskTestName>, <MaskSegmIdx>
MTESt:SEGMent:RESCale:YFACtor <MaskTestName>, <MaskSegmIdx>,<ExpansFactY>
MTESt:SEGMent:RESCale:YFACtor? <MaskTestName>, <MaskSegmIdx>

Stretches or compresses the selected mask segment in horizontal (XFACtor) or vertical direction (YFACtor). The x- or y-values of all points of the selected mask segment are multiplied with this factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.

Only takes effect after the `MTESt:SEGMent:RESCale:RECalculate` command.

Parameters:

<ExpansFactY>	Range: -100 to 100 Increment: 1 *RST: 1

Parameters for setting and query:

<MaskTestName>	String with the name of the mask test
<MaskSegmIdx>	Number of the segment. Counting starts from 0.
Usage:	Asynchronous command

MTESt:SEGMent:RESCale:XOFFset <MaskTestName>, <MaskSegmIdx>,<OffsetX>
MTESt:SEGMent:RESCale:XOFFset? <MaskTestName>, <MaskSegmIdx>

Moves the mask segment horizontally. The specified offset is added to the x-values of all points of the selected mask segment.

Only takes effect after the MTESt:SEGMent:RESCale:RECalculate command.

Parameters:

<OffsetX>	Range: -50 to 50 Increment: 1E-9 *RST: 0 Default unit: s

Parameters for setting and query:

<MaskTestName>	String with the name of the mask test
<MaskSegmIdx>	Number of the segment. Counting starts from 0.
Usage:	Asynchronous command

MTESt:SEGMent:RESCale:YOFFset <MaskTestName>, <MaskSegmIdx>,<OffsetY>
MTESt:SEGMent:RESCale:YOFFset? <MaskTestName>, <MaskSegmIdx>

Moves the mask segment vertically. The specified offset is added to the y-values of all points of the selected mask segment.

Only takes effect after the MTESt:SEGMent:RESCale:RECalculate command.

Parameters:

<OffsetY>	Range: -1000 to 1000 Increment: 1E-6 *RST: 0 Default unit: V

Parameters for setting and query:

<MaskTestName>	String with the name of the mask test
<MaskSegmIdx>	Number of the segment. Counting starts from 0.
Usage:	Asynchronous command

17.14.3 Mask Definition: Waveform Mask

The chapter contains commands required for the definition of waveform masks - MTESt:CTYPe is set to WFML.

MTESt:REFWfm.. 1304
MTESt:WFMLupdate.. 1305
MTESt:WFMRescale:XWIDth.. 1305
MTESt:WFMRescale:YWIDth.. 1305
MTESt:WFMRescale:YPOSition... 1306
MTESt:WFMRescale:YSTRetch... 1306

MTESt:REFWfm <MaskTestName>,<Source>
MTESt:REFWfm? <MaskTestName>

Sets the reference waveform from which the mask is created.

The reference waveform can be created before, or loaded from a file with REFCurve commands, see Chapter 17.10.2, "Reference Waveforms", on page 1183.

Parameters:

<Source>	REF1 | REFerence1 | REF2 | REFerence2 | REF3 | REFerence3 | REF4 | REFerence4 REF1 = REFerence1, REF2 = REFerence2, REF3 = REFerence3, REF4 = REFerence4: reference waveforms *RST: REF1

Parameters for setting and query:

<MaskTestName>	String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMLupdate <MaskTestName>

Creates the upper and lower mask limit from the envelope of the selected reference waveform. If the reference waveform was not defined before, it is created automatically from the mask test source waveform which is set wtih `MTESt:SOURce`.

Setting parameters:

<MaskTestName>	String containing the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:WFMRescale:XWIDth <MaskTestName>,<HorizontalWidth>
MTESt:WFMRescale:XWIDth? <MaskTestName>

Sets the width of the mask in horizontal direction. The specified factor in divisions is added to the positive x-values and subtracted from the negative x-values of the mask limits in relation to the source waveform of the mask.

Parameters:

<HorizontalWidth>	Range: 0 to 1000 Increment: 0.01 *RST: 0 Default unit: div

Parameters for setting and query:

<MaskTestName>	String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMRescale:YWIDth <MaskTestName>,<VerticalWidth>
MTESt:WFMRescale:YWIDth? <MaskTestName>

Sets the width of the waveform mask in vertical direction. The specified factor in divisions is added to the y-values of the upper mask limit and subtracted from the y-values of the lower mask limit. Thus, the upper half of the mask is pulled upwards, the lower half is pulled down.

Parameters:

<VerticalWidth> Vertical mask width in divisions

Range: 0 to 1000
Increment: 0.01
*RST: 0
Default unit: div

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMRescale:YPOSition <MaskTestName>,<VertPosi>
MTESt:WFMRescale:YPOSition? <MaskTestName>

Moves the mask vertically within the display.

Parameters:

<VertPosi> Range: -1000 to 1000
Increment: 0.01
*RST: 0
Default unit: div

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMRescale:YSTRetch <MaskTestName>,<VerticalStretch>
MTESt:WFMRescale:YSTRetch? <MaskTestName>

Sets the vertical scaling to stretch the mask in y-direction. The scaling axis is the horizontal line through the lowest value of the lower mask limit.

Parameters:

<VerticalStretch> Scale factor in %

Range: 10 to 1000
Increment: 1
*RST: 100
Default unit: %

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

17.14.4 Event Actions

MTESt:ONViolation:BEEP....1307
MTESt:ONViolation:STOP....1307
MTESt:ONViolation:PRINt....1307

MTESt:ONViolation:SAVewaveform.. 1308
MTESt:ONViolation:REPort...1308
MTESt:ONViolation:TRIGgerout... 1308
MTESt:ONViolation:RUNexec.. 1308

MTESt:ONViolation:BEEP <MaskTestName>,<Beep>
MTESt:ONViolation:BEEP? <MaskTestName>

Generates a beep sound for the specified event.

Parameters:
<Beep> NOACtion | SUCCess | VIOLation

See Chapter 17.4.5, "Event Parameter", on page 1038

*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:STOP <MaskTestName>,<StopAcq>
MTESt:ONViolation:STOP? <MaskTestName>

Stops data acquisition for the specified event.

Parameters:
<StopAcq> NOACtion | SUCCess | VIOLation

See Chapter 17.4.5, "Event Parameter", on page 1038

*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:PRINt <MaskTestName>,<Print>
MTESt:ONViolation:PRINt? <MaskTestName>

Prints a screenshot including the measurement results to the printer defined using `SYSTem:COMMunicate:PRINter:SELect<1..2>` for the specified event.

Parameters:
<Print> NOACtion | SUCCess | VIOLation

See Event Parameter

*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:SAVewaveform <MaskTestName>,<SaveWfm>
MTESt:ONViolation:SAVewaveform? <MaskTestName>

Saves the waveform data.

Parameters:

<SaveWfm> NOAction | SUCCess | VIOLation

See Chapter 17.4.5, "Event Parameter", on page 1038

*RST: NOAction

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:REPort <MaskTestName>,<Report>
MTESt:ONViolation:REPort? <MaskTestName>

Creates and saves a report of the current settings and results.

Parameters:

<Report> NOAction | SUCCess | VIOLation

See Chapter 17.4.5, "Event Parameter", on page 1038

*RST: NOAction

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:TRIGgerout <MaskTestName>,<TriggerOutPulse>
MTESt:ONViolation:TRIGgerout? <MaskTestName>

Creates a trigger out pulse on mask violation or successful completion of the test cycle.

Parameters:

<TriggerOutPulse> NOAction | SUCCess | VIOLation

*RST: NOAction

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:RUNexec <MaskTestName>,<RunExecutable>
MTESt:ONViolation:RUNexec? <MaskTestName>

Starts an external application if the command is set to `VIOLation` or `SUCCess`.

Use the following commands to set up the application:

- `EXECutable:NAME` on page 1069
- `EXECutable:PARameter` on page 1069

- EXECutable:WDIRectory on page 1069

Parameters:
<RunExecutable> NOACtion | SUCCess | VIOLation
See Chapter 17.4.5, "Event Parameter", on page 1038
*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

17.14.5 Mask Display

MTESt:LABel....1309
MTESt:REName....1309
MTESt:HIGHlight:INFinite....1309
MTESt:HIGHlight:STATe....1310
MTESt:HIGHlight:TIME....1310
MTESt:COLor:UNMatch....1310
MTESt:COLor:MATCh....1310
MTESt:COLor:BORDer....1311
MTESt:COLor:INTerior....1311

MTESt:LABel <LabelState>

Switches the display of the mask test name on or off.

To change the name of the mask test, use MTESt:REName on page 1309.

Parameters:
<LabelState> ON | OFF
*RST: ON

Usage: Asynchronous command

MTESt:REName <MaskTestName>, <NewName>
MTESt:REName? <MaskTestName>

Changes the name of the mask test.

Parameters:
<NewName> String with the new mask test name.

Parameters for setting and query:
<MaskTestName> String with the existing mask test name.

MTESt:HIGHlight:INFinite <State>

If ON, the mask hits are highlighted for an unlimited period of time.

Parameters:

<State>	ON | OFF	
	*RST:	OFF

Usage: Asynchronous command

MTESt:HIGHlight:STATe <HighlightHits>

If ON, the mask hits are highlighted on the screen. You can define the color and the time of the hit display.

Parameters:

<HighlightHits>	ON | OFF	
	*RST:	ON

Usage: Asynchronous command

MTESt:HIGHlight:TIME <HighlightTime>

Sets the time how long the mask hits are highlighted.

Parameters:

<HighlightTime>	Range:	0.05 to 50
	Increment:	0.1
	*RST:	1
	Default unit:	s

Usage: Asynchronous command

MTESt:COLor:UNMatch <ClrUnmatchedMsk>

Sets the color of masks segments that were not hit.

Parameters:

<ClrUnmatchedMsk> Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range:	0 (black) to 4294967295 (white)
Increment:	1
*RST:	2578098858

Usage: Asynchronous command

MTESt:COLor:MATCh <ClrMatchedPixel>

Sets the color of samples that violated the mask.

Parameters:

<ClrMatchedPixel> Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 4294901760

Usage: Asynchronous command

MTESt:COLor:BORDer <ClrMatchedMskBorder>

Sets the color of masks segments that were touched at the border. In this case, the resolution is not sufficient to detect if the mask was really hit or not. Zoom into the concerned area to see the actual result.

Parameters:

<ClrMatchedMskBorder> Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 2580309606

Usage: Asynchronous command

MTESt:COLor:INTerior <ClrMatchedMskInterior>

Sets the color of mask segments the signal has entered into.

Parameters:

<ClrMatchedMskInterior> Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 2578055168

Usage: Asynchronous command

17.14.6 Results

MTESt:RESult:STATe?....1312
MTESt:RESult[:RESult]?....1312
MTESt:RESult:COUNt:WAVeforms?....1312
MTESt:RESult:COUNt:REMaining?....1313
MTESt:RESult:COUNt:FWAVeforms?....1313
MTESt:RESult:COUNt:FAILures?....1313
MTESt:RESult:FRATe?....1314

MTESt:RESult:STATe? <MaskTestName>

Shows if the test is running or has finished. The state is set to "Finished" if no acquisitions remain (see `MTESt:RESult:COUNt:REMaining?` on page 1313).

Query parameters:
<MaskTestName>

Return values:

<State>	RUNNing | FINished	
	*RST:	RUNNing
Usage:	Query only Asynchronous command	

MTESt:RESult[:RESult]? <MaskTestName>

Returns the test result.

A test has failed if the number of sample hits or acquisition hits exceeds the limit of "Violation tolerance" hits (see `MTESt:TOLerance` on page 1297, `MTESt:RESult:COUNt:FAILures?` on page 1313 and `MTESt:RESult:COUNt:FWAVeforms?` on page 1313).

Query parameters:
<MaskTestName>

Return values:

<TestResult>	PASS | FAIL	
	*RST:	PASS
Usage:	Query only Asynchronous command	

MTESt:RESult:COUNt:WAVeforms? <MaskTestName>

Returns the number of tested acquisitions.

Query parameters:
<MaskTestName>

Return values:
<AcqsCompleted> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:REMaining? <MaskTestName>

Remaining acquisitions until "Average count / Nx Single count" is reached.

The value is useful if you test a specified number of acquisitions with action "Stop acquisition" on violation, or if the acquisition has been stopped manually.

See also: Chapter 9.3.4, "Running a Mask Test", on page 403.

Query parameters:
<MaskTestName>

Return values:
<AcqsRemaining> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:FWAVeforms? <MaskTestName>

Returns the number of acquisitions that contained at least one sample hit.

Query parameters:
<MaskTestName>

Return values:
<AcquisitionHits> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:FAILures? <MaskTestName>

Returns the number of sample hits that violated the mask.

Query parameters:
<MaskTestName>

Return values:
<SampleHits> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:FRATe? <MaskTestName>

Ratio of acquisition hits to the number of tested acquisitions.

Query parameters:
<MaskTestName>

Return values:

<FailRate>	Range:	-100E+24 to 100E+24
	*RST:	0
	Default unit:	%

Usage: Query only
Asynchronous command

17.15 Search

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- General Search Settings....1315
- Basic Trigger Search Conditions....1316
- Edge Search Conditions....1319
- Glitch Search Conditions....1319
- Interval Search Conditions....1320
- Runt Search Conditions....1322
- Slew Rate Search Conditions....1324
- Timeout Search Conditions....1327
- Width Search Conditions....1327
- Window Search Conditions....1329
- Data2Clock Search Conditions....1332
- Pattern Search Conditions....1334
- State Search Conditions....1338
- Search on Spectrum....1341
- Search Scope Settings....1342
- Noise Rejection....1345
- Search Results....1347

17.15.1 General Search Settings

SEARch:ADD..1315
SEARch:CLEar..1315
SEARch:REMove...1315
SEARch:SOURce...1315
SEARch:ONLine..1316
SEARch:ALL...1316

SEARch:ADD <SearchName>

Creates a new search definition with the specified name.

Setting parameters:

<SearchName> String with the name of the search

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Setting only
Asynchronous command

SEARch:CLEar <SearchName>

Clears the search results once to start a new search.

Setting parameters:

<SearchName> Search definition

Usage: Setting only
Asynchronous command

SEARch:REMove <Key>

Deletes the specified search definition.

Setting parameters:

<Key> String with the name of the search

Usage: Setting only
Asynchronous command

SEARch:SOURce <SearchName>,<Source>
SEARch:SOURce? <SearchName>

Defines the source on which the search conditions are applied. The source can be any analog or digital channel, math or reference waveform as well as a serial bus configured for a supported protocol.

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | SBUS1 | SBUS2 | SBUS3 | SBUS4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Parameters for setting and query:

<SearchName> String with the name of the search

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Asynchronous command

SEARch:ONLine <SearchName>,<OnlineState>

SEARch:ONLine? <SearchName>

If enabled, a search is performed repeatedly for each new data acquisition.

Parameters:

<OnlineState> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:ALL <SearchName>

Performs a search for all results on the existing data from the selected source.

Setting parameters:

<SearchName> Search definition

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Setting only

Asynchronous command

17.15.2 Basic Trigger Search Conditions

SEARch:TRIGger:DATatoclock[:STATe]....1317
SEARch:TRIGger:EDGE[:STATe]....1317
SEARch:TRIGger:GLITch[:STATe]....1317
SEARch:TRIGger:INTerval[:STATe]....1317
SEARch:TRIGger:PATTern[:STATe]....1317
SEARch:TRIGger:RUNT[:STATe]....1317

SEARch:TRIGger:SLEWrate[:STATe].. 1317
SEARch:TRIGger:STATe[:STATe]..1317
SEARch:TRIGger:TIMeout[:STATe].. 1317
SEARch:TRIGger:WIDTh[:STATe]..1317
SEARch:TRIGger:WINDow[:STATe]... 1317
SEARch:TRIGger:LEVel[:VALue]... 1318
SEARch:TRIGger:DATatoclock:ACOPy.. 1318
SEARch:TRIGger:EDGE:ACOPy...1318
SEARch:TRIGger:GLITch:ACOPy...1318
SEARch:TRIGger:INTerval:ACOPy... 1318
SEARch:TRIGger:PATTern:ACOPy... 1318
SEARch:TRIGger:RUNT:ACOPy... 1318
SEARch:TRIGger:SLEWrate:ACOPy... 1318
SEARch:TRIGger:STATe:ACOPy..1318
SEARch:TRIGger:TIMeout:ACOPy... 1318
SEARch:TRIGger:WIDTh:ACOPy... 1318
SEARch:TRIGger:WINDow:ACOPy.. 1318
SEARch:TRIGger:EDGE:BCOPy..1318

SEARch:TRIGger:DATatoclock[:STATe] <SearchName>,<State>
SEARch:TRIGger:DATatoclock[:STATe]? <SearchName>
SEARch:TRIGger:EDGE[:STATe] <SearchName>,<State>
SEARch:TRIGger:EDGE[:STATe]? <SearchName>
SEARch:TRIGger:GLITch[:STATe] <SearchName>,<State>
SEARch:TRIGger:GLITch[:STATe]? <SearchName>
SEARch:TRIGger:INTerval[:STATe] <SearchName>,<State>
SEARch:TRIGger:INTerval[:STATe]? <SearchName>
SEARch:TRIGger:PATTern[:STATe] <SearchName>,<State>
SEARch:TRIGger:PATTern[:STATe]? <SearchName>
SEARch:TRIGger:RUNT[:STATe] <SearchName>,<State>
SEARch:TRIGger:RUNT[:STATe]? <SearchName>
SEARch:TRIGger:SLEWrate[:STATe] <SearchName>,<State>
SEARch:TRIGger:SLEWrate[:STATe]? <SearchName>
SEARch:TRIGger:STATe[:STATe] <SearchName>,<State>
SEARch:TRIGger:STATe[:STATe]? <SearchName>
SEARch:TRIGger:TIMeout[:STATe] <SearchName>,<State>
SEARch:TRIGger:TIMeout[:STATe]? <SearchName>
SEARch:TRIGger:WIDTh[:STATe] <SearchName>,<State>
SEARch:TRIGger:WIDTh[:STATe]? <SearchName>
SEARch:TRIGger:WINDow[:STATe] <SearchName>,<State>
SEARch:TRIGger:WINDow[:STATe]? <SearchName>

Includes the search conditions for the selected trigger event type in the next search.

Parameters:

<State> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:LEVel[:VALue] <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel[:VALue]? <Key>, <SignalSource>

Sets the voltage of the trigger level that is used to determine other parameters.

Parameters:

<Value> Voltage value

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

SEARch:TRIGger:DATatoclock:ACOPy <SearchName>
SEARch:TRIGger:EDGE:ACOPy <SearchName>
SEARch:TRIGger:GLITch:ACOPy <SearchName>
SEARch:TRIGger:INTerval:ACOPy <SearchName>
SEARch:TRIGger:PATTern:ACOPy <SearchName>
SEARch:TRIGger:RUNT:ACOPy <SearchName>
SEARch:TRIGger:SLEWrate:ACOPy <SearchName>
SEARch:TRIGger:STATe:ACOPy <SearchName>
SEARch:TRIGger:TIMeout:ACOPy <SearchName>
SEARch:TRIGger:WIDTh:ACOPy <SearchName>
SEARch:TRIGger:WINDow:ACOPy <SearchName>

Copies the trigger event configuration from Trigger A for the selected channel source to the search condition settings.

See Chapter 5.3, "Trigger Types", on page 203.

Setting parameters:

<SearchName> Search definition

Usage: Setting only
Asynchronous command

SEARch:TRIGger:EDGE:BCOPy <SearchName>

Copies the trigger event configuration from trigger B for the selected channel source to the search condition settings.

Setting parameters:
<SearchName> String with name of the search

Usage: Setting only
Asynchronous command

17.15.3 Edge Search Conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1318

SEARch:TRIGger:EDGE:SLOPe..1319

SEARch:TRIGger:EDGE:SLOPe <SearchName>,<Slope>
SEARch:TRIGger:EDGE:SLOPe? <SearchName>

Selects the edge type.

Parameters:
<Slope> POSitive | NEGative | EITHer
See Chapter 17.4.3, "Slope Parameter", on page 1038.
*RST: POSitive

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

17.15.4 Glitch Search Conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1318

SEARch:TRIGger:GLITch:POLarity...1319
SEARch:TRIGger:GLITch:RANGe...1320
SEARch:TRIGger:GLITch:WIDTh...1320

SEARch:TRIGger:GLITch:POLarity <SearchName>,<Polarity>
SEARch:TRIGger:GLITch:POLarity? <SearchName>

Indicates the polarity of a pulse, that is the direction of the first pulse slope.

Parameters:
<Polarity> POSitive | NEGative | EITHer
See Chapter 17.4.4, "Polarity Parameter", on page 1038.
*RST: POSitive

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:GLITch:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:GLITch:RANGe? <SearchName>

Selects which glitches are identified: shorter or longer than the specified width (see `SEARch:TRIGger:GLITch:WIDTh` on page 1320).

Parameters:

<RangeMode> SHORter | LONGer

*RST: SHORter

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:GLITch:WIDTh <SearchName>,<Width>
SEARch:TRIGger:GLITch:WIDTh? <SearchName>

Sets the length of a glitch. The instrument triggers on pulses shorter or longer than this value (see also `SEARch:TRIGger:GLITch:RANGe` on page 1320).

You need to know the expected pulse widths of the circuit to set the glitch width correctly.

Parameters:

<Width> Range: 100E-12 to 1E-3
Increment: 100E-6
*RST: 1E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

17.15.5 Interval Search Conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1318

SEARch:TRIGger:INTerval:SLOPe....1320
SEARch:TRIGger:INTerval:DELTa....1321
SEARch:TRIGger:INTerval:RANGe....1321
SEARch:TRIGger:INTerval:WIDTh....1322

SEARch:TRIGger:INTerval:SLOPe <SearchName>,<Slope>
SEARch:TRIGger:INTerval:SLOPe? <SearchName>

Sets the edge for the search.

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Parameters for setting and query:

<SearchName> String parameter, name of the search definition

Usage: Asynchronous command

SEARch:TRIGger:INTerval:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:INTerval:DELTa? <SearchName>

Defines a range around the "Interval width" value (see `SEARch:TRIGger:INTerval:WIDTh` on page 1322).

Parameters:

<WidthDelta> Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:INTerval:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:INTerval:RANGe? <SearchName>

Selects how the range of an interval is defined based on the interval width and delta (see `SEARch:TRIGger:INTerval:WIDTh` on page 1322 and `SEARch:TRIGger:INTerval:DELTa` on page 1321).

Parameters:

<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulse intervals inside a given range. The range is defined by "Interv. width" and "±Delta".

OUTSide
Triggers on intervals outside a given range. The range definition is the same as for "Within" range.

SHORter
Triggers on intervals shorter than the given "Interv. width".

LONGer
Triggers on intervals longer than the given "Interv. width".

*RST: OUTSide

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:INTerval:WIDTh <SearchName>,<Width>
SEARch:TRIGger:INTerval:WIDTh? <SearchName>

Defines the time between two pulses.

Parameters:

<Width> Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

17.15.6 Runt Search Conditions

SEARch:TRIGger:RUNT:DELTa....1322
SEARch:TRIGger:RUNT:POLarity....1322
SEARch:TRIGger:RUNT:RANGe....1323
SEARch:TRIGger:RUNT:WIDTh....1323
SEARch:TRIGger:LEVel:RUNT:LOWer....1324
SEARch:TRIGger:LEVel:RUNT:UPPer....1324

SEARch:TRIGger:RUNT:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:RUNT:DELTa? <SearchName>

Defines a range around the given runt width.

Parameters:

<WidthDelta> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-12
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:RUNT:POLarity <SearchName>,<Polarity>
SEARch:TRIGger:RUNT:POLarity? <SearchName>

Indicates the polarity of a pulse, that is the direction of the first pulse slope.

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 17.4.4, "Polarity Parameter", on page 1038.

*RST: POSitive

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:RUNT:RANGe <SearchName>,<Mode>
SEARch:TRIGger:RUNT:RANGe? <SearchName>

Selects how the time limit of the runt pulse is defined based on the runt width and delta (see `SEARch:TRIGger:RUNT:WIDTh` on page 1323 and `SEARch:TRIGger:RUNT:DELTa` on page 1322).

Parameters:

<Mode> ANY | LONGer | SHORter | WITHin | OUTSide

ANY
Triggers on all runts fulfilling the level condition, without time limitation.

LONGer
Triggers on runts longer than the given "Runt width".

SHORter
Triggers on runts shorter than the given "Runt width".

WITHin
Triggers if the runt length is inside a given time range. The range is defined by "Runt width" and "±Delta".

OUTSide
Triggers if the runt length is outside a given time range. The range definition is the same as for "Within" range.

*RST: ANY

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:RUNT:WIDTh <SearchName>,<Width>
SEARch:TRIGger:RUNT:WIDTh? <SearchName>

For the ranges "Shorter" and "Longer", the runt width defines the maximum and minimum pulse width, respectively.

For the ranges "Within" and "Outside", the runt width defines the center of a range which is defined by "±Delta".

The range is defined using `SEARch:TRIGger:RUNT:RANGe` on page 1323.

Parameters:

<Width>	Range: 100E-12 to 864 Increment: 100E-9 *RST: 5E-9 Default unit: s

Parameters for setting and query:

<SearchName>	Search definition

Usage: Asynchronous command

SEARch:TRIGger:LEVel:RUNT:LOWer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:RUNT:LOWer? <Key>, <SignalSource>
SEARch:TRIGger:LEVel:RUNT:UPPer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:RUNT:UPPer? <Key>, <SignalSource>

Set the lower and upper voltage threshold, respectively.

Parameters:

<Value>	Voltage value

Parameters for setting and query:

<SearchName>	String with the name of the search
<SignalSource>	C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

17.15.7 Slew Rate Search Conditions

SEARch:TRIGger:SLEWrate:DELTa....1324
SEARch:TRIGger:SLEWrate:RANGe....1325
SEARch:TRIGger:SLEWrate:SLOPe....1325
SEARch:TRIGger:SLEWrate:TIME....1326
SEARch:TRIGger:LEVel:TRANsition:LOWer....1326
SEARch:TRIGger:LEVel:TRANsition:UPPer....1326

SEARch:TRIGger:SLEWrate:DELTa <SearchName>,<TimeDelta>
SEARch:TRIGger:SLEWrate:DELTa? <SearchName>

Defines a time range around the given slew rate.

Parameters:

<TimeDelta> Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:SLEWrate:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:SLEWrate:RANGe? <SearchName>

Selects how the time limit for the slew rate is defined. The time measurement starts when the signal crosses the first trigger level - the upper or lower level depending on the selected slope - and stops when the signal crosses the second level.

Parameters:

<RangeMode> INSRange | OUTRange | LTHan | GTHan

INSRange
Triggers on slew rates inside a given time range. The range is defined by "Slew rate" and "±Delta".

OUTRange
Triggers on slew rates outside a given time range. The range definition is the same as for "Within" range.

LTHan
Triggers on slew rates shorter than the given "Slew rate" limit.

GTHan
Triggers on slew rates longer than the given "Slew rate" limit.

*RST: GTHan

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:SLEWrate:SLOPe <SearchName>,<Slope>
SEARch:TRIGger:SLEWrate:SLOPe? <SearchName>

Selects the edge type.

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:SLEWrate:TIME <SearchName>,<Time>
SEARch:TRIGger:SLEWrate:TIME? <SearchName>

For the ranges "Within" and "Outside", the slew rate defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the slew rate defines the maximum and minimum slew rate limits, respectively.

The range is defined using `SEARch:TRIGger:SLEWrate:RANGe`.

Parameters:

<Time> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-12
Default unit: s

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:LEVel:TRANsition:LOWer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:TRANsition:LOWer? <Key>, <SignalSource>
SEARch:TRIGger:LEVel:TRANsition:UPPer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:TRANsition:UPPer? <Key>, <SignalSource>

Set the lower and upper voltage thresholds, respectively. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope.

Parameters:

<Value> Voltage value

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

17.15.8 Timeout Search Conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1318

SEARch:TRIGger:TIMeout:RANGe ... 1327
SEARch:TRIGger:TIMeout:TIME ... 1327

SEARch:TRIGger:TIMeout:RANGe <SearchName>,<TimeoutMode>
SEARch:TRIGger:TIMeout:RANGe? <SearchName>

Selects the relation of the signal level to the trigger level:

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH
The signal level stays above the trigger level.

LOW
The signal level stays below the trigger level.

EITHer
The signal level stays above or below the trigger level.

*RST: HIGH

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:TIMeout:TIME <SearchName>,<Time>
SEARch:TRIGger:TIMeout:TIME? <SearchName>

Defines the time limit for the timeout at which the instrument triggers.

Parameters:

<Time> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

17.15.9 Width Search Conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1318

SEARch:TRIGger:WIDTh:DELTa ... 1328
SEARch:TRIGger:WIDTh:POLarity ... 1328
SEARch:TRIGger:WIDTh:RANGe ... 1328
SEARch:TRIGger:WIDTh:WIDTh ... 1329

SEARch:TRIGger:WIDTh:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:WIDTh:DELTa? <SearchName>

Defines a range around the given width value (see also `SEARch:TRIGger:WIDTh:WIDTh` on page 1329).

Parameters:

<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Asynchronous command

SEARch:TRIGger:WIDTh:POLarity <SearchName>,<Polarity>
SEARch:TRIGger:WIDTh:POLarity? <SearchName>

Indicates the polarity of a pulse, that is the direction of the first pulse slope.

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 17.4.4, "Polarity Parameter", on page 1038.

*RST: POSitive

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WIDTh:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:WIDTh:RANGe? <SearchName>

Selects how the range of a pulse width is defined in relation to the width and delta (see `SEARch:TRIGger:WIDTh:WIDTh` on page 1329 and `SEARch:TRIGger:WIDTh:DELTa` on page 1328).

Parameters:

<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range of the pulse width is defined by "Width" and "±Delta".

OUTSide
Triggers on pulses outside a given range. The range definition is the same as for "Within" range.

SHORter
Triggers on pulses shorter than the given "Width".

LONGer
Triggers on pulses longer than the given "Width".

*RST: WITHin

Parameters for setting and query:

<SearchName> Search definition

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Asynchronous command

SEARch:TRIGger:WIDTh:WIDTh <SearchName>,<Width>
SEARch:TRIGger:WIDTh:WIDTh? <SearchName>

For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

The range is defined using `SEARch:TRIGger:WIDTh:RANGe`.

Parameters:

<Width> Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Asynchronous command

17.15.10 Window Search Conditions

SEARch:TRIGger:WINDow:DELTa....1330
SEARch:TRIGger:WINDow:RANGe....1330
SEARch:TRIGger:WINDow:TIMerange....1331
SEARch:TRIGger:WINDow:WIDTh....1331
SEARch:TRIGger:LEVel:WINDow:LOWer....1332
SEARch:TRIGger:LEVel:WINDow:UPPer....1332

SEARch:TRIGger:WINDow:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:WINDow:DELTa? <SearchName>

Defines a range around the "Width" value (see `SEARch:TRIGger:WINDow:WIDTh` on page 1331).

Parameters:

<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WINDow:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:WINDow:RANGe? <SearchName>

Selects how the signal run is compared with the window.

Parameters:

<RangeMode> ENTer | EXIT | WITHin | OUTSide

ENTer
Triggers when the signal crosses the upper or lower level and thus enters the window made up of these two levels.

EXIT
Triggers when the signal leaves the window.

WITHin
Triggers if the signal stays between the upper and lower level for a specified time. The time is defined in various ways by the `SEARch:TRIGger:WINDow:TIMerange` command.

OUTSide
Triggers if the signal stays above the upper level or below the lower level for a specified time. The time is also defined by the `SEARch:TRIGger:WINDow:TIMerange` command.

*RST: ENTer

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WINDow:TIMerange <SearchName>,<TimeRangeMode>
SEARch:TRIGger:WINDow:TIMerange? <SearchName>

Selects how the time limit of the window is defined. Time conditioning is available for the vertical conditions "WITHin" and "OUTSide" (see `SEARch:TRIGger:WINDow:RANGe` on page 1330).

Parameters:

<TimeRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers if the signal stays inside or outside the vertical window limits at least for the time *Width - Delta* and for *Width + Delta* at the most.

OUTSide
"Outside" is the opposite definition of "Within". The instrument triggers if the signal stays inside or outside the vertical window limits for a time shorter than *Width - Delta* or longer than *Width + Delta*.

SHORter
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

LONGer
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

*RST: WITHin

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WINDow:WIDTh <SearchName>,<Width>
SEARch:TRIGger:WINDow:WIDTh? <SearchName>

For the ranges "Within" and "Outside", the width defines the center of a time range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", it defines the maximum and minimum time lapse, respectively.

The range is defined using `SEARch:TRIGger:WINDow:RANGe`.

Parameters:

<Width> Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:LEVel:WINDow:LOWer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:WINDow:LOWer? <Key>, <SignalSource>
SEARch:TRIGger:LEVel:WINDow:UPPer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:WINDow:UPPer? <Key>, <SignalSource>

Set the lower and upper voltage limits for the window.

Parameters:

<Value> Voltage value

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

17.15.11 Data2Clock Search Conditions

Data level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1318

SEARch:TRIGger:DATatoclock:CEDGe.......... 1332
SEARch:TRIGger:DATatoclock:CLEVel.......... 1333
SEARch:TRIGger:DATatoclock:CSOurce.......... 1333
SEARch:TRIGger:DATatoclock:HTIMe.......... 1333
SEARch:TRIGger:DATatoclock:STIMe.......... 1334

SEARch:TRIGger:DATatoclock:CEDGe <SearchName>,<ClockEdge>
SEARch:TRIGger:DATatoclock:CEDGe? <SearchName>

Sets the edge of the clock signal to define the time reference point for the setup and hold time.

Parameters:

<ClockEdge> POSitive | NEGative | EITHer

See Chapter 17.4.3, "Slope Parameter", on page 1038.

*RST: POSitive

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:DATatoclock:CLEVel <SearchName>,<ClockLevel>
SEARch:TRIGger:DATatoclock:CLEVel? <SearchName>

Sets the voltage level for the clock signal. Both this command and `SEARch:TRIGger:DATatoclock:CEDGe` define the starting point for calculation of the setup and hold time.

Parameters:

<ClockLevel>	Range: -10 to 10 Increment: 1E-3 *RST: 0 Default unit: V

Parameters for setting and query:

<SearchName>	Search definition

Usage: Asynchronous command

SEARch:TRIGger:DATatoclock:CSOurce <SearchName>,<ClockSource>
SEARch:TRIGger:DATatoclock:CSOurce? <SearchName>

Selects the waveform used for the clock signal.

Parameters:

<ClockSource>	C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 Source of the clock signal, see Chapter 17.4.2, "Waveform Parameter", on page 1037 *RST: C1W1

Parameters for setting and query:

<SearchName>	Search definition name

Usage: Asynchronous command

SEARch:TRIGger:DATatoclock:HTIMe <SearchName>,<HoldTime>
SEARch:TRIGger:DATatoclock:HTIMe? <SearchName>

Sets the minimum time **after** the clock edge while the data signal must stay steady above or below the data level.

The hold time can be negative. In this case, the setup time is always positive. The setup/hold interval starts before the clock edge (setup time) and ends before the clock edge (hold time). If you change the negative hold time, the setup time is adjusted by the intrument.

Parameters:

<HoldTime>	Range: -99.999E-9 to 0.1 Increment: 1E-9 *RST: 0 Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:DATatoclock:STIMe <SearchName>,<SetupTime>
SEARch:TRIGger:DATatoclock:STIMe? <SearchName>

Sets the minimum time **before** the clock edge while the data signal must stay steady above or below the data level.

The setup time can be negative. In this case, the hold time is always positive. The setup/hold interval starts after the clock edge (setup time) and ends after the clock edge (hold time). If you change the negative setup time, the hold time is adjusted by the intrument.

Parameters:

<SetupTime> Range: -99.999E-9 to 0.1
Increment: 1E-9
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

17.15.12 Pattern Search Conditions

SEARch:TRIGger:PATTern:A[:ENABle]....1335
SEARch:TRIGger:PATTern:B[:ENABle]....1335
SEARch:TRIGger:PATTern:C[:ENABle]....1335
SEARch:TRIGger:PATTern:D[:ENABle]....1335
SEARch:TRIGger:PATTern:A:LOGic....1335
SEARch:TRIGger:PATTern:B:LOGic....1335
SEARch:TRIGger:PATTern:C:LOGic....1335
SEARch:TRIGger:PATTern:D:LOGic....1335
SEARch:TRIGger:PATTern:AB:LOGic....1336
SEARch:TRIGger:PATTern:CD:LOGic....1336
SEARch:TRIGger:PATTern:ABCD:LOGic....1336
SEARch:TRIGger:PATTern:MODE....1336
SEARch:TRIGger:PATTern:TIMeout:MODE....1337
SEARch:TRIGger:PATTern:TIMeout[:TIME]....1337
SEARch:TRIGger:PATTern:WIDTh:RANGe....1337
SEARch:TRIGger:PATTern:WIDTh[:WIDTh]....1338
SEARch:TRIGger:PATTern:WIDTh:DELTa....1338

SEARch:TRIGger:PATTern:A[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:A[:ENABle]? <Searchname>
SEARch:TRIGger:PATTern:B[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:B[:ENABle]? <Searchname>
SEARch:TRIGger:PATTern:C[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:C[:ENABle]? <Searchname>
SEARch:TRIGger:PATTern:D[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:D[:ENABle]? <Searchname>

Enables the channel to be considered in the pattern search. The trigger source channel is selected by default.

- A[:ENABle]: CH1
- B[:ENABle]: CH2
- C[:ENABle]: CH3
- D[:ENABle]: CH4

Digital channels are not available.

Parameters:
<State> ON | OFF

Parameters for setting and query:
<Searchname> String with name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:A:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:A:LOGic? <Searchname>
SEARch:TRIGger:PATTern:B:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:B:LOGic? <Searchname>
SEARch:TRIGger:PATTern:C:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:C:LOGic? <Searchname>
SEARch:TRIGger:PATTern:D:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:D:LOGic? <Searchname>

Defines the logic for the indicated channel:

- A: CH1
- B: CH2
- C: CH3
- D: CH4

Parameters:
<Operator> DIRect | NOT

DIRect
Input value remains unchanged

NOT
Input value is inverted

Parameters for setting and query:
<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:AB:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:AB:LOGic? <Searchname>
SEARch:TRIGger:PATTern:CD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:CD:LOGic? <Searchname>
SEARch:TRIGger:PATTern:ABCD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:ABCD:LOGic? <Searchname>

Defines the logical combination of the indicated channels after evaluating the previous logical operations:

- AB: CH1 and CH2
- CD: CH3 and CH4
- ABCD: result of AB and CD

Parameters:

<Operator> AND | NAND | OR | NOR

AND: logical AND, conjunctive combination
NAND:logical NOT AND
OR:logical OR, disjunctive combination
NOR: logical NOT OR

Parameters for setting and query:

<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:MODE <SearchName>,<Mode>
SEARch:TRIGger:PATTern:MODE? <SearchName>

Adds additional time limitation to the pattern definition.

Parameters:

<Mode> OFF | TIMeout | WIDTh

OFF
No time limitation. The event is found if the pattern condition is fulfilled.

TIMeout
Defines how long the result of the pattern condition stays high or low. The duration of the timeout is defined using `SEARch:TRIGger:PATTern:TIMeout[:TIME]` The result state is defined using `SEARch:TRIGger:PATTern:TIMeout:MODE`.

WIDTh
Defines a time range for keeping up the true result of the pattern condition. The range is defined using `SEARch:TRIGger:PATTern:WIDTh:RANGe`.

*RST: OFF

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:TIMeout:MODE <SearchName>,<TimeoutMode>
SEARch:TRIGger:PATTern:TIMeout:MODE? <SearchName>

Defines the condition for the timeout.

Parameters:
<TimeoutMode> HIGH | LOW | EITHer

EITHer
High or low, the pattern remains unchanged for the given time.

*RST: HIGH

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:TIMeout[:TIME] <SearchName>,<Time>
SEARch:TRIGger:PATTern:TIMeout[:TIME]? <SearchName>

Defines how long the result of the pattern condition must keep the given state.

Parameters:
<Time> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-9
Default unit: s

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:WIDTh:RANGe <SearchName>,<WidthRangeMode>
SEARch:TRIGger:PATTern:WIDTh:RANGe? <SearchName>

Defines the time range of a pulse width for keeping up the true result of the pattern condition. The width and delta are specified using `SEARch:TRIGger:PATTern:WIDTh[:WIDTh]` and `SEARch:TRIGger:PATTern:WIDTh:DELTa`.

Parameters:
<WidthRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin | OUTSide
Triggers on pulses inside or outside a given range. The range is defined by the width ±delta.

SHORter | LONGer
Triggers on pulses shorter or longer than the given width.

*RST: WITHin

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:WIDTh[:WIDTh] <SearchName>,<Width>
SEARch:TRIGger:PATTern:WIDTh[:WIDTh]? <SearchName>

For the ranges WITHin and OUTSide, the width defines the center of a range that is defined by the limits ±delta.

For the ranges SHORter and LONGer, the width defines the maximum and minimum pulse width, respectively.

To set the range mode, use `SEARch:TRIGger:PATTern:WIDTh:RANGe`. To set the delta value, use `SEARch:TRIGger:PATTern:WIDTh:DELTa`.

Parameters:

<Width> Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:WIDTh:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:PATTern:WIDTh:DELTa? <SearchName>

Defines a range around the width value specified using `SEARch:TRIGger:PATTern:WIDTh[:WIDTh]`.

Parameters:

<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

17.15.13 State Search Conditions

SEARch:TRIGger:STATe:CSOurce 1339
SEARch:TRIGger:STATe:CEDGe 1339
SEARch:TRIGger:STATe:CLEVel 1339
SEARch:TRIGger:STATe:A[:ENABle] 1340

SEARch:TRIGger:STATe:B[:ENABle].. 1340
SEARch:TRIGger:STATe:C[:ENABle].. 1340
SEARch:TRIGger:STATe:D[:ENABle].. 1340
SEARch:TRIGger:STATe:A:LOGic... 1340
SEARch:TRIGger:STATe:B:LOGic... 1340
SEARch:TRIGger:STATe:C:LOGic... 1340
SEARch:TRIGger:STATe:D:LOGic... 1340
SEARch:TRIGger:STATe:AB:LOGic... 1341
SEARch:TRIGger:STATe:CD:LOGic... 1341
SEARch:TRIGger:STATe:ABCD:LOGic... 1341

SEARch:TRIGger:STATe:CSOurce <SearchName>,<Source>
SEARch:TRIGger:STATe:CSOurce? <SearchName>

Sets the source of the clock signal.

Parameters:
<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:CEDGe <SearchName>,<ClockEdge>
SEARch:TRIGger:STATe:CEDGe? <SearchName>

Sets the trigger edge of the clock signal.

Parameters:
<ClockEdge> POSitive | NEGative | EITHer

*RST: POSitive

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:CLEVel <SearchName>,<ClockLevel>
SEARch:TRIGger:STATe:CLEVel? <SearchName>

Sets the trigger level of the clock signal.

The command has the same effect as with `SEARch:TRIGger:LEVel[:VALue]`.

Parameters:
<ClockLevel> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:A[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:A[:ENABle]? <Searchname>
SEARch:TRIGger:STATe:B[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:B[:ENABle]? <Searchname>
SEARch:TRIGger:STATe:C[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:C[:ENABle]? <Searchname>
SEARch:TRIGger:STATe:D[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:D[:ENABle]? <Searchname>

Enables the channel to be considered in the state search. You can enable all channel signals except for the trigger source.

- A[:ENABle]: CH1
- B[:ENABle]: CH2
- C[:ENABle]: CH3
- D[:ENABle]: CH4

Digital channels are not available.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:A:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:A:LOGic? <Searchname>
SEARch:TRIGger:STATe:B:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:B:LOGic? <Searchname>
SEARch:TRIGger:STATe:C:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:C:LOGic? <Searchname>
SEARch:TRIGger:STATe:D:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:D:LOGic? <Searchname>

Defines the logic for the indicated channel:

- A: CH1
- B: CH2
- C: CH3
- D: CH4

Parameters:

<Operator> DIRect | NOT

DIRect
Input value remains unchanged

NOT
Input value is inverted

Parameters for setting and query:

<Searchname>	String with the name of the search

Usage:	Asynchronous command

SEARch:TRIGger:STATe:AB:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:AB:LOGic? <Searchname>
SEARch:TRIGger:STATe:CD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:CD:LOGic? <Searchname>
SEARch:TRIGger:STATe:ABCD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:ABCD:LOGic? <Searchname>

Defines the logical combination of the indicated channels after evaluating the previous logical operations:

- AB: CH1 and CH2
- CD: CH3 and CH4
- ABCD: result of AB and CD

Parameters:

<Operator> AND | NAND | OR | NOR

AND: logical AND, conjunctive combination
NAND:logical NOT AND
OR:logical OR, disjunctive combination
NOR: logical NOT OR

Parameters for setting and query:

<Searchname>	String with the name of the search

Usage:	Asynchronous command

17.15.14 Search on Spectrum

CURSor<m>:PEXCursion <Value>

Defines the minimum level by which the waveform must rise or fall so that it will be identified as a maximum or a minimum by the search functions.

Suffix:

<m>	The suffix is irrelevant

Parameters:

<Value>
Range: 0 to 100
Increment: 1
*RST: 5
Default unit: dB

Usage:	Asynchronous command

CURSor<m>:THReshold <Value>

Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

Suffix:

<m> The suffix is irrelevant.

Parameters:

<Value> Threshold in dBm

Usage: Asynchronous command

Firmware/Software: Version 2.70

17.15.15 Search Scope Settings

SEARch:GATE[:STATe]....1342
SEARch:GATE:MODE....1342
SEARch:GATE:SHOW....1343
SEARch:GATE:ABSolute:STARt....1343
SEARch:GATE:ABSolute:STOP....1343
SEARch:GATE:RELative:STARt....1343
SEARch:GATE:RELative:STOP....1344
SEARch:GATE:ZCOupling....1344
SEARch:GATE:ZDIagram....1344

SEARch:GATE[:STATe] <SearchName>,<State>
SEARch:GATE[:STATe]? <SearchName>

Performs the search only on the defined gate area of the source waveform.

Parameters:

<State> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:MODE <SearchName>,<Mode>
SEARch:GATE:MODE? <SearchName>

Defines whether the gate settings are configured using absolute or relative values.

Parameters:

<Mode> ABS | REL

*RST: ABS

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:SHOW <SearchName>,<DisplayState>
SEARch:GATE:SHOW? <SearchName>

If enabled, the gate area is indicated in the source diagram.

Parameters:

<DisplayState> ON | OFF
*RST: OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:ABSolute:STARt <SearchName>,<Start>
SEARch:GATE:ABSolute:STARt? <SearchName>

Defines the starting value for the gate.

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:ABSolute:STOP <SearchName>,<Stop>
SEARch:GATE:ABSolute:STOP? <SearchName>

Defines the end value for the gate.

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:RELative:STARt <SearchName>,<RelativeStart>
SEARch:GATE:RELative:STARt? <SearchName>

Defines the starting value for the gate.

Parameters:

<RelativeStart>	Range: 0 to 100 Increment: 0.1 *RST: 0 Default unit: %

Parameters for setting and query:

<SearchName>	Search definition

Usage: Asynchronous command

SEARch:GATE:RELative:STOP <SearchName>,<RelativeStop>
SEARch:GATE:RELative:STOP? <SearchName>

Defines the end value for the gate.

Parameters:

<RelativeStop>	Range: 0 to 100 Increment: 0.1 *RST: 100 Default unit: %

Parameters for setting and query:

<SearchName>	Search definition

Usage: Asynchronous command

SEARch:GATE:ZCOupling <SearchName>,<ZoomCoupling>
SEARch:GATE:ZCOupling? <SearchName>

If enabled, the gate area is set to the limits of a zoom area.

The zoom diagramm is selected using `SEARch:GATE:ZDIagram`.

Parameters:

<ZoomCoupling>	ON | OFF *RST: OFF

Parameters for setting and query:

<SearchName>	String with the name of the search

Usage: Asynchronous command

SEARch:GATE:ZDIagram <SearchName>, <DiagramName>, <ZoomName>
SEARch:GATE:ZDIagram? <SearchName>

Selects the zoom to which the gate area is set if `SEARch:GATE:ZCOupling` is set to "ON".

Parameters:

<ZoomDiagram>	String with the name of the diagram, where the zoom is defined

Parameters for setting and query:

<SearchName>	String with the name of the search

Example: Prerequisite: Search1 and Zoom2 are available.

```
SEARch:GATE:ZCOupling 'Search1',ON
SEARch:GATE:ZDIagram 'Search1','Diagram1','Zoom2'
SEARch:GATE:ZDIagram? 'Search1'
<-- Diagram1;Zoom2
SEARch:GATE:STATe 'Search1', ON
SEARch:GATE:SHOW 'Search2', ON
```

Enables the zoom coupling to define the gate,and selects Zoom2 as gate area. Zoom2 is based on Diagram1. Then the gate is activated and displayed.

Usage: Asynchronous command

17.15.16 Noise Rejection

SEARch:TRIGger:LEVel:NOISe:ABSolute....................1345
SEARch:TRIGger:LEVel:NOISe:MODE....................1345
SEARch:TRIGger:LEVel:NOISe:RELative....................1346
SEARch:TRIGger:LEVel:NOISe[:STATe]....................1346

SEARch:TRIGger:LEVel:NOISe:ABSolute <SearchName>, <SignalSource>, <Value>

Defines the trigger hysteresis, a range in absolute values around the trigger level. If the signal jitters inside this range and crosses the trigger level, no trigger event is detected.

Parameters:

<Value> Hysteresis value

Setting parameters:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

SEARch:TRIGger:LEVel:NOISe:MODE <SearchName>, <SignalSource>, <Mode>
SEARch:TRIGger:LEVel:NOISe:MODE? <Key>, <SignalSource>

Defines whether absolute values or relative values to the vertical scaling are used as a hysteresis for noise rejection.

Parameters:

<Mode> ABS | REL

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the trigger waveform, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

SEARch:TRIGger:LEVel:NOISe:RELative <SearchName>, <SignalSource>, <Value>

Defines a range around the trigger level in relative values. If the signal jitters inside this range and crosses the trigger level, no trigger event is detected.

Parameters:

<Value> Hysteresis value in %

Setting parameters:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

SEARch:TRIGger:LEVel:NOISe[:STATe] <SearchName>, <SignalSource>, <State>
SEARch:TRIGger:LEVel:NOISe[:STATe]? <Key>, <SignalSource>

If enabled, the noise reject settings for the waveform are considered for the search.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 17.4.2, "Waveform Parameter", on page 1037

Usage: Asynchronous command

17.15.17 Search Results

SEARch:RESDiagram:HORZ:ABSolute:POSition.. 1347
SEARch:RESDiagram:HORZ:ABSolute:SPAN.. 1347
SEARch:RESDiagram:HORZ:MODE..1348
SEARch:RESDiagram:HORZ:RELative:POSition... 1348
SEARch:RESDiagram:HORZ:RELative:SPAN..1348
SEARch:RESDiagram:SHOW... 1349
SEARch:RESDiagram:VERT:ABSolute:POSition..1349
SEARch:RESDiagram:VERT:ABSolute:SPAN... 1349
SEARch:RESDiagram:VERT:MODE... 1349
SEARch:RESDiagram:VERT:RELative:POSition.. 1350
SEARch:RESDiagram:VERT:RELative:SPAN...1350
SEARch:RESult:LIMit... 1350
SEARch:RESult:SHOW.. 1351
SEARch:RESult:SORT:ASCending... 1351
SEARch:RESult:SORT[:MODE]...1351
SEARch:RESult[:ALL]?...1352

SEARch:RESDiagram:HORZ:ABSolute:POSition <SearchName>,<Position>
SEARch:RESDiagram:HORZ:ABSolute:POSition? <SearchName>

Defines the x-value of the centerpoint of the zoom area.

Parameters:
<Position> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:ABSolute:SPAN <SearchName>,<Span>
SEARch:RESDiagram:HORZ:ABSolute:SPAN? <SearchName>

Defines the width of the zoom area.

Parameters:

<Span>	Range:	0 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:MODE <SearchName>,<Mode>
SEARch:RESDiagram:HORZ:MODE? <SearchName>

Defines whether absolute or relative values are used to specify the x-axis values.

Parameters:

<Mode> ABS | REL

*RST: ABS

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:RELative:POSition <SearchName>,<RelPosi>
SEARch:RESDiagram:HORZ:RELative:POSition? <SearchName>

Defines the x-value of the centerpoint of the zoom area.

Parameters:

<RelPosi>	Range:	0 to 100
	Increment:	0.1
	*RST:	100
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:RELative:SPAN <SearchName>,<RelativeSpan>
SEARch:RESDiagram:HORZ:RELative:SPAN? <SearchName>

Defines the width of the zoom area.

Parameters:

<RelativeSpan>	Range:	1E-15 to 100
	Increment:	0.1
	*RST:	1
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:SHOW <SearchName>,<ShwSearchWind>
SEARch:RESDiagram:SHOW? <SearchName>

If enabled, a zoom window is displayed for the currently selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.

Parameters:
<ShwSearchWind> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:ABSolute:POSition <SearchName>,<Position>
SEARch:RESDiagram:VERT:ABSolute:POSition? <SearchName>

Defines the y-value of the centerpoint of the zoom area.

Parameters:
<Position> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:ABSolute:SPAN <SearchName>,<Span>
SEARch:RESDiagram:VERT:ABSolute:SPAN? <SearchName>

Defines the height of the zoom area.

Parameters:
<Span> Range: 0 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:MODE <SearchName>,<Mode>
SEARch:RESDiagram:VERT:MODE? <SearchName>

Defines whether absolute or relative values are used to specify the y-axis values.

Parameters:

<Mode>	ABS | REL	
	*RST:	ABS

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:RELative:POSition <SearchName>,<RelPosi>
SEARch:RESDiagram:VERT:RELative:POSition? <SearchName>

Defines the y-value of the centerpoint of the zoom area.

Parameters:

<RelPosi>	Range:	0 to 100
	Increment:	0.1
	*RST:	100
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:RELative:SPAN <SearchName>,<RelativeSpan>
SEARch:RESDiagram:VERT:RELative:SPAN? <SearchName>

Defines the height of the zoom area.

Parameters:

<RelativeSpan>	Range:	1E-15 to 100
	Increment:	0.1
	*RST:	1
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult:LIMit <SearchName>,<ResultListLimit>
SEARch:RESult:LIMit? <SearchName>

Defines the maximum number of entries in the search result table.

Parameters:

<ResultListLimit>	Range:	1 to 1000
	Increment:	1
	*RST:	100

Parameters for setting and query:

<SearchName> Search definition

Example: See Chapter 17.3.4.1, "Searching for a Pulse of Specified Width", on page 1023

Usage: Asynchronous command

SEARch:RESult:SHOW <SearchName>,<ShowResultTable>
SEARch:RESult:SHOW? <SearchName>

Displays or hides the search result table.

Parameters:
<ShowResultTable> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult:SORT:ASCending <SearchName>,<SortAscending>
SEARch:RESult:SORT:ASCending? <SearchName>

If enabled, the results are listed in asscending order, i.e. the smallest value at the top.

Parameters:
<SortAscending> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult:SORT[:MODE] <SearchName>,<SortMode>
SEARch:RESult:SORT[:MODE]? <SearchName>

Sorts the search result table by x-value position or value of the result.

Parameters:
<SortMode> POSition | VALue

POSition
Sorts the search result table by the x-value position.

VALue
Sorts the search result table by the value of the result.

*RST: POSition

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult[:ALL]? <SearchName>

Returns all search results.

Query parameters:

<SearchName> Search definition

Return values:

<Data> List of search results, separated by commas. For each result, six values are returned:
1. Acquisition index, currently always 0.
2. X-position of the search result
3. Y-position of the search result, currently not relevant
4. Type of the search result (Edge, Glitch, ...)
5. Slope or polarity of the search result
6. For runt, glitch, width, and window searches, the value contains the width. For timeout and interval searches, it contains the timeout. For transition searches, it contains the slew rate. For all other searches, the value is not relevant.
If a value is not relevant, 9.91E+37 is returned.

Example:

```
SEAR:RES? 'Search1'
0,1.5375e-007,-84,Edge,Positive,9.91E+37,
0,5.3e-008,-84,Edge,Positive,9.91E+37
```

The query returns two search results for edge search on rising edges at X-position 153,75 ns and 53 ns.

Usage: Query only
Asynchronous command

17.16 Data Management

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- Instrument Settings....................1353
- Savesets and One File....................1360
- Autonaming....................1361
- Waveform Data Transmission....................1362
- Waveform Data Export to File....................1362
- Waveform Histogram Export to File....................1369
- Numeric Results....................1371

- CSV Export.. 1372
- Long Term Measurement Results and Measurement Histogram Export to File..1372
- Screenshots... 1374
- Reports...1379

17.16.1 Instrument Settings

The Mass MEMory subsystem provides commands to access the storage media and to save and reload instrument settings.

File and directory names

The <file_name> and <directory_name> parameters are strings. Some commands use a fixed directory; for others the <file_name> can contain the complete path including the drive name and all subdirectories, e.g. '`C:\TEMP\TRASH\test.txt`' for the file named test.txt in the TEMP\TRASH subdirectory of the internal hard disk drive C:\. If no complete path is specified, the file location is relative to the current directory, queried with `MMEMory:CDIRectory?`. The file name itself may contain the period as a separator for extensions.

File and directory names can be chosen according to Windows™ conventions; the restrictions placed on file names known from DOS systems do not apply. All letters and numbers are allowed, as well as the special characters "_", "^", "$", "~", "!", "#", "%", "&", "-", "{", "}", "(", ")", "@" and "`". Reserved file names are CON, AUX, COM1, ..., COM4, LPT1, ..., LPT3, NUL and PRN.

The use of wildcards ? and * is not allowed.

MMEMory:DRIVes?...1353
MMEMory:MSIS...1354
MMEMory:DCATalog?... 1354
MMEMory:DCATalog:LENGth?... 1354
MMEMory:CDIRectory...1355
MMEMory:MDIRectory.. 1355
MMEMory:RDIRectory...1355
MMEMory:CATalog?...1355
MMEMory:CATalog:LENGth?.. 1356
MMEMory:COPY.. 1356
MMEMory:MOVE... 1357
MMEMory:DELete..1357
MMEMory:DATA.. 1357
MMEMory:ATTRibute... 1358
MMEMory:SAV.. 1358
MMEMory:RCL.. 1359
MMEMory:STORe:STATe..1359
MMEMory:LOAD:STATe...1360

MMEMory:DRIVes?

Returns a list of the logical drives of the instrument as configured in the operating system.

Return values:

<Drive> List of strings, for example, `"C:\","F:\","H:\"`

Usage: Query only

MMEMory:MSIS [<msus>]

Changes the default storage device to the indicated drive or network server.

Parameters:

<msus> String parameter. Drives are indicated with their drive letter, network servers require the UNC format.

Example: `MMEM:MSIS 'C:'`

Example: `MMEM:MSIS '\\server1\share1'`

MMEMory:DCATalog? [<PathName>]

Returns the subdirectories of the current or of a specified directory.

Query parameters:

<PathName> String parameter to specify the directory. If the directory is omitted, the command queries the contents of the current directory, to be set and queried with `MMEMory:CDIRectory`.

Return values:

<FileEntry> Names of the subdirectories separated by colons. The first two strings are related to the parent directory.

Example:
```
MMEM:DCAT?
".","..","Documents and Settings","Program
Files","temp"
```

Usage: Query only

MMEMory:DCATalog:LENGth? [<PathName>]

Returns the number of subdirectories of the current or of a specified directory. The number includes the parent directory strings "." and ".." and corresponds to the number of strings returned by the `MMEMory:DCATalog?` command.

Query parameters:

<PathName> String parameter to specify the directory. If the directory is omitted, the command queries the contents of the current directory, to be set and queried with `MMEMory:CDIRectory`.

Return values:

<FileEntryCount> Number of parent and subdirectories.

Example:
```
MMEM:DCAT:LENG?
5
```

Usage: Query only

MMEMory:CDIRectory [<DirectoryName>]

Changes the default directory for file access.

Parameters:

<DirectoryName> String parameter to specify the directory. If the string also contains a drive letter or network server name, the command `MMEMory:MSIS` is executed implicitely.

*RST: "\"

Example: `MMEM:CDIR 'C:\USER\DATA'`

Usage: SCPI confirmed

MMEMory:MDIRectory <DirectoryName>

Creates a new directory with the specified name.

Setting parameters:

<DirectoryName> String parameter to specify the new directory. If the path consists of several subdirectories, the complete tree will be created if necessary. If no drive letter or server name is indicated, the directory is created on the default soreage device specified with `MMEMory:MSIS`.

Example: `MMEM:MDIR 'C:\USER\DATA'`

Usage: Setting only

MMEMory:RDIRectory <DirectoryName>

Deletes the specified directory.

Setting parameters:

<DirectoryName> String parameter to specify the directory to be deleted.

Example: `MMEM:RDIR 'C:\USER\TEST'`

Usage: Setting only

MMEMory:CATalog? [<PathName>][, <Format>]

Returns the a list of files contained in the specified directory. The result corresponds to the number of files returned by the MMEMory:CATalog:LENgth command.

Query parameters:

<PathName> String parameter to specify the directory. If the directory is omitted, the command queries directory specified with `MMEMory:CDIRectory`.

<Format> ALL | WTIMe

ALL: Extended result including file, date, time and attributes
WTIMe: Extended result including file, date, time

Return values:

<UsedMemory> Total amount of storage currently used in the directory, in bytes.

<FreeMemory> Total amount of storage available in the directory, in bytes.

<FileEntry> All files and subdirectories of the directory are listed with their file name, format and size in bytes. The first two strings are related to the parent directory.

Example:

```
MMEM:CAT? 'C:\USER\DATA'
529479,1831777894400,".,DIR,0","..,DIR,0",
"Backup,DIR,0","CSS,DIR,0","DATEN,DIR,0",
"Commands.jar,BIN,529479","FAVORITES,DIR,0",
"LOG,DIR,0","DATA,DIR,0","test,DIR,0",
"TotalCMD,DIR,0"
```

Usage: Query only
SCPI confirmed

MMEMory:CATalog:LENGth? [<PathName>]

Returns the number of files and subdirectories of the current or specified directory. The number includes the parent directory strings "." and ".." and it corresponds to the number of <FileEntry> strings returned by the `MMEMory:CATalog?` command.

Query parameters:

<PathName> String parameter, directory to be queried. If the directory is omitted, the current directory is queried, specified with `MMEMory:CDIRectory`.

Return values:

<Count> Number of files and subdirectories including parent directory entries.

Example:

```
MMEM:CDIR 'C:\USER\DATA'
MMEM:CAT:LENG?
11
```

Usage: Query only

MMEMory:COPY <FileSource>[, <FileDestination>]

Copies an existing file to a new file.

Setting parameters:

<FileSource> String parameter, contains name and path of the file to be copied. Wildcards (* and ?) are allowed.

<FileDestination> String parameter, contains name and path of the new file. If the file already exists, it is overwritten without notice. If no file destination is specified, the source file is written to the current directory specified with `MMEMory:CDIRectory`.

Example: `MMEM:COPY 'C:\Users\Public\Documents\Rohde-Schwarz\RTx\RefWaveforms\RefCurve_2011-03-16*.bin','E:'`
Copies all reference waveforms saved on March 16, 2011 to an external storage medium, mapped to drive `E:\`.

Usage: Setting only
SCPI confirmed

MMEMory:MOVE <FileSource>, <FileDestination>

Moves the specified file to a new location on the same drive and renames it.

Setting parameters:

<FileSource> String parameter, contains name and path of the file to be copied.
Wildcards (* and ?) are allowed. Therefore, specify a directory for <FileDestination>. Renaming is not possible.

<FileDestination> String parameter, contains name and path of the new file. If no path is specified, the <FileSource> directory is used - the file is renamed.

Example: `MMEM:MOVE 'C:\USER\DATA\SETUP.CFG','C:\STORE'`
Moves the file "Setup.cfg" from the directory `C:\USER\DATA` to `C:\STORE`.

Usage: Setting only
SCPI confirmed

MMEMory:DELete <FileName>

Removes the specified file(s). To delete directories, use `MMEMory:RDIRectory`.

Setting parameters:

<FileName> String parameter to specify the name and directory of the file to be removed. Wildcards (* and ?) are allowed.
If no path is defined, the current directory is used, specified with `MMEMory:CDIRectory`.

Example: `MMEM:DEL '*.CFG'`
Deletes all cfg files from the current directory.

Usage: Setting only
SCPI confirmed

MMEMory:DATA <FileName>, <Data>
MMEMory:DATA? <FileName>

Stores data in the specified file to the storage location specified using `MMEMory:CDIRectory`.

Parameters:

<Data> <block>

488.2 block data format. The delimiter EOI must be selected to achieve correct data transfer.
The block begins with character '#'. The next digit is the length of the length information, followed by this given number of digits providing the number of bytes in the following binary data.

Parameters for setting and query:

<FileName> String parameter, the name of the file the data is stored to.

Example: `MMEM:DATA 'abc.txt', #216This is the file`
#2: the length infomation has two digits
16: the binary data has 16 bytes

Example: `MMEM:DATA? 'abc.txt'`
Returns the data from file `abc.txt`.

MMEMory:ATTRibute <FileName>, <Attributes>
MMEMory:ATTRibute? <FileName>

Sets file attributes for the specified file(s). The command can be used for files only.

Setting parameters:

<Attributes> String with attributes and setting information.
'+' before the attribute: sets the attribute
'-' before the attribute: deletes the attribute
'R': read only
'A': archive file
'S': system file
'H': hidden file

Parameters for setting and query:

<FileName> String parameter, contains name and path of the file. Wildcards (* and ?) are allowed.

Return values:

<FileEntry> String containing: "<file_name>,<file_attributes>"

Example: `MMEM:ATTR 'C:\USER\DATA\*.LOG', '-R -A'`
Deletes the read-only and archive attributes from all LOG files in the directory`C:\USER\DATA\*.LOG`.

Example:
```
MMEM:ATTR? 'C:\USER\DATA\*.*'
"Datei1.LOG,A", "Datei2.LOG,A",
"Datei3.LOG,ASH", "Datei4.DLL,RSH",
"Datei5.INI,SH"
```

MMEMory:SAV <FileDestination>

Stores the current instrument settings to the specified file.

This command has the same effect as the combination of `*SAV` and `MMEMory:STORe:STATe`.

Parameters:

<FileDestination> String parameter specifying path and filename of the target file. Wildcards are not allowed.

Example: `MMEM:SAV 'C:\mysavefile.dfl'`
Saves the current instrument settings to the file `mysavefile.dfl` located in the directory C:\.

Usage: Event

MMEMory:RCL <FileSource>

Restores the instrument settings from the specified file.

This command has the same effect as the combination of `MMEMory:LOAD:STATe` and `*RCL`.

Parameters:

'<FileSource>' String parameter specifying the path and filename of the source file. Wildcards are not allowed.

Example: `MMEM:RCL 'C:\mysavefile.dfl'`
Loads and activates the instrument settings from the file `mysavefile.dfl` located in the directory C:\.

Usage: Event

MMEMory:STORe:STATe <MemoryNumber>, <FileName>

Stores the instrument settings from the specified internal memory to the specified file. To store the current instrument settings to the internal memory, use `*SAV` first.

Setting parameters:

<MemoryNumber> Number of the internal memory

Range: 1 to 99

<FileName> String parameter specifying the complete path and filename of the source file.

Example:
```
*SAV 4
MMEM:STORe:STATe 4,'C:
\Settings\Settings_1051.dfl'
```
Saves current instrument settings to the internal memory number 4. Then stores the settings from the internal memory number 4 to the file `C:\Settings\Settings_1051.dfl`.

Usage: Setting only

MMEMory:LOAD:STATe <MemoryNumber>, <FileName>

Loads the instrument settings from the specified file to the specified internal memory. After the file has been loaded, the settings must be activated using a `*RCL` command.

Setting parameters:

<MemoryNumber> Number of the internal memory

Range: 1 to 99

<FileName> String parameter specifying the complete path and filename of the source file.

Example:
```
MMEM:LOAD:STATe 4,'C:
\Settings\Settings_1051.dfl'
*RCL 4
```
Loads instrument settings from the file `C:\Settings\Settings_1051.dfl` to the internal memory number 4, and then activates the settings in internal memory number 4.

Usage: Setting only

17.16.2 Savesets and One File

SAVeset:CONFig:PREView....................1360
SAVeset:ONEFile:NAME....................1360
SAVeset:ONEFile:OPEN....................1360
SAVeset:ONEFile:SAVE....................1361

SAVeset:CONFig:PREView <Include>

If set to OFF, the saveset is stored without the preview image to reduce the file size.

Use the command each time before you save a saveset.

Parameters:

<Include> ON | OFF

*RST: ON

SAVeset:ONEFile:NAME <Name>

Sets the path and the file name of the One File. The file format is ZIP.

Parameters:

<Name> String parameter

SAVeset:ONEFile:OPEN

Loads the One File that is specified with `SAVeset:ONEFile:NAME`.

Usage: Event

SAVeset:ONEFile:SAVE

Usage: Event

Saves the One File data to the file that is specified with `SAVeset:ONEFile:NAME`.

17.16.3 Autonaming

MMEMory:AUTonaming:PREFix ... 1361
MMEMory:AUTonaming:USERtext ... 1361
MMEMory:AUTonaming:DATE ... 1361
MMEMory:AUTonaming:INDex ... 1361
MMEMory:AUTonaming:TIME ... 1361
MMEMory:AUTonaming:TEXT ... 1361
MMEMory:AUTonaming:DEFaultpath ... 1361
MMEMory:AUTonaming:RESPath ... 1362
MMEMory:AUTonaming:RESall ... 1362

MMEMory:AUTonaming:PREFix <State>
MMEMory:AUTonaming:USERtext <State>
MMEMory:AUTonaming:DATE <State>
MMEMory:AUTonaming:INDex <State>
MMEMory:AUTonaming:TIME <State>

Includes or excludes the name part in the file name pattern for automatic file name generation. This name is used as the default file name.

The prefix indicates the type of data that is saved, for example, Histogram, RefCurve, Settings.

To define a user text, use `MMEMory:AUTonaming:TEXT`.

Parameters:
<State> ON | OFF
*RST: ON

MMEMory:AUTonaming:TEXT <NameString>

Defines a text, that can be included in the autonaming pattern.

Parameters:
<NameString> String parameter

MMEMory:AUTonaming:DEFaultpath <Path>

Sets the path where data and settings files will be stored. The factory default path is:

- `"C:\Users\Public\Documents\Rohde-Schwarz\RTx"` if no USB flash drive is connected

- Drive letter of the USB flash drive, for example, "E:\" or "F:\" if a USB flash drive is connected.

Parameters:

<Path> String parameter

MMEMory:AUTonaming:RESPath

Resets the path for file operations to the factory default path.

Usage: Event

MMEMory:AUTonaming:RESall

Resets all autonaming settings to the default value, including the path.

Usage: Event

17.16.4 Waveform Data Transmission

The R&S RTE provides specific data export commands for the various waveform types. The commands transmit the data of the waveform points from the instrument to the controlling computer. The data can be used in MATLAB, for example.

The commands are described in the relevant chapters:

- Analog waveforms: Chapter 17.8.6, "Waveform Data", on page 1084
- Reference waveforms: Chapter 17.10.2.3, "Waveform Data Export", on page 1186
- Math waveforms: Chapter 17.10.3, "Mathematics", on page 1189
- Spectrum waveforms: Chapter 17.13.2, "Waveform Data", on page 1291
- Logic channels: Chapter 17.18.5, "MSO Data ", on page 1884

17.16.5 Waveform Data Export to File

The resulting files of waveforms exports are described in Chapter 11.2.1, "Waveform Export Files", on page 439.

The export settings for manual operation are explained in Chapter 11.2.2, "Waveforms - Export Settings", on page 445.

EXPort:WAVeform:SOURce.......1363
EXPort:WAVeform:MULTichannel.......1364
CHANnel<m>:EXPortstate.......1364
EXPort:WAVeform:NAME.......1364
EXPort:WAVeform:SAVE.......1365
EXPort:WAVeform:SCOPe.......1365
EXPort:WAVeform:STARt.......1365
EXPort:WAVeform:STOP.......1366
EXPort:WAVeform:ZOOM.......1366
EXPort:WAVeform:CURSorset.......1366

EXPort:WAVeform:MEAS....1366
EXPort:WAVeform:DLOGging....1367
EXPort:WAVeform:TIMestamps....1368
EXPort:WAVeform:INCXvalues....1368
EXPort:WAVeform:RAW....1368
EXPort:WAVeform:DISPlayoff....1369
EXPort:WAVeform:FASTexport....1369

EXPort:WAVeform:SOURce <Source>

Selects the waveform to be exported to file.

The commands takes effect if `EXPort:WAVeform:MULTichannel` is `OFF`.

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | BATHtub1 | BMEas1 | SREsponse1 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTrack1 | TJSPectrum1 | DDJSpectrum1 | PJSPectrum1 | RJOSpectrum1 | SEYE1 | BATHtub2 | BMEas2 | SREsponse2 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHist2 | RJOHist2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJOTrack2 | TJSPectrum2 | DDJSpectrum2 | PJSPectrum2 | RJOSpectrum2 | SEYE2 | NBAThtub1 | NBMeas1 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNOSpectrum1 | NBAThtub2 | NBMeas2 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSpectrum2 | PJDHist1 | BUJHist1 | RJHist1 | PJDTrack1 | PJDSpectrum1 | PJDHist2 | BUJHist2 | RJHist2 | PJDTrack2 | PJDSpectrum2 | PNDHist1 | BUNHist1 | RNHist1 | PNDTrack1 | PNDSpectrum1 | PNDHist2 | BUNHist2 | RNHist2 | PNDTrack2 | PNDSpectrum2

*RST: C1W1

Example: See Chapter 17.3.5.2, "Exporting Waveform Data to File", on page 1024

Usage: Asynchronous command

EXPort:WAVeform:MULTichannel <MltChXpt>

Enables or disables the multichannel export.

If you enable the multichannel export, all active channels are included to the export data. You can change the export state using the `CHANnel<m>:EXPortstate` command.

If multichannel export is disabled, select the waveform to be exported using the `EXPort:WAVeform:SOURce` command.

Note that `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?` returns the data of all channels that are selected for export, no matter of the channel suffix.

Parameters:

<MltChXpt> ON | OFF

*RST: OFF

Usage: Asynchronous command

CHANnel<m>:EXPortstate <ExportState>

Includes or excludes the indicated channel in waveform export. The data of channel waveform 1 is exported.

The commands takes effect if `EXPort:WAVeform:MULTichannel` is `ON`.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<ExportState> ON | OFF

If you enable the multichannel export, the export state of all active channels is automatically set ON.

*RST: OFF

Usage: Asynchronous command

EXPort:WAVeform:NAME <FileName>

Sets the file name, file format and path to save the waveform to.

See also: Chapter 11.2.1, "Waveform Export Files", on page 439

Parameters:

<FileName> String with path and file name with extension .xml, .bin, or .csv

Example:

```
EXPort:WAVeform:NAME 'C:\temp\Export_Ch11.xml'
EXPort:WAVeform:SAVE
```

Saves the waveform data in XML format to `C:\temp\Export_Ch1.xml`.

Example: `EXPort:WAVeform:NAME 'C:\temp\Export_Ch2.bin'`
`EXPort:WAVeform:SAVE`
Saves the waveform data in binary format to
`C:\temp\Export_Ch2.bin`.

Usage: Asynchronous command

EXPort:WAVeform:SAVE

Saves the waveform(s) to the file specified with `EXPort:WAVeform:NAME`. The file format is also set using the `...NAME` command.

Example: See Chapter 17.3.5.2, "Exporting Waveform Data to File", on page 1024

Usage: Event

EXPort:WAVeform:SCOPe <Scope>

Defines the part of the waveform record that has to be stored.

Parameters:
<Scope> WFM | ZOOM | CURSor | GATE | MANual

WFM
Complete waveform

ZOOM
Data included in the zoom area if a zoom is defined for the source waveform.

CURSor
Data between the cursor lines if a cursor measurement is defined for the source waveform.

GATE
data included in the measurement gate if a gated measurement is defined for the source waveform.

MANual
Saves the data between user-defined start and stop values to be set with `EXPort:WAVeform:STARt` and `EXPort:WAVeform:STOP`.

*RST: WFM

Example: See Chapter 17.3.5.2, "Exporting Waveform Data to File", on page 1024

Usage: Asynchronous command

EXPort:WAVeform:STARt <Start>

Sets the start value of the waveform section for export, if `EXPort:WAVeform:SCOPe` is set to `Manual`.

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01
Default unit: s

Usage: Asynchronous command

EXPort:WAVeform:STOP <Stop>

Sets the end value of the waveform section for export, if `EXPort:WAVeform:SCOPe` is set to `Manual`.

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01
Default unit: s

Usage: Asynchronous command

EXPort:WAVeform:ZOOM <ZoomDiagram>

Sets the zoom area to be used for limited data export if `EXPort:WAVeform:SCOPe` is set to `ZOOM`.

Parameters:

<ZoomDiagram> Name of the diagram on which the zoom area is based.

Example: See "Exporting Interleaved x/y Data of a Zoom to CSV File" on page 1027

Usage: Asynchronous command

EXPort:WAVeform:CURSorset <Cursorset>

Sets the cursor set to be used for limited data export if `EXPort:WAVeform:SCOPe` is set to `CURSor`.

Parameters:

<Cursorset> CURSOR1 | CURSor1 | CURSOR2 | CURSor2 | CURSOR3 | CURSor3 | CURSOR4 | CURSor4

CURSOR1 = CURSor1, CURSOR2 = CURSor2, CURSOR3 = CURSor3, CURSOR4 = CURSor4

Usage: Asynchronous command

EXPort:WAVeform:MEAS <MeasGate>

Sets the gate to be used for limited data export if `EXPort:WAVeform:SCOPe` is set to `GATE`.

Parameters:

<MeasGate> MEAS1 | MEAS2 | MEAS3 | MEAS4 | MEAS5 | MEAS6 | MEAS7 | MEAS8

Measurement for which the gate is defined.

Example: See "Exporting Raw Data of a Measurement Gate to BIN File" on page 1026

Usage: Asynchronous command

EXPort:WAVeform:DLOGging <DataLogging>

The command enables the export of subsequent acquisitions of the selected waveforms. The waveforms are taken from a running Nx Single acquisition (data logging, history is disabled), or from the history (multiple waveforms, history is enabled).

If the history is disabled (`CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]`) and data logging is enabled, a specified number of waveforms is transferred to file directly during RUN Nx SINGLE acquisition. Enabling data logging stops a running acquisition. Set the number of acquisitions to be acquired and stored with `ACQuire:COUNt` and start export using `RUNSingle`.

If the history is enabled, the subsequent waveforms are taken from the history. Specify the range with `CHANnel<m>[:WAVeform<n>]:HISTory:STARt` and `CHANnel<m>[:WAVeform<n>]:HISTory:STOP`. Then play the history with `CHANnel<m>[:WAVeform<n>]:HISTory:PLAY`.

The commands `EXPort:WAVeform:SAVE`, `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?` and `CHANnel<m>[:WAVeform<n>]:DATA:HEADer?` are not available if data logging is enabled. The `RUNContinous` command disables data logging.

If data logging is off, and the history is enabled, one waveform out of the history is written to file. Specify the waveform using `CHANnel<m>[:WAVeform<n>]:HISTory:CURRent` and save it using `EXPort:WAVeform:SAVE`.

Parameters:

<DataLogging> ON | OFF

*RST: OFF

Example: See:
"Exporting Multiple Running Acquisitions of a Single Waveform to XML File" on page 1027
"Exporting Multiple Acquisition of the History to XML File" on page 1028
"Exporting a Single Acquisition of the History to BIN File" on page 1028

Usage: Asynchronous command

EXPort:WAVeform:TIMestamps <UseTimestamps>

Exports the relative timestamps of all history waveforms to the waveform data file. The time is written at the beginning of each waveform record.

Parameters:

<UseTimestamps> ON | OFF

*RST: OFF

Usage: Asynchronous command

EXPort:WAVeform:INCXvalues <IncHorValues>

Includes horizontal values in the retrieved data (time or frequency values, depending on the waveform). X and Y-values are written alternately to the file. If disabled, only Y-values - mostly voltage values - are written. The X-values are always returned in 64 bit real format, regardless of the defined data format.

The setting is not available for the export of raw data.

The command affects the content of export files as well as data retrieved with:

- `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?`
- `CALCulate:MATH<m>:DATA[:VALues]?`
- `REFCurve<m>:DATA[:VALues]?`
- `DIGital<m>:DATA[:VALues]?`
- `BUS<m>:PARallel:DATA[:VALues]?`
- `ZVC:Z<m>:I<n>:DATA[:VALues]?`
- `ZVC:Z<m>:V<n>:DATA[:VALues]?`

Parameters:

<IncHorValues> ON | OFF

*RST: OFF

Example: See:
"Exporting Interleaved x/y Data of a Single Waveform to CSV File" on page 1026
"Exporting Interleaved x/y Data of a Zoom to CSV File" on page 1027

Usage: Asynchronous command

EXPort:WAVeform:RAW <RawValues>

Enables the export of raw sample data, and sets the data format to integer 8 bit. In high definition acquisition mode, the data format is integer 16 bit. Fot INT16, you can set the byte order using the `FORMat:BORDer` command.

The raw format reduces the file size but changes also the precision of the values.

Currently, the setting is not available for the export of digital channel data and data of R&S RT-ZVC channels.

The raw export of interleaved X/Y values is not supported.

Parameters:

<RawValues> ON | OFF

*RST: OFF

Example: See:
"Exporting Raw Data of a Single Waveform to BIN File"
on page 1025
"Exporting Raw Data of a Measurement Gate to BIN File"
on page 1026

Usage: Asynchronous command

EXPort:WAVeform:DISPlayoff <FastExport>

Enables or disables the display update during an Nx Single acquisition.

Parameters:

<FastExport> ON | OFF

ON: Disables the display update for maximum export speed.
OFF: Enables the display update. The export is slower.

*RST: OFF

Usage: Asynchronous command

EXPort:WAVeform:FASTexport <Enable>

To improve the performance of data export to file, the measurements are performed slower while the data export speeds up.

Setting parameters:

<Enable> ON | OFF

Example: See Chapter 17.3.5.2, "Exporting Waveform Data to File", on page 1024

Usage: Setting only
Asynchronous command

17.16.6 Waveform Histogram Export to File

EXPort:HISTogram:SELect....1369
EXPort:HISTogram:INCidence....1370
EXPort:HISTogram:NAME....1370
EXPort:HISTogram:SAVE....1370
EXPort:HISTogram:DATA?....1370

EXPort:HISTogram:SELect <Name>

Selects the histogram to be exported.

Parameters:

<Name> String with the histogram name.

Example: See "Exporting Histogram Data to File" on page 1022

Usage: Asynchronous command

EXPort:HISTogram:INCidence <Incidence>

Sets the mode of exported data: relative or absolute frequency of amplitude values.

Parameters:

<Incidence> ABS | REL

*RST: REL

Example: See "Exporting Histogram Data to File" on page 1022

Usage: Asynchronous command

EXPort:HISTogram:NAME <Path>

Sets the file name and path to save the histogram to.

Parameters:

<Path> String with path and file name. The file extension defines the file format: XML, CSV, or BIN.

Example: See "Exporting Histogram Data to File" on page 1022

Usage: Asynchronous command

EXPort:HISTogram:SAVE

Saves the histogram to the file specified with `EXPort:HISTogram:NAME`.

Example: See "Exporting Histogram Data to File" on page 1022

Usage: Event
Asynchronous command

EXPort:HISTogram:DATA?

Transfers the histogram data to the controlling computer. The data can be used in MATLAB, for example.

To set the export data format, use `FORMat[:DATA]`.

Return values:

<Data> List of values according to the format settings and `EXPort:HISTogram:INCidence`.
See also: Chapter 11.2.3, "Waveform Histogram", on page 450

Example: See "Transferring Histogram Data" on page 1022.

Usage: Query only

17.16.7 Numeric Results

EXPort:RESult:SELect..1371
EXPort:RESult:NUMeric..1371
EXPort:RESult:NAME...1371
EXPort:RESult:SAVE..1371

EXPort:RESult:SELect <Item>, [<STATe>]
EXPort:RESult:SELect? <Item>

Select the results that you want to save to file. All results are written into one file. To save several result boxes into one file, use the command several times, one command for each result box.

Parameters:
<STATe> ON | OFF

Parameters for setting and query:
<Item> String parameter, contains the name of the result box as written in the "Numeric Results" dialog box.

Example: `EXPort:RESult:SELect 'Meas Results',1`
Selects the 'Meas Results' box for export of numeric values.

EXPort:RESult:NUMeric <Numeric>

If ON, the result values are saved without unit and with more decimal places.

Parameters:
<Numeric> ON | OFF
*RST: OFF

EXPort:RESult:NAME <ExportFilename>

Sets the path, the file name, and the file format for the numeric results file. Available file formats are CSV and HTML.

Parameters:
<ExportFilename> String parameter

EXPort:RESult:SAVE

Saves the selected result boxes to the file that is specified with `EXPort:RESult:NAME`.

Usage: Event

17.16.8 CSV Export

EXPort:RESult:DECSymbol........ 1372
EXPort:RESult:DELimiter........ 1372

EXPort:RESult:DECSymbol <DecimalSymbol>

Sets if point or comma is used as a decimal symbol.

Parameters:

<DecimalSymbol> POINt | COMMa

*RST: POINt

EXPort:RESult:DELimiter <CsvDelimiter>

Selects the list separator symbol from a list.

Parameters:

<CsvDelimiter> SEMICOLON | COMMA | SPACE | TAB | COLON

*RST: COMMA

17.16.9 Long Term Measurement Results and Measurement Histogram Export to File

EXPort:MEASurement:SELect........ 1372
EXPort:MEASurement:TYPE........ 1372
EXPort:MEASurement:NAME........ 1373
EXPort:MEASurement:SAVE........ 1373
EXPort:MEASurement:DATA?........ 1373

EXPort:MEASurement:SELect <SelcMeas>

Selects the measurement for export of long term or measurement histogram data.

Parameters:

<SelcMeas> MEAS1 | MEAS2 | MEAS3 | MEAS4 | MEAS5 | MEAS6 | MEAS7 | MEAS8

*RST: MEAS1

Example: See "Exporting Long-Term Measurement Data to File" on page 1022

Usage: Asynchronous command

EXPort:MEASurement:TYPE <ExportType>

You can export the result data of the long term measurement, or the measurement histogram, or the track data.

To export the measurement histogram, it must be enabled using `MEASurement<m>:STATistics:HISTogram`.

To export the long term results, the long term measurement must be enabled using `MEASurement<m>:LTMeas[:STATe]`.

To export a track, the track must be enabled before. Track measurements require an option, see "Enable (Track)" on page 350.

Parameters:

<ExportType>	LONGterm | HISTogram | TRACk LONGTERM = LONGterm, HISTOGRAM = HISTogram *RST: HISTOGRAM
Example:	See "Exporting Long-Term Measurement Data to File" on page 1022
Usage:	Asynchronous command

EXPort:MEASurement:NAME <Path>

Sets the file name and path to save the long term or measurement histogram data to.

Parameters:

<Path>	String with path and file name. The file extension defines the file format: XML, CSV, or BIN.
Example:	See "Exporting Long-Term Measurement Data to File" on page 1022
Usage:	Asynchronous command

EXPort:MEASurement:SAVE

Saves the long term or measurement histogram results to the file specified using `EXPort:MEASurement:NAME`.

The measurement data can be exported as absolute or relative values, which is defined using `EXPort:HISTogram:INCidence`.

Example:	See "Exporting Long-Term Measurement Data to File" on page 1022
Usage:	Event

EXPort:MEASurement:DATA?

Transfers the long term measurement data to the controlling computer. The data can be used in MATLAB, for example.

To set the export data format, use `FORMat[:DATA]`.

Return values:

<Data>	List of values according to the format settings

Long term data:
If statistics are enabled (MEASurement<m>:STATistics[:ENABle]), six values for each long term point are returned: maximum, minimum, average, standard deviation, number of measured results per long term point, number of waveforms per long term point.
If statistics are disabled, the current value of each long term point is returned.
For measurement histograms, absolute values are returned.
See also: Chapter 11.2.5, "Result Analysis", on page 454

Example: See "Transferring Long-Term Measurement Data" on page 1022

Usage: Query only

17.16.10 Screenshots

The HCOPy subsystem and some other commands control the output of display information for documentation purposes on output devices (printer and clipboard) or files (also for report files). The instrument allows two independent output configurations which can be set separately with the suffix.

Note that the remote mode is intended for maximum performance. Therefore, the display does not follow the remote commmands consistantly. To get a correct screenshot, turn the display on using SYST:DISP:UPD ON.

HCOPy:DESTination<1..2> 1374
MMEMory:NAME 1375
HCOPy:DEVice<m>:LANGuage 1375
HCOPy:PAGE:ORIentation<1..2> 1376
HCOPy:DEVice<m>:COLor 1376
HCOPy:DEVice<m>:INVerse 1376
HCOPy:WBKG 1376
HCOPy:CMAP<m>:DEFault 1377
HCOPy:SSD 1377
HCOPy:ISBA 1377
HCOPy:IMMediate<m>[:DUM] 1378
HCOPy:IMMediate<m>:NEXT 1378
SYSTem:COMMunicate:PRINter:ENUMerate:FIRSt? 1378
SYSTem:COMMunicate:PRINter:ENUMerate[:NEXT]? 1379
SYSTem:COMMunicate:PRINter:SELect<1..2> 1379

HCOPy:DESTination<1..2> <Medium>

Selects the output medium: file, printer or clipboard.

Suffix:

<1..2> Selects the output configuration.

Parameters:

<Medium> MMEM | SYST:COMM:PRIN | SYST:COMM:CLIP

String parameter

MMEM
Directs the display image to a file. The MMEMory:NAME command defines the file name. The file format is defined with HCOPy:DEVice<m>:LANGuage.

SYST:COMM:PRIN
Directs the display image to the printer. The printer is selected with the SYSTem:COMMunicate:PRINter:SELect<1..2> command. The HCOPy:DESTination command should always be sent after setting the printer.

SYST:COMM:CLIP
Directs the hardcopy to the clipboard.

*RST: SYST:COMM:CLIP

Example: `HCOP:DEST 'SYST:COMM:PRIN'`
See also Chapter 17.3.5.1, "Saving a Screenshot to File", on page 1024

MMEMory:NAME <FileName>

Defines the file name when an image of the display is stored to a file rather than printed to a printer using the HCOPy:IMMediate<m>[:DUM] command.

Setting parameters:

<FileName> String parameter specifiying path and file name of the screenshot

Example: See Chapter 17.3.5.1, "Saving a Screenshot to File", on page 1024

Usage: Setting only
SCPI confirmed

HCOPy:DEVice<m>:LANGuage <FileFormat>

Defines the file format for output of the display image to file.

To set the output to file, use HCOPy:DESTination<1..2> with parameter 'MMEM'.

Suffix:

<m> 1..2
Selects the output configuration.

Parameters:

<FileFormat> PNG | JPG | BMP | TIFF | PDF
*RST: PNG

Example: See Chapter 17.3.5.1, "Saving a Screenshot to File", on page 1024

HCOPy:PAGE:ORIentation<1..2> <Orientation>

Defines the page orientation for output of the display image to a printer.

To set the output to printer, use `HCOPy:DESTination<1..2>` with parameter `'SYST:COMM:PRIN'`.

Suffix:

1..2	Selects the output configuration.

Parameters:

<Orientation>	PORTrait | LANDscape *RST: LANDscape

HCOPy:DEVice<m>:COLor <Color>

Selects between color and monochrome printing of the display image.

To set the output to printer, use `HCOPy:DESTination<1..2>` with parameter `'SYST:COMM:PRIN'`.

Suffix:

<m>	1..2 Selects the output configuration.

Parameters:

<Color>	ON | OFF ON: Color output OFF: Black and white output *RST: ON

HCOPy:DEVice<m>:INVerse <InverseColor>

Inverts the colors of the output, i.e. a dark waveform is printed on a white background.

See also:

- `HCOPy:WBKG` on page 1376
- "White background" on page 464

Suffix:

<m>	1..2 Selects the output configuration.

Parameters:

<InverseColor>	ON | OFF *RST: ON

HCOPy:WBKG <WhiteBackground>

Inverts the background color. So you can print waveforms with normal waveform colors on white background.

If both `HCOPy:WBKG` and `HCOPy:DEVice<m>:INVerse` are ON, the instrument inverts the background twice, and it appears black.

See also: "White background" on page 464.

Parameters:
<WhiteBackground> ON | OFF
*RST: OFF

HCOPy:CMAP<m>:DEFault <PrintColorSet>

Defines the default color set for printing of the display image.

To set the output to printer, use `HCOPy:DESTination<1..2>` with parameter `'SYST:COMM:PRIN'`.

Suffix:
<m> 1..2
Selects the output configuration.

Parameters:
<PrintColorSet> DEF1 | DEF4

DEF1
Current screen colors with white background and black grid.

DEF4
Current screen colors without any changes (black background).

*RST: DEF1

HCOPy:SSD <ShowSetupDialog>

Enables or disables the display of open dialog boxes in screenshots. Use this command if you want to document settings in screenshots.

Parameters:
<ShowSetupDialog> ON | OFF
*RST: OFF

Firmware/Software: FW 3.20

HCOPy:ISBA IncludeSignalBar

If OFF, the screenshot shows only the diagram area, without the signal bar.

Parameters:
IncludeSignalBar ON | OFF
*RST: ON

HCOPy:IMMediate<m>[:DUM]

Starts the immediate output of the display image to printer, file, or clipboard, depending on the `HCOPy:DESTination<1..2>` setting.

To get a correct screenshot of the diagrams, results, and dialog boxes, turn on the display using `SYST:DISP:UPD ON`.

Suffix:

<m> 1..2
Selects the output configuration.

Example:

```
SYST:DISP:UPD ON
HCOP:DEST 'MMEM'
MMEM:NAME 'C:\Temp\Print.bmp'
HCOP:IMMediate; *OPC?
```

Example: See Chapter 17.3.5.1, "Saving a Screenshot to File", on page 1024

Usage: Event
Asynchronous command

HCOPy:IMMediate<m>:NEXT

Starts the output of the next display image to printer, file, or clipboard, depending on the `HCOPy:DESTination<1..2>` setting.

If the output is printed to a file, the file name used in the last saving process is automatically counted up to the next unused name.

Suffix:

<m> 1..2
Selects the output configuration.

Example: See Chapter 17.3.5.1, "Saving a Screenshot to File", on page 1024

Usage: Event
Asynchronous command

SYSTem:COMMunicate:PRINter:ENUMerate:FIRSt?

Queries the name of the first printer in the list of printers that is configured in the Windows operating system.

To query the names of other installed printers, use the `SYSTem:COMMunicate:PRINter:ENUMerate[:NEXT]?` command.

Return values:

<PrinterName> If no printer is configured an empty string is returned.

Usage: Query only

SYSTem:COMMunicate:PRINter:ENUMerate[:NEXT]?

Queries the name of the next printer that is configured in the Windows operating system.

Before you send the ...NEXT command, send `SYSTem:COMMunicate:PRINter:ENUMerate:FIRSt?` to return to the beginning of the printer list and query the name of the first printer.

Return values:

<PrinterName> After all available printer names have been returned, an empty string enclosed by quotation marks (") is returned for the next query. Further queries are answered by a query error.

Usage: Query only

SYSTem:COMMunicate:PRINter:SELect<1..2> <PrinterName>

Selects a configured printer.

Parameters:

<PrinterName> Enter the string as it is returned with `SYSTem:COMMunicate:PRINter:ENUMerate:FIRSt?` or `SYSTem:COMMunicate:PRINter:ENUMerate[:NEXT]?`.

17.16.11 Reports

The following commands configure and save report files. To configure the screenshot that is included in the report, use the commands explained in Chapter 17.16.10, "Screenshots", on page 1374.

REPort:LANGuage....1379
REPort:PAPersize....1380
REPort:LOGType....1380
REPort:LOGO....1380
REPort:USER....1380
REPort:COMMent....1380
REPort:FILE:NAME....1381
REPort:FILE:SAVE....1381

REPort:LANGuage <Language>

Sets the language to be used in the report. Available languages are listed in the data sheet.

Parameters:

<Language> String with the english language name, upper case.

Example: `REPort:LANGuage 'Spanish'`

REPort:PAPersize <PaperSize>

Selects the paper size: A4 or US Letter.

Parameters:
<PaperSize> A4 | USL
*RST: A4

Usage: Asynchronous command

REPort:LOGType <Logo>

By default, the Rohde & Schwarz logo is shown in the header of the report pages. You can switch the logo off, or select your logo to be shown.

Parameters:
<Logo> RS | CUST | NONE

CUST
Select the logo file using `REPort:LOGO`.

*RST: RS

Usage: Asynchronous command

REPort:LOGO <LogoFile>

Defines the logo file that is used on the report if `REPort:LOGType` is set to `CUSTom`.

Parameters:
<LogoFile> String with the path and filename of the logo image.

Example: `REPort:LOGO 'C:\Company files\logo.jpg'`

Usage: Asynchronous command

REPort:USER <User>

Enter the user name that appears in the general information section at the beginning of the report.

Parameters:
<User> String parameter

Usage: Asynchronous command

REPort:COMMent <Comment>

Enter a comment that appears in the general information section at the beginning of the report.

Parameters:
<Comment> String parameter

Usage: Asynchronous command

REPort:FILE:NAME <ReportFile>

Sets the file name and path to save the report to.

Parameters:

<ReportFile> String with path and file name. The file extension defines the file format: PDF, HTML, or DOC.

Usage: Asynchronous command

REPort:FILE:SAVE

Saves the report to the specified file.

Usage: Event
Asynchronous command

17.17 Protocols

- Configuration Settings for all Serial Protocols....1381
- Trigger Settings for all Serial Protocols....1386
- I²C (Option R&S RTE-K1)....1388
- SPI (Option R&S RTE-K1)....1420
- UART/RS-232/RS-422/RS-485 (Option R&S RTE-K2)....1440
- CAN (Option R&S RTE-K3/R&S RTE-K9)....1450
- LIN (Option R&S RTE-K3)....1491
- FlexRay (Option R&S RTE-K4)....1519
- Audio Signals (Option R&S RTE-K5)....1553
- MIL-1553 (Option R&S RTE-K6)....1568
- ARINC 429 (Option R&S RTE-K7)....1594
- Ethernet 10BASE-T and 100BASE-TX (Option R&S RTE-K8)....1610
- Ethernet 100BASE-T1 (Option R&S RTE-K57)....1634
- SENT (Option R&S RTE-K10)....1664
- Custom: Manchester / NRZ (Option R&S RTE-K50)....1694
- MDIO (Option R&S RTE-K55)....1738
- USB (Option R&S RTE-K60)....1753
- USBPD (Option R&S RTE-K63)....1794
- Space Wire (Option R&S RTE-K65)....1818
- CXPI (Option R&S RTE-K76)....1835

17.17.1 Configuration Settings for all Serial Protocols

BUS<m>:TYPE....1382
BUS<m>[:STATe]....1382
BUS<m>:SETReflevels....1383
BUS<m>:FAUToset....1383
BUS<m>:LABel....1383
BUS<m>:RESult....1383

BUS<m>:THReshold....1383
BUS<m>:RESDetail....1384
BUS<m>:FORMat....1384
BUS<m>:NEWList....1384
BUS<m>:SYMBols....1385
BUS<m>:ZCOupling....1385
DISPlay:RESultboxes:DEPosition....1385

BUS<m>:TYPE <Type>

Defines the bus or protocol type for analysis.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Type> I2C | SPI | UART | CAN | CANFd | LIN | MILS1553 | MILStd1553 | ARIN429 | ARINc429 | SWIRe | MDIO | HBTO | USB | USBPD | ETHernet | CMSB | FLXRay | I2S | SENT | CXPI

MILS1553 = MILStd1553: specification MIL-STD-1553 (option R&S RTE-K6)
ARIN429 = ARINc429: specification ARINC 429 (option R&S RTE-K7)
HBTO: Ethernet 100BASE-T1 (BroadR-Reach, option R&S RTE-K57)
CXPI: Clock extension peripheral interface (option R&S RTE-K76)
CMSB: custom decode serial bus (option R&S RTE-K50)
SWIRe: SpaceWire (option R&S RTE-K65)

*RST: I2C

Usage: Asynchronous command

BUS<m>[:STATe] <State>

Enables the decoding of the specified bus.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

BUS<m>:SETReflevels

Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Suffix:

<m> 1..4
Selects the serial bus.

Usage: Event
Asynchronous command

BUS<m>:FAUToset

Starts software algorithms for determining the signal threshold levels and bitrate.

Suffix:

<m> 1..4

Usage: Event
Asynchronous command

BUS<m>:LABel <Label>

Defines a label to be displayed with the bus.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Label> String containing the label text.

Usage: Asynchronous command

BUS<m>:RESult <ShowResultTable>

Opens a table with decoded data of the serial signal. The function affects all protocol types and requires the option for the analyzed protocol.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ShowResultTable> ON | OFF
*RST: ON

Usage: Asynchronous command

BUS<m>:THReshold <ShwThresLines>

If ON, the threshold levels are displayed in the diagram.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ShwThresLines> ON | OFF
*RST: OFF

Usage: Asynchronous command

BUS<m>:RESDetail <ShwResDetails>

Shows detailed information for the frames.

The command is relevant for FlexRay, D-PHY, M-PHY, Ethernet, CXPI, and Custom protocols.

Suffix:
<m> 1..4

Parameters:
<ShwResDetails> ON | OFF
*RST: OFF

Usage: Asynchronous command

Firmware/Software: FW 3.40

BUS<m>:FORMat <DataFormat>

Sets the number format for decoded data values of the indicated serial bus. It defines the format in the decode table, and in the combs of the decoded signal on the screen.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataFormat> HEX | OCT | BIN | ASCII | ASCii | SIGN | USIG
ASCII = ASCii
USIG = unsigned
*RST: HEX

BUS<m>:NEWList <FileName>

Loads a label list file.

Suffix:
<m> 1..4
Selects the serial bus.

Setting parameters:
<FileName> String parameter with path and file name.

Example:
```
BUS1:NEWList 'C:\Protocols\CAN.csv'
BUS1:SYMBols ON
```

Usage: Setting only

BUS<m>:SYMBols <UseTranslation>

Activates the lable list to be used for decoding.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<UseTranslation> ON | OFF
*RST: OFF

Usage: Asynchronous command

BUS<m>:ZCOupling <ZoomCoupling>

If enabled, the protocol decode zoom and result table are synchronized.

Suffix:
<m> 1..4

Parameters:
<ZoomCoupling> ON | OFF
*RST: OFF

DISPlay:RESultboxes:DEPosition <DecodePosition>

Defines the position of the protocol decode result table on the screen.

Parameters:
<DecodePosition> PREV | FLOA | DOCK

PREV
Preview: result icon on the sidebar.

FLOA
Floating result box in front of the diagrams.

DOCK
Docked: fixed tab below the diagrams.

*RST: FLOA

Usage: Asynchronous command

17.17.2 Trigger Settings for all Serial Protocols

The following commands are available for all serial protocols that have a protocol trigger.

TRIGger<m>:SOURce[:SELect]....................1386
TRIGger<m>:SOURce:SBSelect.................... 1387
BUS<m>:TYPE....................1387

TRIGger<m>:SOURce[:SELect] <SourceDetailed>

Selects the source of the trigger signal.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
Available values depend on the selected trigger source. For input channels CHAN1...4, a trigger sequence can be configured.
For all other trigger sources, only suffix 1 is allowed.
See also: `TRIGger<m>:SEQuence:MODE`

Parameters:

<SourceDetailed> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4 | EXTernanalog | SBUS | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | LOGIC | MSOB1 | MSOB2 | MSOB3 | MSOB4 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | LINE

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channels

EXTernanalog
External analog signal connected to the External Trigger Input. For this source, only the analog edge trigger is available.

LINE
The instrument generates the trigger from the AC power input and synchronizes the signal to the AC power frequency.

SBUS
Serial bus

D0...D15
Digital channels (option R&S RTE-B1)
See also: Chapter 17.18.4, "Trigger Settings for Digital Signals and Parallel Buses", on page 1874

LOGIc
Logic combination of digital channels, used as trigger source (option R&S RTE-B1)

MSOB1 | MSOB2 | MSOB3 | MSOB4
Parallel bus (option R&S RTE-B1)

Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4
Input channels of R&S RT-ZVCmulti-channel power probe. Only available in the A-trigger with trigger type EDGE.

DIFF1 | DIFF2 | COMMON1 | COMMON2
Differential signals

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:SOURce:SBSelect <SerialBus>

Selects the serial bus to be triggered on.

Suffix:
<m> 1..3
Event in a trigger sequence: 1 = A-event only

Parameters:
<SerialBus> SBUS1 | SBUS2 | SBUS3 | SBUS4

Usage: Asynchronous command

Firmware/Software: Version 2.70

BUS<m>:TYPE <Type>

Defines the bus or protocol type for analysis.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Type> I2C | SPI | UART | CAN | CANFd | LIN | MILS1553 | MILStd1553 | ARIN429 | ARINc429 | SWIRe | MDIO | HBTO | USB | USBPD | ETHernet | CMSB | FLXRay | I2S | SENT | CXPI

MILS1553 = MILStd1553: specification MIL-STD-1553 (option R&S RTE-K6)
ARIN429 = ARINc429: specification ARINC 429 (option R&S RTE-K7)
HBTO: Ethernet 100BASE-T1 (BroadR-Reach, option R&S RTE-K57)
CXPI: Clock extension peripheral interface (option R&S RTE-K76)
CMSB: custom decode serial bus (option R&S RTE-K50)
SWIRe: SpaceWire (option R&S RTE-K65)

*RST: I2C

Usage: Asynchronous command

17.17.3 I²C (Option R&S RTE-K1)

- Configuration....1388
- Trigger....1390
- Decode Results....1396
- I²C Search Settings....1405
- I²C Search Results....1413

17.17.3.1 Configuration

BUS<m>:I2C:SCL:SOURce....1388
BUS<m>:I2C:SDA:SOURce....1388
BUS<m>:I2C:SCL:THReshold....1389
BUS<m>:I2C:SDA:THReshold....1389
BUS<m>:I2C:TECHnology....1389
BUS<m>:I2C:RWBit....1390

BUS<m>:I2C:SCL:SOURce <SCLSource>

Sets the waveform of the clock line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SCLSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C2W1

Usage: Asynchronous command

BUS<m>:I2C:SDA:SOURce <SDASource>

Sets the waveform of the data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:
<SDASource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

BUS<m>:I2C:SCL:THReshold <SCLThreshold>

Sets a user-defined threshold value for the clock line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:I2C:TECHnology`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SCLThreshold> User-defined clock threshold

Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:I2C:SDA:THReshold <SDAThreshold>

Sets a user-defined threshold value for the data line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:I2C:TECHnology`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SDAThreshold> User-defined data threshold

Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:I2C:TECHnology <Technology>

Sets the threshold voltage clock and data lines as defined for various signal technologies.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15 | V25 | V165 | V125 | V09 | V38 | V20 | V0
1.5 V, 2.5 V, 1.65 V ... respectively

VM13
-1.3 V (negative value)

MANual
Manual setting of user-defined values with `BUS<m>:I2C:SCL:THReshold` and `BUS<m>:I2C:SDA:THReshold`.

*RST: V165

BUS<m>:I2C:RWBit <BusConfig>

Defines if the R/W bit of a 7-bit address is considered separately or as part of the address. 10-bit addresses are not affected. The setting defines which address lengths are available with `TRIGger<m>:I2C:AMODe`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<BusConfig> SEParate | INADdress

SEParate
7-bit address and separate R/W bit.

INADdress
8-bit address with R/W bit included.

*RST: SEParate

17.17.3.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:I2C:MODE....1391
TRIGger<m>:I2C:ACCess....1391
TRIGger<m>:I2C:ADNack....1392
TRIGger<m>:I2C:DWNack....1392
TRIGger<m>:I2C:DRNack....1392

TRIGger<m>:I2C:AMODe....1392
TRIGger<m>:I2C:ACONdition....1393
TRIGger<m>:I2C:ADDRess....1393
TRIGger<m>:I2C:ADDTo....1393
TRIGger<m>:I2C:ADOR<n>:ENABle....1393
TRIGger<m>:I2C:ADOR<n>:ADRType....1394
TRIGger<m>:I2C:ADOR<n>[:VALue]....1394
TRIGger<m>:I2C:ADOR<n>:RWBit....1394
TRIGger<m>:I2C:DPOPerator....1395
TRIGger<m>:I2C:DPOSition....1395
TRIGger<m>:I2C:DPTO....1395
TRIGger<m>:I2C:DCONdition....1395
TRIGger<m>:I2C:DMIN....1396
TRIGger<m>:I2C:DMAX....1396

TRIGger<m>:I2C:MODE <Type>

Selects the trigger type for I²C analysis.

See: "Trigger type" on page 488

Parameters:

<Type> STARt | REPStart | STOP | NACK | ADDRess | ADOR | ADAT

STARt
Start condition

REPStart
Repeated start - the start condition occurs without previous stop condition.

STOP
Stop condition, end of frame

NACK
Missing acknowledge bit. To localize specific missing acknowledge bits, use `TRIGger<m>:I2C:ADNack`, `TRIGger<m>:I2C:DWNack`, and `TRIGger<m>:I2C:DRNack`.

ADDRess
Triggers on one specific address

ADOR
Triggers on an OR combination with up to four four address conditions.

ADAT
Triggers on a combination of address and data condition.

*RST: STARt

Usage: Asynchronous command

TRIGger<m>:I2C:ACCess <RWBitAddress>

Sets the trigger condition for the R/W bit - the transfer direction of the data.

Parameters:

<RWBitAddress> READ | WRITe | EITHer

EITHer
Transfer direction is not relevant.

*RST: EITHer

Usage: Asynchronous command

TRIGger<m>:I2C:ADNack <AddressNack>

Triggers if the address acknowledge bit is missing - no slave recognizes the address.

Parameters:

<AddressNack> ON | OFF

*RST: ON

Usage: Asynchronous command

TRIGger<m>:I2C:DWNack <DataWriteNack>

Triggers if a date acknowledge bit is missing - the addressed slave does not accept the data.

Parameters:

<DataWriteNack> ON | OFF

*RST: ON

Usage: Asynchronous command

TRIGger<m>:I2C:DRNack <DataReadNack>

Triggers on the end of the read process when the master reads data from the slave. This Nack is sent according to the protocol definition, it is not an error.

Parameters:

<DataReadNack> ON | OFF

*RST: ON

Usage: Asynchronous command

TRIGger<m>:I2C:AMODe <AddressType>

Sets the address length. The setting affects the address input with `TRIGger<m>:I2C:ADDRess` and `TRIGger<m>:I2C:ADDTo`.

Parameters:

<AddressType> BIT7 | BIT7_RW | BIT10 | ANY

BIT7
Enter the 7 address bits. Only available if `BUS<m>:I2C:RWBit SEParate` is set.

BIT7_RW
Enter 7 address bits and the R/W bit. Only available if `BUS<m>:I2C:RWBitINADdress` is set.

BIT10
10-bit address

ANY
Only available for trigger type "Address + data" (`TRIGger<m>:I2C:MODE ADAT`). Used to trigger on data only, regardless of the address.

*RST: BIT7

Usage: Asynchronous command

TRIGger<m>:I2C:ACONdition <AddressOperator>

Sets the operator to set a specific address or an address range. The address values are set with `TRIGger<m>:I2C:ADDRess` and `TRIGger<m>:I2C:ADDTo`.

Parameters:

<AddressOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

*RST: EQUal

TRIGger<m>:I2C:ADDRess <Address>

Triggers on the specified slave address, or sets the the start value of an address range depending on the condition set with `TRIGger<m>:I2C:ACONdition`.

Parameters:

<Address> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:I2C:ADDTo <AddressTo>

Sets the the end value of an address range if the condition is set to an address range with `TRIGger<m>:I2C:ACONdition`.

Parameters:

<AddressTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Usage: Asynchronous command

TRIGger<m>:I2C:ADOR<n>:ENABle <UseAddress>

Includes the indicated ADOR address in the "address OR" trigger condition.

Suffix:

<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<UseAddress> ON | OFF
*RST: OFF

TRIGger<m>:I2C:ADOR<n>:ADRType <AddressType>

Sets the address type for the indicated ADOR address in the "address OR" trigger condition.

Suffix:
<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<AddressType> BIT7 | BIT7_RW | BIT10
*RST: BIT7

TRIGger<m>:I2C:ADOR<n>[:VALue] <Address>

Defines the address pattern of the indicated ADOR address in the "address OR" trigger condition.

Suffix:
<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<Address> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:I2C:ADOR<n>:RWBit <RWBit>

Defines the R/W bit of the indicated ADOR address in the "address OR" trigger condition.

Suffix:
<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<RWBit> UNDefined | READ | WRITe | EITHer
UNDefined
Return value only
*RST: EITHer

TRIGger<m>:I2C:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The position of the required pattern is not relevant.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:I2C:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `TRIGger<m>:I2C:DPOSition` and `TRIGger<m>:I2C:DPTO`.

*RST: ANY

TRIGger<m>:I2C:DPOSition <DataPosition>

Sets the number of data bytes before the first byte of interest. These bytes are ignored.

Parameters:

<DataPosition> The index 0 is associated with the first data byte.

Range: 0 to 4095
Increment: 1
*RST: 0

TRIGger<m>:I2C:DPTO <DataPositionTo>

Defines the last byte of interest, if `TRIGger<m>:I2C:DPOPerator` is set to `RANGe`.

Parameters:

<DataPositionTo> Range: 0 to 4095
Increment: 1
*RST: 0

TRIGger<m>:I2C:DCONdition <DataOperator>

Sets the operator to set a specific data value or a data range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

*RST: EQUal

TRIGger<m>:I2C:DMIN <Data>

Specifies the data bit pattern, or sets the the start value of a data pattern range.. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Parameters:

<Data> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

TRIGger<m>:I2C:DMAX <DataTo>

Sets the the end value of an data range if `TRIGger<m>:I2C:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

17.17.3.3 Decode Results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1384
- `BUS<m>:SYMBols` on page 1385

BUS<m>:I2C:FRAMe<n>:DATA? 1397
BUS<m>:I2C:FCOunt? 1397
BUS<m>:I2C:FRAMe<n>:AACCess? 1397
BUS<m>:I2C:FRAMe<n>:ACCess? 1398
BUS<m>:I2C:FRAMe<n>:ACOMplete? 1398
BUS<m>:I2C:FRAMe<n>:ADBStart? 1398
BUS<m>:I2C:FRAMe<n>:ADDRess? 1399
BUS<m>:I2C:FRAMe<n>:ADEVice? 1399
BUS<m>:I2C:FRAMe<n>:AMODe? 1399
BUS<m>:I2C:FRAMe<n>:ASTart? 1400
BUS<m>:I2C:FRAMe<n>:BITRate? 1400
BUS<m>:I2C:FRAMe<n>:RWBStart? 1400
BUS<m>:I2C:FRAMe<n>:STATus? 1401
BUS<m>:I2C:FRAMe<n>:STARt? 1401
BUS<m>:I2C:FRAMe<n>:STOP? 1402
BUS<m>:I2C:FRAMe<n>:SYMBol? 1402
BUS<m>:I2C:FRAMe<n>:BCOunt? 1402
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACCess? 1403
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACKStart? 1403

BUS<m>:I2C:FRAMe<n>:BYTE<o>:COMPlete?.. 1403
BUS<m>:I2C:FRAMe<n>:BYTE<o>:STARt?.. 1404
BUS<m>:I2C:FRAMe<n>:BYTE<o>:VALue?.. 1404

BUS<m>:I2C:FRAMe<n>:DATA?

Returns the data words of the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<Data>	Comma-separated list of integer values (N, D1, D2,…, DN). N is the number of bytes in the frame, and D1…DN are the values of the bytes.

Example:

```
BUS:I2C:FRAMe4:DATA?
<-- 3,74,164,18
```

Usage: Query only

BUS<m>:I2C:FCOunt?

Returns the number of decoded frames.

Suffix:

<m>	1..4 Selects the serial bus.

Return values:

<Count>	Total number of decoded frames.

Usage: Query only

BUS<m>:I2C:FRAMe<n>:AACCess?

Returns the address acknowledge bit value for the indicated frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<AddressAckBit>	INComplete | ACK | NACK | EITHer *RST: INComplete

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ACCess?

Returns the value of the R/W bit of the indicated frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<RWBit>	UNDefined | READ | WRITe | EITHer *RST: UNDefined

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ACOMplete?

Returns if the address is completely contained in the acquisition.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<AddressComplete>	ON | OFF *RST: OFF

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ADBStart?

Returns the start time of the address acknowledge bit.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<AddrAckBtStrt>	Range: -100E+24 to 100E+24 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ADDRess?

Returns the device address value of the indicated frame, that is, the address value that is shown in the decoded cells and in the decode results table.

If the frame has a 7-bit address, the command considers the status of `BUS<m>:I2C:RWBit`. If `BUS<m>:I2C:RWBit INADdress` is set, the returned address includes the R/W bit (8 bit). Otherwise, the pure address without the R/W bit is returned (7 bit, same result as returned with `BUS<m>:I2C:FRAMe<n>:ADEVice?`).

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddressValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and default – are decimal values.

Range: 0 to 1023
*RST: 0

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ADEVice?

Returns the pure device address of the indicated frame *without* the R/W bit.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<DeviceAddress> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and default – are decimal values.

Range: 0 to 1023
*RST: 0

Usage: Query only

BUS<m>:I2C:FRAMe<n>:AMODe?

Returns the address length.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<AddressType> BIT7 | BIT7_RW | BIT10 | AUTO | ANY
*RST: BIT7

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ASTart?

Returns the start time of the address for the indicated frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<AddressStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:I2C:FRAMe<n>:RWBStart?

Returns the start time of the R/W bit

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<RWBitStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:STATus?

Returns the overall state of the frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<FrameState> INComplete | OK | UNEXpstop | INSufficient | ADDifferent

INComplete
The stop bit is missing.

OK
The frame is valid.

UNEXpstop
A stop bit was detected but clock and data are continued.

INSufficient
The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

ADDifferent
Error in 10 bit address. In case of a read access on a 10 bit address, the first address byte is sent twice, first as write, the second as read. The first seven bits of the byte must be identical. If they are not identical, the ADDiffernt error is indicated.

*RST: OK

Usage: Query only

BUS<m>:I2C:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStart>	Range: -100E+24 to 100E+24 *RST: 0 Default unit: s
Usage:	Query only

BUS<m>:I2C:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<FrameStop>	Range: -100E+24 to 100E+24 *RST: 0 Default unit: s
Usage:	Query only

BUS<m>:I2C:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Translation>	String with symbolic name of the address
Usage:	Query only

BUS<m>:I2C:FRAMe<n>:BCOunt?

Returns the number of bytes in the specified frame

Suffix:

<m>	1..4 Selects the input channel.
<n>	* Selects the frame.

Return values:

<Count>	Byte count

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACCess?

Returns the acknowledge bit value of the specified data byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<AckBit> INComplete | ACK | NACK | EITHer
*RST: INComplete

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACKStart?

Returns the start time of the acknowledge bit of the specified byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<AckBitStart> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.
Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:COMPlete?

Returns if the indicated byte is completely contained in the acquisition.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<ValueComplete> ON | OFF
*RST: OFF

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:STARt?

Returns the start time of the specified data byte.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<Start> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:VALue?

Returns the data value of the specified byte.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<Value> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and default – are decimal values.
Range: 0 to 255
*RST: 0

Usage: Query only

17.17.3.4 I²C Search Settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to I²C trigger commands.

SEARch:TRIGger:I2C:SCONdition....1405
SEARch:TRIGger:I2C:RCONdition....1405
SEARch:TRIGger:I2C:STCNdition....1406
SEARch:TRIGger:I2C:NACKnowledge....1406
SEARch:TRIGger:I2C:SADDress....1406
SEARch:TRIGger:I2C:ADOR....1407
SEARch:TRIGger:I2C:ADData....1407
SEARch:TRIGger:I2C:ACONdition....1407
SEARch:TRIGger:I2C:AMODe....1408
SEARch:TRIGger:I2C:ADDRess....1408
SEARch:TRIGger:I2C:ADDTo....1408
SEARch:TRIGger:I2C:ACCess....1409
SEARch:TRIGger:I2C:ADDO<m>:ENABle....1409
SEARch:TRIGger:I2C:ADDO<m>:ADRType....1409
SEARch:TRIGger:I2C:ADDO<m>[:VALue]....1410
SEARch:TRIGger:I2C:ADDO<m>:RWBit....1410
SEARch:TRIGger:I2C:DPOPerator....1411
SEARch:TRIGger:I2C:DPOSition....1411
SEARch:TRIGger:I2C:DPTO....1411
SEARch:TRIGger:I2C:DCONdition....1412
SEARch:TRIGger:I2C:DMIN....1412
SEARch:TRIGger:I2C:DMAX....1412
SEARch:TRIGger:I2C:ADNack....1412
SEARch:TRIGger:I2C:DRNack....1413
SEARch:TRIGger:I2C:DWNack....1413

SEARch:TRIGger:I2C:SCONdition <SearchName>,<Start>
SEARch:TRIGger:I2C:SCONdition? <SearchName>

Enables the search for the start of the message.

Parameters:

<Start>	ON | OFF
	*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:RCONdition <SearchName>,<RepeatedStart>
SEARch:TRIGger:I2C:RCONdition? <SearchName>

Enables the search for a start condition without previous stop condition.

Parameters:

<RepeatedStart>	ON | OFF *RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:STCNdition <SearchName>,<Stop>
SEARch:TRIGger:I2C:STCNdition? <SearchName>

Enables the search for the start of the message.

Parameters:

<Stop>	ON | OFF *RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:NACKnowledge <SearchName>,<NoAcknowledge>
SEARch:TRIGger:I2C:NACKnowledge? <SearchName>

Searches for missing address acknowledge bits.

Parameters:

<NoAcknowledge>	ON | OFF *RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:SADDress <SearchName>,<Address>
SEARch:TRIGger:I2C:SADDress? <SearchName>

Enables the search for one specific address condition or for a combination of address conditions.

To define the address condition, use the following commands:

- `SEARch:TRIGger:I2C:ACONdition` on page 1407
- `SEARch:TRIGger:I2C:ADDRess` on page 1408
- `SEARch:TRIGger:I2C:ADDTo` on page 1408
- `SEARch:TRIGger:I2C:AMODe` on page 1408
- `SEARch:TRIGger:I2C:ACCess` on page 1409

Parameters:

<Address>	ON | OFF *RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADOR <SearchName>,<AddressOr>
SEARch:TRIGger:I2C:ADOR? <SearchName>

Enables the search for one to four address conditions.

- `SEARch:TRIGger:I2C:ADDO<m>:ENABle` on page 1409
- `SEARch:TRIGger:I2C:ADDO<m>:ADRType` on page 1409
- `SEARch:TRIGger:I2C:ADDO<m>[:VALue]` on page 1410
- `SEARch:TRIGger:I2C:ADDO<m>:RWBit` on page 1410

Parameters:

<AddressOr> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:I2C:ADDData <SearchName>,<AddressData>
SEARch:TRIGger:I2C:ADDData? <SearchName>

Enables the search for a combination of address and data conditions.

Parameters:

<AddressData> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:I2C:ACONdition <SearchName>,<AddressOperator>
SEARch:TRIGger:I2C:ACONdition? <SearchName>

Sets the operator to set a specific address or an address range.

Parameters:

<AddressOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `SEARch:TRIGger:I2C:ADDRess` on page 1408.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:I2C:ADDRess` on page 1408 and `SEARch:TRIGger:I2C:ADDTo` on page 1408.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:AMODe <SearchName>,<AddressType>
SEARch:TRIGger:I2C:AMODe? <SearchName>

Sets the address length.

Parameters:
<AddressType> BIT7 | BIT7_RW | BIT10 | ANY

BIT7 | BIT10
Enter only address bits in the address pattern.

BIT7_RW
Enter seven address bits and also the R/W bit in the address pattern.

ANY
Only available for search criteria "Address and data" (`SEARch:TRIGger:I2C:ADData` is set `ON`). Used to search for data only, regardless of the address.

*RST: BIT7

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ADDRess <SearchName>,<Address>
SEARch:TRIGger:I2C:ADDRess? <SearchName>

Specifies an address pattern, or sets the the start value of an address range.

Parameters:
<Address> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.
The pattern length is defined with `SEARch:TRIGger:I2C:AMODe`.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDTo <SearchName>,<AddressTo>
SEARch:TRIGger:I2C:ADDTo? <SearchName>

Sets the the end value of an address range if `SEARch:TRIGger:I2C:ACONdition` is set to `INRange` or `OORange`.

Parameters:
<AddressTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.
The pattern length is defined with `SEARch:TRIGger:I2C:AMODe`.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ACCess <SearchName>,<RWBitAddress>
SEARch:TRIGger:I2C:ACCess? <SearchName>

Sets the transfer direction of the data.

Parameters:
<RWBitAddress> READ | WRITe | EITHer
*RST: EITHer

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ADDO<m>:ENABle <SearchName>,<UseAddress>
SEARch:TRIGger:I2C:ADDO<m>:ENABle? <SearchName>

Includes the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<UseAddress> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDO<m>:ADRType <SearchName>,<AddressType>
SEARch:TRIGger:I2C:ADDO<m>:ADRType? <SearchName>

Sets the address type for the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<AddressType> BIT7 | BIT7_RW | BIT10

BIT7 | BIT10
Enter only address bits in the address pattern.

BIT7_RW
Enter seven address bits and also the R/W bit in the address pattern.

*RST: BIT7

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDO<m>[:VALue] <SearchName>,<Address>
SEARch:TRIGger:I2C:ADDO<m>[:VALue]? <SearchName>

Defines the address pattern of the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<Address> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.
The pattern length is defined with `SEARch:TRIGger:I2C:ADDO<m>:ADRType`.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDO<m>:RWBit <SearchName>,<RWBit>
SEARch:TRIGger:I2C:ADDO<m>:RWBit? <SearchName>

Defines the R/W bit of the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<RWBit> UNDefined | READ | WRITe | EITHer

UNDefined
Only return value

*RST: EITHer

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:I2C:DPOPerator? <SearchName>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The position of the required pattern is not relevant.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `SEARch:TRIGger:I2C:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `SEARch:TRIGger:I2C:DPOSition` and `SEARch:TRIGger:I2C:DPTO`.

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:I2C:DPOSition? <SearchName>

Defines the first byte of interest. All bytes before that byte are ignored.

Parameters:

<DataPosition> The index 0 is associated with the first data byte.

Range: 0 to 4095
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:I2C:DPTO? <SearchName>

Defines the last byte of interest, if `SEARch:TRIGger:I2C:DPOPerator` defines a range.

Parameters:

<DataPositionTo> Range: 0 to 4095
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:I2C:DCONdition? <SearchName>

Sets the operator to set a specific data value or a data range.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange
*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DMIN <SearchName>,<Data>
SEARch:TRIGger:I2C:DMIN? <SearchName>

Specifies the data bit pattern, or sets the the start value of a data pattern range.. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Parameters:
<Data> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:DMAX <SearchName>,<DataTo>
SEARch:TRIGger:I2C:DMAX? <SearchName>

Sets the the end value of an address range if `SEARch:TRIGger:I2C:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ADNack <SearchName>,<AddressNack>
SEARch:TRIGger:I2C:ADNack? <SearchName>

Parameters:
<AddressNack> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:DRNack <SearchName>,<DataReadNack>
SEARch:TRIGger:I2C:DRNack? <SearchName>

Searches for the end of the read process when the master reads data from the slave. This Nack is sent according to the protocol definition, it is not an error.

Parameters:
<DataReadNack> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:DWNack <SearchName>,<DataWriteNack>
SEARch:TRIGger:I2C:DWNack? <SearchName>

Searches for missing data write acknowledge bits.

Parameters:
<DataWriteNack> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

17.17.3.5 I²C Search Results

The search on decoded CAN data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.3.3, "Decode Results", on page 1396.

SEARch:RESult:I2C:FCOunt?......1414
SEARch:RESult:I2C:FRAMe<m>:STATus?......1414
SEARch:RESult:I2C:FRAMe<m>:STARt?......1414
SEARch:RESult:I2C:FRAMe<m>:STOP?......1415
SEARch:RESult:I2C:FRAMe<m>:AACCess?......1415
SEARch:RESult:I2C:FRAMe<m>:ACCess?......1415
SEARch:RESult:I2C:FRAMe<m>:ACOMplete?......1416
SEARch:RESult:I2C:FRAMe<m>:ADBStart?......1416

SEARch:RESult:I2C:FRAMe<m>:ADDRess?....1416
SEARch:RESult:I2C:FRAMe<m>:ADEVice?....1416
SEARch:RESult:I2C:FRAMe<m>:AMODe?....1417
SEARch:RESult:I2C:FRAMe<m>:ASTart?....1417
SEARch:RESult:I2C:FRAMe<m>:DATA?....1417
SEARch:RESult:I2C:FRAMe<m>:RWBStart?....1418
SEARch:RESult:I2C:FRAMe<m>:SYMBol?....1418
SEARch:RESult:I2C:FRAMe<m>:BCOunt?....1418
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess?....1418
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart?....1419
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete?....1419
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt?....1419
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue?....1420

SEARch:RESult:I2C:FCOunt? <SearchName>

Query parameters:

<SearchName>

Return values:

<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:STATus? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<State> INComplete | OK | UNEXpstop | INSufficient | ADDifferent
*RST: OK

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:STARt? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:STOP? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:AACCess? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<AddressAckBit> INComplete | ACK | NACK | EITHer
*RST: INComplete

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ACCess? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<RWBit> UNDefined | READ | WRITe | EITHer
*RST: UNDefined

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ACOMplete? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddressComplete> ON | OFF
*RST: OFF

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ADBStart? <SearchName>

Returns the start time of the address acknowledge bit.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddrAckBtStrt> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ADDRess? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddressValue> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ADEVice? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<DeviceAddress>	Range:	0 to 1023
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:AMODe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddressType> BIT7 | BIT7_RW | BIT10 | AUTO | ANY

*RST: BIT7

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ASTart? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<AddressStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:DATA? <SearchName>

Returns the data bytes of the indicated frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:RWBStart? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<RWBitStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BCOunt?

Suffix:
<m> *

Return values:
<Count>

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<ByteAckBit> INComplete | ACK | NACK | EITHer
*RST: INComplete

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart? <SearchName>

Returns the start time of the acknowledge bit of the indicated data byte.

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteAckBitStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteComplete>	ON | OFF	
	*RST:	OFF

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:

<SearchName>

Return values:

<Value>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

17.17.4 SPI (Option R&S RTE-K1)

- SPI Bus Configuration....................1420
- SPI Trigger....................1425
- SPI Decode Results....................1428
- SPI Search Settings....................1433
- SPI Search Results....................1436

17.17.4.1 SPI Bus Configuration

BUS<m>:SPI:BORDer....................1420
BUS<m>:SPI:WSIZe....................1421
BUS<m>:SPI:SCLK:SOURce....................1421
BUS<m>:SPI:SSELect:SOURce....................1421
BUS<m>:SPI:SSELect:POLarity....................1422
BUS<m>:SPI:MISO:SOURce....................1422
BUS<m>:SPI:MISO:POLarity....................1422
BUS<m>:SPI:MOSI:SOURce....................1423
BUS<m>:SPI:MOSI:POLarity....................1423
BUS<m>:SPI:TECHnology....................1423
BUS<m>:SPI:SCLK:THReshold....................1424
BUS<m>:SPI:MISO:THReshold....................1424
BUS<m>:SPI:MOSI:THReshold....................1424
BUS<m>:SPI:SSELect:THReshold....................1424
BUS<m>:SPI:FRCondition....................1424
BUS<m>:SPI:TIMeout....................1425

BUS<m>:SPI:BORDer <BitOrder>

Defines if the data of the messages starts with msb (most significant bit) or lsb (least significant bit).

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<BitOrder> LSBF | MSBF

*RST: MSBF

Usage: Asynchronous command

BUS<m>:SPI:WSIZe <WordLength>

Sets the number of bits in a message.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<WordLength> Number of bits

Range: 4 to 32
Increment: 1
*RST: 8

Usage: Asynchronous command

BUS<m>:SPI:SCLK:SOURce <SCLKSource>

Sets the input channel of the clock line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SCLKSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C2W1

Usage: Asynchronous command

BUS<m>:SPI:SSELect:SOURce <SlaveSelectSource>

Sets the input channel of the slave select line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SlaveSelectSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: None

Usage: Asynchronous command

BUS<m>:SPI:SSELect:POLarity <SSPolarity>

Selects whether transmitted slave select signal is high active (high = 1) or low active (low = 1).

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SSPolarity> ACTLow | ACTHigh

*RST: ACTLow

Usage: Asynchronous command

BUS<m>:SPI:MISO:SOURce <MISOSource>

Sets the input channel of the MISO line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<MISOSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: None

Usage: Asynchronous command

BUS<m>:SPI:MISO:POLarity <MISOPolarity>

Selects whether transmitted data is high active (high = 1) or low active (low = 1).

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<MISOPolarity> ACTLow | ACTHigh

*RST: ACTHigh

Usage: Asynchronous command

BUS<m>:SPI:MOSI:SOURce <MOSISource>

Sets the input channel of the MOSI line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<MOSISource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

BUS<m>:SPI:MOSI:POLarity <MOSIPolarity>

Selects whether transmitted data is high active (high = 1) or low active (low = 1).

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<MOSIPolarity> ACTLow | ACTHigh

*RST: ACTHigh

Usage: Asynchronous command

BUS<m>:SPI:TECHnology <Technology>

Sets the threshold voltage clock, slave select and data lines as defined for various signal technologies.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15 | V25 | V165 | V125 | V09 | V38 | V20 | V0
1.5 V, 2.5 V, 1.65 V ... respectively

VM13
-1.3 V (negative value)

MANual
Manual setting of user-defined values with
`BUS<m>:SPI:SCLK|SSEL|MISO|MOSI:THReshold`.

*RST: V165

BUS<m>:SPI:SCLK:THReshold <SCLKThreshold>
BUS<m>:SPI:MISO:THReshold <MISOThreshold>
BUS<m>:SPI:MOSI:THReshold <MOSIThreshold>
BUS<m>:SPI:SSELect:THReshold <SSThreshold>

Set user-defined threshold values for the clock, MISO, MOSI and slave select lines.

Alternatively, you can set the thresholds according to the signal technology with `BUS<m>:SPI:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SSThreshold> User-defined value

Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:SPI:FRCondition <FrameCondition>

Defines the start of a frame. A frame contains a number of successive words, at least one word.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<FrameCondition> SS | CLKTimeout

SS
Start and end of the frame is defined by the active state of the slave select signal, see `BUS<m>:SPI:SSELect:POLarity`.

CLKTimeout
Defines a timeout on the clock line SCLK as limiter between two frames. The timeout condition is used for SPI connections without an SS line.

*RST: SS

BUS<m>:SPI:TIMeout <ClockTimeout>

Defines a timeout on the clock line SCLK as limiter between two frames. The timeout condition is used for SPI connections without an SS line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ClockTimeout> Range: 50E-9 to 10
Increment: 1E-6
*RST: 1E-3
Default unit: s

17.17.4.2 SPI Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:SPI:MODE....1425
TRIGger<m>:SPI:PALignment....1426
TRIGger<m>:SPI:DPOPerator....1426
TRIGger<m>:SPI:DPOSition....1427
TRIGger<m>:SPI:DPTO....1427
TRIGger<m>:SPI:FCONdition....1427
TRIGger<m>:SPI:MISopattern....1427
TRIGger<m>:SPI:MOSipattern....1427

TRIGger<m>:SPI:MODE <Type>

Selects the trigger type for SPI analysis.

Parameters:

<Type> SSACtive | TIMeout | MOSI | MISO | MOMI

SSACtive
Start of the message: slave select signal SS changes to the active state.

TIMeout
Triggers on the next message start after the "Timeout" time.

MOSI
Triggers on a specified data pattern in that is expected on the MOSI line. Define the pattern with `TRIGger<m>:SPI:MOSipattern`.

MISO
Triggers on a specified data pattern in that is expected on the MISO line. Define the pattern with `TRIGger<m>:SPI:MISopattern`

MOMI
Triggers on a specified data patterns on the MISO and MISO lines.

*RST: SSACtive

Usage: Asynchronous command

TRIGger<m>:SPI:PALignment <DataAlignment>

Defines how the specified data pattern is searched.

Parameters:

<DataAlignment> WORD | BIT

WORD
The pattern is matched only at word boundaries.

BIT
Bit-by bit: the pattern can be at any position in the data word.

*RST: WORD

Usage: Asynchronous command

TRIGger<m>:SPI:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The position of the required pattern is not relevant.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:SPI:DPOSition`.

INRange = RANGe
Set the minimum and maximum value of the range with `TRIGger<m>:SPI:DPOSition` and `TRIGger<m>:SPI:DPTO`.

*RST: ANY

TRIGger<m>:SPI:DPOSition <DataPosition>

Sets the number of bits or words to be ignored before the first bit or word od interest. The effect is defined by `TRIGger<m>:SPI:PALignment`.

Parameters:

<DataPosition> The index 0 is associated with the first data byte.

Range: 0 to 4095 for triggering on one line (MISO or MOSI), 2047 for triggering on both lines.
Increment: 1
*RST: 0

TRIGger<m>:SPI:DPTO <DataPositionTo>

Defines the last bit or word of interest, if `TRIGger<m>:SPI:DPOPerator` is set to `INRange`.

Parameters:

<DataPositionTo> Range: 1 to 4095 for triggering on one line (MISO or MOSI), 2047 for triggering on both lines.
Increment: 1
*RST: 1

TRIGger<m>:SPI:FCONdition <DataOperator>

Selects the operator for the MISO and MOSI pattern.

Parameters:

<DataOperator> EQUal | NEQual

*RST: EQUal

TRIGger<m>:SPI:MISopattern <MISOPattern>

Specifies the pattern to be triggered on the MISO line.

Parameters:

<MISOPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

TRIGger<m>:SPI:MOSipattern <MOSIPattern>

Specifies the pattern to be triggered on the MOSI line.

Parameters:

<MOSIPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

17.17.4.3 SPI Decode Results

BUS<m>:SPI:FRAMe<n>:DATA?....1428
BUS<m>:SPI:FCOunt?....1428
BUS<m>:SPI:FRAMe<n>:COUNt?....1428
BUS<m>:SPI:FRAMe<n>:BITRate?....1429
BUS<m>:SPI:FRAMe<n>:STATus?....1429
BUS<m>:SPI:FRAMe<n>:STARt?....1430
BUS<m>:SPI:FRAMe<n>:STOP?....1430
BUS<m>:SPI:FRAMe<n>:WCOunt?....1430
BUS<m>:SPI:FRAMe<n>:WORD<o>:STARt?....1431
BUS<m>:SPI:FRAMe<n>:WORD<o>:STOP?....1431
BUS<m>:SPI:FRAMe<n>:WORD<o>:MISO?....1431
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMISo?....1432
BUS<m>:SPI:FRAMe<n>:WORD<o>:MOSI?....1432
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMOSi?....1433

BUS<m>:SPI:FRAMe<n>:DATA?

Returns the data words of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameData> Comma-separated sequence of integer values (N, L1, R1,…, LN, RN). N is the number of word pairs in the frame, and {L1,R1}…{LN,RN} are the value pairs. The values Lx and Rx are associated with the MOSI and the MISO channel, respectively. If a channel is disabled, an empty value is returned.

Example: `BUS:SPI:FRAMe3:DATA?`
`<-- 2,10,108,35,70` (MOSI+MISO)
`2,10,,35,` (MOSI only)
`2,,108,,70` (MISO only)

Usage: Query only

BUS<m>:SPI:FCOunt?
BUS<m>:SPI:FRAMe<n>:COUNt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

Return values:

<Count> Total number of decoded frames.

Usage: Query only

BUS<m>:SPI:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:SPI:FRAMe<n>:STATus?

Returns the overall state of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | VOID | INCFirst | INCLast | INSufficient

OK: the frame is valid.
VOID: the frame is empty.
INCFirst: INComplete First word. The first word does not have the expected word length.
INCLast: INComplete Last word. The last word does not have the expected word length.
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:SPI:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.	
<n>	* Selects the frame.	

Return values:

<FrameStart>	Range:	-100E+24 to 100E+24
	*RST:	0
	Default unit:	s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.	
<n>	* Selects the frame.	

Return values:

<FrameStop>	Range:	-100E+24 to 100E+24
	*RST:	0
	Default unit:	s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WCOunt?

Returns the number of words in the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.	
<n>	* Selects the frame.	

Return values:

<WordCount>	Range:	0 to 4096
	*RST:	0

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:STARt?

Returns the start time of the specified data word.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the word number.

Return values:

<Start>	Range: -100E+24 to 100E+24 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:STOP?

Returns the end time of the specified data word.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the word number.

Return values:

<Stop>	Range: -100E+24 to 100E+24 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:MISO?

Returns the data value of the specified word on the MISO line.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the word number.

Return values:

<MISOValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.

Range: 0 to 4294967295
*RST: 0

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:FMISo?

Returns the formatted value of the specified word on the MISO line.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:

<FormattedMISOVal>

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:MOSI?

Returns the data value of the specified word on the MOSI line.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:

<MOSIValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.

Range: 0 to 4294967295
*RST: 0

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:FMOSi?

Returns the formatted value of the specified word on the MOSI line.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:

<FormattedMOSIVal>

Usage: Query only

17.17.4.4 SPI Search Settings

In search setup commands, you must specify the <SearchName> parameter. It is a string parameter that contains the search definition name. All commands are similar to SPI trigger commands.

SEARch:TRIGger:SPI:MODE....1433
SEARch:TRIGger:SPI:FCONdition....1434
SEARch:TRIGger:SPI:MISopattern....1434
SEARch:TRIGger:SPI:MOSipattern....1434
SEARch:TRIGger:SPI:DPOPerator....1434
SEARch:TRIGger:SPI:DPOSition....1435
SEARch:TRIGger:SPI:DPTO....1435
SEARch:TRIGger:SPI:PALignment....1435

SEARch:TRIGger:SPI:MODE <SearchName>,<Type>
SEARch:TRIGger:SPI:MODE? <SearchName>

Sets the event to be searched for.

Parameters:

<Type> SSACtive | TIMeout | MOSI | MISO | MOMI

SSACtive
Searches for the start of the frame when slave select signal SS changes to the active state. This type is available if the slave select line is configured in the bus setup, and `BUS<m>:SPI:FRCondition` is `SS`.

TIMeout
Searches for the start of the frame when the clock idle time exceeds the timeout. This type is available if the slave select line is configured in the bus setup, and `BUS<m>:SPI:FRCondition` is `CLKTimeout`.

MOSI | MISO
Searches for a specified data pattern expected on the MOSI line or on the MISO line, respectively.

MOMI
Searches in parallel for specified data patterns expected on the MOSI and MISO lines.

*RST: SSACtive

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

Firmware/Software: FW 3.30

SEARch:TRIGger:SPI:FCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:SPI:FCONdition? <SearchName>

Selects the operator for the data pattern: equal or not equal.

Parameters:
<DataOperator> EQUal | NEQual

*RST: EQUal

Parameters for setting and query:
<SearchName>

Firmware/Software: FW 3.30

SEARch:TRIGger:SPI:MISopattern <SearchName>,<MISOPattern>
SEARch:TRIGger:SPI:MISopattern? <SearchName>
SEARch:TRIGger:SPI:MOSipattern <SearchName>,<MOSIPattern>
SEARch:TRIGger:SPI:MOSipattern? <SearchName>

Specifies a data pattern for the MISO or MOSI line, respectively.

Parameters:
<MISOPattern>
<MOSIPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

Firmware/Software: FW 3.30

SEARch:TRIGger:SPI:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:SPI:DPOPerator? <SearchName>

Operator for the data position. You can define an exact position, a position range, or let the position undefined (ANY).

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF, INRange = RANGe

*RST: ANY

Parameters for setting and query:

<SearchName>

Firmware/Software: FW 3.30

SEARch:TRIGger:SPI:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:SPI:DPOSition? <SearchName>

Sets the number of bits or words before the first word of interest, see also `SEARch:TRIGger:SPI:PALignment`. These offset bits/words are skipped. The index 0 is associated with the first data bit or word.

If the position operator defines a range, also define the last bit/word of interest using `SEARch:TRIGger:SPI:DPTO`

Parameters:

<DataPosition> Range: 0 to 32767
Increment: 1
*RST: 0

Parameters for setting and query:

<SearchName>

Firmware/Software: FW 3.30

SEARch:TRIGger:SPI:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:SPI:DPTO? <SearchName>

Sets the the end value of a data postion range.

Parameters:

<DataPositionTo> Range: 1 to 32767
Increment: 1
*RST: 1

Parameters for setting and query:

<SearchName>

Firmware/Software: FW 3.30

SEARch:TRIGger:SPI:PALignment <SearchName>,<DataAlignment>
SEARch:TRIGger:SPI:PALignment? <SearchName>

Defines how the specified data pattern is searched.

Parameters:

<DataAlignment> WORD | BIT

WORD
The pattern is matched only at word boundaries.

BIT
Bit-by-bit: the pattern can start at any position in the message.

*RST: WORD

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

Firmware/Software: FW 3.30

17.17.4.5 SPI Search Results

The search on decoded SPI data returns the same results as the queries for decode results.

In search result commands, you must specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

The suffix FRAMe<m> indicates the frame index. The suffix WORD<n> indicates the word index inside a frame-

For a description of the returned values, see the corresponding commands in Chapter 17.17.4.3, "SPI Decode Results", on page 1428. All SPI search commands are first implemented in firmware version 3.30.

SEARch:RESult:SPI:FCOunt?....1436
SEARch:RESult:SPI:FRAMe<m>:COUNt?....1437
SEARch:RESult:SPI:FRAMe<m>:DATA?....1437
SEARch:RESult:SPI:FRAMe<m>:STARt?....1437
SEARch:RESult:SPI:FRAMe<m>:STATus?....1437
SEARch:RESult:SPI:FRAMe<m>:STOP?....1438
SEARch:RESult:SPI:FRAMe<m>:WCOunt?....1438
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MISO?....1438
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MOSI?....1439
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STARt?....1439
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STOP?....1439

SEARch:RESult:SPI:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:
<Count>
Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:COUNt?

Returns the number of frames that have matched the search criteria. In the search result table on the display, the number of rows is the number of frames that match the search criteria.

Suffix:
<m> *

Return values:
<Count>

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameData>

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | VOID | INCFirst | INCLast | INSufficient
*RST: OK

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:STOP? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WCOunt? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameWordCount> Range: 0 to 4096
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MISO? <SearchName>

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<WordMISOValue> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MOSI? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:

<SearchName>

Return values:

<WordMOSIValue> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STARt? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:

<SearchName>

Return values:

<FrameWordStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STOP? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:

<SearchName>

Return values:

<FrameWordStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

17.17.5 UART/RS-232/RS-422/RS-485 (Option R&S RTE-K2)

- Configuration....1440
- Trigger....1445
- Decode Results....1447

17.17.5.1 Configuration

BUS<m>:UART:RX:SOURce....1440
BUS<m>:UART:TX:SOURce....1440
BUS<m>:UART:RX:THReshold....1441
BUS<m>:UART:TX:THReshold....1441
BUS<m>:UART:TECHnology....1441
BUS<m>:UART:BITRate....1442
BUS<m>:UART:BAUDrate....1442
BUS<m>:UART:PARity....1442
BUS<m>:UART:POLarity....1443
BUS<m>:UART:SBIT....1443
BUS<m>:UART:SSIZe....1443
BUS<m>:UART:PACKets....1444
BUS<m>:UART:TOUT....1444
BUS<m>:UART:EWORd....1445

BUS<m>:UART:RX:SOURce <RxSource>

Selects the input channel for the receiver signal.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<RxSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for RX and TX lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: None

Usage: Asynchronous command

BUS<m>:UART:TX:SOURce <TxSource>

Selects the input channel for the transmitter signal.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<TxSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for RX and TX lines.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

BUS<m>:UART:RX:THReshold <RxThreshold>

Sets a user-defined threshold value for the Rx line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:UART:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<RxThreshold> User-defined clock threshold

Range: -15 to 15
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:UART:TX:THReshold <TxThreshold>

Sets a user-defined threshold value for the Tx line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:UART:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<TxThreshold> User-defined clock threshold

Range: -15 to 15
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:UART:TECHnology <Technology>

Sets the threshold voltage Tx and Rx lines as defined for various signal technologies.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15 | V25 | V165 | V125 | V09 | V38 | V20 | V0
1.5 V, 2.5 V, 1.65 V ... respectively

VM13
-1.3 V (negative value)

MANual
Manual setting of user-defined values with `BUS<m>:UART:RX:THReshold` and `BUS<m>:UART:TX:THReshold`.

*RST: V165

BUS<m>:UART:BITRate <Bitrate>

Sets the number of transmitted bits per second.

Suffix:

<m> 1..4

Parameters:

<Bitrate>

Range: 300 to 20000000
Increment: 1
*RST: 9600
Default unit: bps

Usage: Asynchronous command

BUS<m>:UART:BAUDrate <Bitrate>

Same as `BUS<m>:UART:BITRate`.

Suffix:

<m> 1..4

Parameters:

<Bitrate>

Range: 300 to 20000000
Increment: 1
*RST: 9600
Default unit: bps

Usage: Asynchronous command

BUS<m>:UART:PARity <Parity>

Defines the optional parity bit that is used for error detection.

See also: "Parity" on page 517.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Parity> NONE | ODD | EVEN | MARK | SPC | DC

MARK
The parity bit is always a logic 1.

SPC
SPaCe: The parity bit is always a logic 0.

DC
Don't Care: the parity is ignored.

*RST: NONE

Usage: Asynchronous command

BUS<m>:UART:POLarity <Polarity>

Defines the idle state of the bus. The idle state corresponds to a logic 1. The transmitted data on the bus is high (high = 1) or low (low = 1) active.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Polarity> IDLLow | IDLHigh

*RST: IDLHigh

Usage: Asynchronous command

BUS<m>:UART:SBIT <StopBits>

Sets the number of stop bits: 1; 1.5 or 2 stop bits are possible.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<StopBits> B1 | B15 | B2

*RST: B1

Usage: Asynchronous command

BUS<m>:UART:SSIZe <DataBits>

Sets the number of data bits in a message.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<DataBits> Number of data bits. 9 data bits are only possible with parity = none.

Range: 5 to 9
Increment: 1
*RST: 8

Usage: Asynchronous command

BUS<m>:UART:PACKets <Packets>

Defines the method of packet separation. A packet is a number of subsequent words in a date stream.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Packets> NONE | EWORd | TOUT

NONE
Packets are not considered.

EWORd
End word, the end condition of a packet is a pattern. To define the end word, use `BUS<m>:UART:EWORd`

TOUT
Defines a timeout between the packets. To set the timeout, use `BUS<m>:UART:TOUT`

*RST: NONE

Firmware/Software: FW 2.25

BUS<m>:UART:TOUT <InterframeTime>

Sets the timeout between packets in a UART data stream. A new packet starts with the first start bit after the timeout.

The command is relevant if `BUS<m>:UART:PACKets` is set to `TOUT`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<InterframeTime> Range: 1E-6 to 1
Increment: 1
*RST: 1E-3
Default unit: s

Usage: Asynchronous command

Firmware/Software: FW 2.25

BUS<m>:UART:EWORd <EndOfFrame>

Sets the end pattern of the packets. A new packet starts with the first start bit after the defined end pattern.

The command is relevant if `BUS<m>:UART:PACKets` is set to `EWORd`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<EndOfFrame> End word value in decimal format (range 0 to 255) or hexadecimal format (prefix #H). The query always returns hexadecimal values.

Example:

```
:BUS:UART:PACK EWOR
:BUS:UART:EWOR 10 // Decimal value
:BUS:UART:EWOR?
#H0A // Query returns hex
```

Example:

```
:BUS:UART:PACK EWOR
:BUS:UART:EWOR #Hff // Hexadecimal, prefix #H
:BUS:UART:EWOR?
#HFF
```

Usage: Asynchronous command

Firmware/Software: FW 2.25

17.17.5.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:UART:TYPE....................1446
TRIGger<m>:UART:SOURce....................1446
TRIGger<m>:UART:DPOPerator....................1446
TRIGger<m>:UART:DPOSition....................1446
TRIGger<m>:UART:DPTO....................1447
TRIGger<m>:UART:FCONdition....................1447
TRIGger<m>:UART:DATA....................1447

TRIGger<m>:UART:TYPE <Type>

Selects the trigger type for UART analysis.

See also: "Type" on page 519

Parameters:

<Type> STBT | PCKS | DATA | PRER | BRKC | STPerror

STBT: Start bit
PCKS: Packet start
DATA: Serial pattern
PRER: Parity error
BRKC: Break condition
STPerror: Stop error

*RST: STBT

Usage: Asynchronous command

TRIGger<m>:UART:SOURce <Source>

Selects the transmitter or receiver line as trigger source.

Parameters:

<Source> TX | RX

*RST: TX

Usage: Asynchronous command

TRIGger<m>:UART:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> EQUal | GETHan | INRange | RANGe

INRange = RANGe

*RST: GETHan

Usage: Asynchronous command

TRIGger<m>:UART:DPOSition <DataPosition>

Sets the number of words before the first word of interest. These offset words are ignored.

Parameters:

<DataPosition> Number of words

Range: 0 to 32767
Increment: 1
*RST: 0

Usage: Asynchronous command

TRIGger<m>:UART:DPTO <DataPositionTo>

Defines the last word of interest, if `TRIGger<m>:UART:DPOPerator` defines a position range.

Parameters:

<DataPositionTo>	Range:	0 to 32767
	Increment:	1
	*RST:	0

Usage: Asynchronous command

TRIGger<m>:UART:FCONdition <DataOperator>

Selects the operator for the data pattern (`TRIGger<m>:UART:DATA`).

Parameters:

<DataOperator> EQUal | NEQual

*RST: EQUal

TRIGger<m>:UART:DATA <Data>

Specifies the data pattern to be found on the specified trigger source, in binary or hex format. Enter the words in msb first bit order.

Parameters:

<Data> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

17.17.5.3 Decode Results

BUS<m>:UART:WORD<n>:RXValue? 1447
BUS<m>:UART:WORD<n>:TXValue? 1447
BUS<m>:UART:WORD<n>:BITRate? 1448
BUS<m>:UART:WORD<n>:COUNt? 1448
BUS<m>:UART:WORD<n>:SOURce? 1448
BUS<m>:UART:WORD<n>:STARt? 1449
BUS<m>:UART:WORD<n>:STATe? 1449

BUS<m>:UART:WORD<n>:RXValue?
BUS<m>:UART:WORD<n>:TXValue?

Returns the value of the specified word on the Rx line or Tx line, respectively.

Suffix:

<m> 1..4
Selects the serial bus.

<n>	* Selects the word.

Return values:

<Value>	To set the value format, use FORMat:BPATtern. The stated values for range, increment and reset are decimal values. Range: 0 to 511 *RST: 0

Usage: Query only

BUS<m>:UART:WORD<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* The suffix is irrelevant.

Return values:

<PrimaryBitRate>	Range: 0 to 100000000000 Increment: 1 *RST: 0 Default unit: bps

Usage: Query only

BUS<m>:UART:WORD<n>:COUNt?

Returns the number of words in the acquisition.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* The suffix is irrelevant.

Return values:

<Count>	Number of words

Usage: Query only

BUS<m>:UART:WORD<n>:SOURce?

Returns the line on which the specified word was transferred.

Suffix:

<m>	1..4 Selects the serial bus.

<n>	*
	Selects the word.

Return values:

<WordSource>	TX | RX
	*RST: TX

Usage: Query only

BUS<m>:UART:WORD<n>:STARt?

Returns the start time of the specified word.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the word.

Return values:

<WordStart>	Range: -100E+24 to 100E+24 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:UART:WORD<n>:STATe?

Returns the status of the specified word.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the word.

Return values:

<WordState>

OK | FRSTart | FRENd | FRME | BREak | STERror | SPERror | PRERror | INSufficient

OK: the frame is valid.
BREak:stop bit error with 0x00 word
STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

17.17.6 CAN (Option R&S RTE-K3/R&S RTE-K9)

- Configuration....1450
- Trigger....1455
- Decode Results....1463
- Search Settings....1471
- Search Results....1480
- Symbolic Trigger, Decode and Search....1486

17.17.6.1 Configuration

BUS<m>:CAN:DATA:SOURce....1450
BUS<m>:CAN:TYPE....1451
BUS<m>:CAN:FDATa:PSTandard....1451
BUS<m>:CAN:DATA:THReshold....1451
BUS<m>:CAN:TECHnology....1452
BUS<m>:CAN:BITRate....1452
BUS<m>:CAN:FDATa:ENABle....1452
BUS<m>:CAN:FDATa:DBITrate....1453
BUS<m>:CAN:FDATa:SAMPlepoint....1453
BUS<m>:CAN:SAMPlepoint....1453
BUS<m>:CAN:FDATa:T1Segment....1453
BUS<m>:CAN:T1Segment....1453
BUS<m>:CAN:FDATa:T2Segment....1454
BUS<m>:CAN:T2Segment....1454
BUS<m>:CAN:FDATa:JWIDth....1454
BUS<m>:CAN:JWIDth....1454

BUS<m>:CAN:DATA:SOURce <DataSource>

Sets the source of the data line that is selected with `BUS<m>:CAN:TYPE`.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<DataSource>	C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 *RST: C1W1

Usage: Asynchronous command

BUS<m>:CAN:TYPE <SignalType>

Selects the CAN-High or CAN-Low line. Both lines are required for differential signal transmission used by CAN.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<SignalType>	CANH | CANL *RST: CANL

BUS<m>:CAN:FDATa:PSTandard <ProtStd>

Only available for CAN FD buses. Selects whether the tested signal is an ISO CAN FD signal or not.

Suffix:

<m>	1..4

Parameters:

<ProtStd>

ISO | NISO

ISO
Signals are decoded according to the the ISO CAN FD protocol. This protocol has an additional stuff count field before the CRC sequence.

NISO
Non-ISO. Signals are decoded according to the the Bosch CAN FD protocol.

*RST: ISO

Firmware/Software: FW 3.35

BUS<m>:CAN:DATA:THReshold <Threshold>

Sets a user-defined threshold value.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:CAN:TECHnology`.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<Threshold>	Range: -12 to 12 Increment: 0.1 *RST: 0 Default unit: V

BUS<m>:CAN:TECHnology <Technology>

Sets the threshold voltage as defined for various signal technologies.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Technology> V25 | V3 | V2 | V0 | MANual

V25
2.5 Volt (CMOS 5.0 V)

V3
3.0 Volt (CAN_H HS / CAN_L LS)

V2
2.0 Volt (CAN_L HS / CAN_H LS)

V0
Ground

MANual
Manual setting of user-defined values with `BUS<m>:CAN:DATA:THReshold`.

*RST: V25

BUS<m>:CAN:BITRate <Bitrate>

For CAN buses, the "Bit rate" sets the number of transmitted bits per second.

For CAN FD buses, this parameter is called "Arbitration rate" and sets the bit rate of the arbitration phase.

The maximum bit rate for High Speed CAN is 1 Mbit/s. The bit rate is uniform and fixed for a given CAN or CAN FD bus.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Bitrate>
Range: 100 to 5E+6
Increment: 1
*RST: 100E+3
Default unit: bps

BUS<m>:CAN:FDATa:ENABle <CANFDEnabled>

Enables the CAN FD protocol configuration.

The setting is available in CAN FD option R&S RTE-K9.

Suffix:

<m> 1..4

Parameters:

<CANFDEnabled> ON | OFF

*RST: ON

BUS<m>:CAN:FDATa:DBITrate <FlexDatBitrate>

Sets the bit rate of the data phase. The data rate can be higher than the arbitration rate, but it is uniform and fixed for a given CAN FD bus.

The setting is available in CAN FD option R&S RTE-K9.

Suffix:

<m> 1..4

Parameters:

<FlexDatBitrate> Range: 100 to 15E+6
Increment: 1
*RST: 1E+6
Default unit: bps

BUS<m>:CAN:FDATa:SAMPlepoint <FlexDatSmpPt>
BUS<m>:CAN:SAMPlepoint <SamplePoint>

Sets the position of the sample point within the bit in percent of the nominal bit time.

Alternatively, you can set the sample point with `BUS<m>:CAN:T1Segment` and `BUS<m>:CAN:T2Segment`.

For CAN FD signals, `BUS<m>:CAN:SAMPlepoint` defines the synchronization of the arbitration phase, and `BUS<m>:CAN:FDATa:SAMPlepoint` defines the synchroniza-tion of the data phase.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<FlexDatSmpPt>
<SamplePoint> Range: 12 to 96 if number of segments is 24. The range reduces if the signal has fewer segments.
Increment: 5
*RST: 66
Default unit: %

BUS<m>:CAN:FDATa:T1Segment <FlexDatTimeSeg1>
BUS<m>:CAN:T1Segment <TimeSeg1>

Sets the number of time quanta before the sample point (T1Segment). T1Segment comprises the segments Synch_seg, Prop_seg, and Phase_seg1 which are specified in the CAN standard.

Make sure to set also `BUS<m>:CAN:T2Segment` for correct definition of the sample point. Alternatively, you can use `BUS<m>:CAN:SAMPlepoint`.

See also: "Synchronization: Sample point, Time segments, Jump width" on page 527

For CAN FD signals, `BUS<m>:CAN:T1Segment` defines the synchronization of the arbitration phase, and `BUS<m>:CAN:FDATa:T1Segment` defines the synchronization of the data phase.

Suffix:

<m>	1..4

Parameters:

<FlexDatTimeSeg1> <TimeSeg1> Time quanta

Range:	3 to 23
Increment:	1
*RST:	6.6

BUS<m>:CAN:FDATa:T2Segment <FlexDatTimeSeg2>
BUS<m>:CAN:T2Segment <TimeSeg2>

Sets the number of time quanta after the sample point (T2Segment). T2Segment matches Phase_seg2 specified in the CAN standard.

Make sure to set also `BUS<m>:CAN:T1Segment` on page 1453 for correct definition of the sample point. Alternatively, you can use `BUS<m>:CAN:SAMPlepoint`.

See also: "Synchronization: Sample point, Time segments, Jump width" on page 527

For CAN FD signals, `BUS<m>:CAN:T2Segment` defines the synchronization of the arbitration phase, and `BUS<m>:CAN:FDATa:T2Segment` defines the synchronization of the data phase.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<FlexDatTimeSeg2> <TimeSeg2> Time quanta

Range:	1 to 21
Increment:	1
*RST:	3.4

BUS<m>:CAN:FDATa:JWIDth <FlexDatJumpWdt>
BUS<m>:CAN:JWIDth <JumpWidth>

Defines the maximum number of time quanta for phase correction. Time segment1 may be lengthened or Time segment2 may be shortened due to resynchronization. Resynchronization corrects the phase error of an edge caused by the drift of the oscillators.

For CAN FD signals, this setting defines the synchronization of the arbitration phase.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<JumpWidth> Time quanta

Range: 1 to 4, available maximum depends on the number of segments and the sample point

Increment: 1

*RST: 1

17.17.6.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:CAN:TYPE ... 1455
TRIGger<m>:CAN:FDATa:STANdard ... 1456
TRIGger<m>:CAN:FTYPe ... 1456
TRIGger<m>:CAN:ITYPe ... 1457
TRIGger<m>:CAN:ICONdition ... 1457
TRIGger<m>:CAN:IMIN ... 1458
TRIGger<m>:CAN:IMAX ... 1458
TRIGger<m>:CAN:FDATa:FDF ... 1458
TRIGger<m>:CAN:FDATa:BRS ... 1458
TRIGger<m>:CAN:FDATa:ESI ... 1458
TRIGger<m>:CAN:DCONdition ... 1459
TRIGger<m>:CAN:DMIN ... 1459
TRIGger<m>:CAN:DMAX ... 1459
TRIGger<m>:CAN:BORDer ... 1459
TRIGger<m>:CAN:DLCCondition ... 1460
TRIGger<m>:CAN:DLC ... 1460
TRIGger<m>:CAN:NDBYtes? ... 1460
TRIGger<m>:CAN:FDATa:DPOPerator ... 1461
TRIGger<m>:CAN:FDATa:DPOSition ... 1461
TRIGger<m>:CAN:FDATa:DPTO ... 1461
TRIGger<m>:CAN:ACKerror ... 1461
TRIGger<m>:CAN:BITSterror ... 1462
TRIGger<m>:CAN:CRCerror ... 1462
TRIGger<m>:CAN:FORMerror ... 1462
TRIGger<m>:CAN:FDATa:SCERror ... 1462

TRIGger<m>:CAN:TYPE <Type>

Selects the trigger type for CAN analysis.

See: "Trigger type" on page 529.

Parameters:

<Type> STOF | FTYP | ID | IDDT | ERRC

STOF
STart Of Frame: triggers on the first edge of the dominant SOF bit (synchronization bit).

FTYP
Frame TYPe: triggers on a specified frame type (data, remote, error, or overload) and on the identifier format.
To set the frame type, use `TRIGger<m>:CAN:FTYPe`. ´Set the identifier format with `TRIGger<m>:CAN:ITYPe`

ID
IDentifier: Sets the trigger to one specific identifier or an identifier range. To set the identifier, use `TRIGger<m>:CAN:ICONdition`, `TRIGger<m>:CAN:IMIN`, and `TRIGger<m>:CAN:IMAX`.

IDDT
IDentifier and DaTa: Combination of identifier and data conditions To set the identifier condition, use `TRIGger<m>:CAN:ICONdition`, `TRIGger<m>:CAN:IMIN`, and `TRIGger<m>:CAN:IMAX`.
To set the data condition, use `TRIGger<m>:CAN:DCONdition`, `TRIGger<m>:CAN:DMIN`, and `TRIGger<m>:CAN:DMAX`.

ERRC
ERRor Condition: Define the error types with
`TRIGger<m>:CAN:ACKerror`,
`TRIGger<m>:CAN:BITSterror`,
`TRIGger<m>:CAN:CRCerror`,
`TRIGger<m>:CAN:FORMerror`,
`TRIGger<m>:CAN:FDATa:SCERror` on page 1462.

*RST: STOF

TRIGger<m>:CAN:FDATa:STANdard <Standard>

Selects the CAN standard. Use `ANY` if the standard of the signal is unknown.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:

<Standard> ANY | CAN | CANFd

*RST: CAN

TRIGger<m>:CAN:FTYPe <FrameType>

Selects the CAN frame type if `TRIGger<m>:CAN:TYPE` is set to FTYP (frame type) or ID (identifier).

For data and remote frames, the identifier format has to be set with `TRIGger<m>:CAN:ITYPe`.

See also: "Frame type" on page 530

Parameters:

<FrameType> ANY | DATA | REMote | ERRor | OVERload

Available values depend on the CAN standard and on the `TRIGger<m>:CAN:TYPE` setting:
Remote frames are not available in the CAN FD protocol.
If the trigger type is set to FTYP (frame type), you can set the values `DATA | REMote | ERRor | OVERload`.
If the trigger type is set to ID (identifier), you can set the values `ANY | DATA | REMote`.

*RST: ANY

TRIGger<m>:CAN:ITYPe <IdentifierType>

Selects the format of data and remote frames.

Remote frames are not available in the CAN FD protocol.

Parameters:

<IdentifierType> ANY | B11 | B29

B11
11 bit identifier (standard format). The instrument triggers on the sample point of the IDE bit.

B29
29 bit identifier (extended format). The instrument triggers on the sample point of the RTR bit.

ANY
The ID type and ID pattern are not relevant for the trigger condition.

*RST: ANY

TRIGger<m>:CAN:ICONdition <IdOperator>

Sets the operator to set a specific identifier or an identifier range.

Parameters:

<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `TRIGger<m>:CAN:IMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:CAN:IMIN` and `TRIGger<m>:CAN:IMAX` on page 1458.

*RST: EQUal

TRIGger<m>:CAN:IMIN <IdPattern>

Specifies a message identifier pattern, or sets the the start value of an identifier range.

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:CAN:IMAX <IdPatternTo>

Sets the the end value of an identifier range if `TRIGger<m>:CAN:ICONdition` is set to `INRange` or `OORange`.

Parameters:

<IdPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:CAN:FDATa:FDF <FDFBit>

Specifies the CAN FD frame format. It corresponds to the EDL bit (extended data length), which only exists in CAN FD format.

Parameters:

<FDFBit> ONE | ZERO | DC

ONE: CAN FD.
ZERO: CAN.
DC: don't care, the format is not relevant.

*RST: DC

TRIGger<m>:CAN:FDATa:BRS <BRSBit>

Sets the bit rate switch bit.

Parameters:

<BRSBit> ONE | ZERO | DC

ONE: the bit rate switches from the bit rate of the arbitration phase to the faster data rate.

*RST: ONE

TRIGger<m>:CAN:FDATa:ESI <ESIBit>

Sets the error state indicator bit.

Parameters:
<ESIBit> ONE | ZERO | DC
DC: don't care, bit is nor relevant
*RST: DC

TRIGger<m>:CAN:DCONdition <DataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:CAN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:CAN:DMIN` and `TRIGger<m>:CAN:DMAX`.
*RST: EQUal

TRIGger<m>:CAN:DMIN <DataPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:
<DataPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:CAN:DMAX <DataPatternTo>

Sets the the end value of an data range if `TRIGger<m>:CAN:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:CAN:BORDer <Endianness>

Sets the byte order (endianness) of the data transfer. Only for CAN protocol.

Parameters:
<Endianness> BENDian | LENDian

BENDian
Big endian, data is analyzed and evaluated in the order of reception.

LENDian
Little endian, the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

*RST: BENDian

TRIGger<m>:CAN:DLCCondition <DLCOperator>

Operator to set the data length code for triggering on CAN and CAN FD data.

For details, see "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 532.

The number of data bytes to be found is set with `TRIGger<m>:CAN:DLC`.

See also: `TRIGger<m>:CAN:BORDer`.

Parameters:

<DLCOperator> EQUal | GETHan

For little endian transfer direction, EQUal must be set.

*RST: GETHan

TRIGger<m>:CAN:DLC <WordCount>

Sets the Data Length Code, the number of data bytes to be found. For complete definition, set also the operator with `TRIGger<m>:CAN:DLCCondition` on page 1460.

Parameters:

<WordCount> Range: CAN: 1 to 8, CAN FD: 1 to 15 (64 bytes)
Increment: 1
*RST: 1

TRIGger<m>:CAN:NDBYtes?

Returns the number of data bytes defined by DLC. DLC and NDB are different in CAN FD for DLCs > 8.

See also: "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 532.

Return values:

<NDBytes> Range: 1 to 64
Increment: 1
*RST: 1

Usage: Query only

TRIGger<m>:CAN:FDATa:DPOPerator <DataPosOperator>

Sets the operator to define an exact position or a data range where the instrument looks for the specified data pattern.

The setting is available in CAN FD option R&S RTE-K9.

The position can be defined if the data field of the frame is longer than 8 bytes - if `TRIGger<m>:CAN:DLC`≥9.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The data position is not relevant for the trigger condition.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:CAN:FDATa:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `TRIGger<m>:CAN:FDATa:DPOSition` and `TRIGger<m>:CAN:FDATa:DPTO`.

*RST: ANY

TRIGger<m>:CAN:FDATa:DPOSition <DataPosition>

Defines the number of the first data byte at which the data pattern may start.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:

<DataPosition> Range: 1 to 57
Increment: 1
*RST: 1

TRIGger<m>:CAN:FDATa:DPTO <DataPositionTo>

Sets the number of the last byte at which the required data pattern may start.

Parameters:

<DataPositionTo> Range: 8 to 64
Increment: 1
*RST: 8

TRIGger<m>:CAN:ACKerror <AckError>

Triggers when the transmitter does not receive an acknowledgment - a dominant bit during the Ack Slot.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<AckError> ON | OFF

*RST: ON

TRIGger<m>:CAN:BITSterror <BitStuffError>

Triggers if astuff error occurs - when the 6th consecutive equal bit level in the mentioned fields is detected.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<BitStuffError> ON | OFF

*RST: ON

TRIGger<m>:CAN:CRCerror <ChecksumError>

Triggers on CRC errors. A CRC error occurs when the CRC calculated by the receiver differs from the received value in the CRC sequence.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<ChecksumError> ON | OFF

*RST: ON

TRIGger<m>:CAN:FORMerror <FormError>

Triggers when a fixed-form bit field contains one or more illegal bits.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<FormError> ON | OFF

*RST: ON

TRIGger<m>:CAN:FDATa:SCERror <StuffCountError>

Triggers on stuff count errors. A stuff count error occurs if th received stuff count value does not match the value calculated from the own stuff bit count.

The trigger type `TRIGger<m>:CAN:TYPE` must be set to `ERRC`.

Only relevant for CAN FD signals in ISO standard.

Parameters:

<StuffCountError> ON | OFF

*RST: ON

17.17.6.3 Decode Results

To load and acivate a label list, use:

- BUS<m>:NEWList on page 1384
- BUS<m>:SYMBols on page 1385

BUS<m>:CAN:FCOunt?..1463
BUS<m>:CAN:FDATa:FRAMe<n>:STANdard?..1463
BUS<m>:CAN:FRAMe<n>:STATus?..1464
BUS<m>:CAN:FRAMe<n>:NDBYtes?..1465
BUS<m>:CAN:FRAMe<n>:STARt?..1465
BUS<m>:CAN:FRAMe<n>:STOP?..1465
BUS<m>:CAN:FRAMe<n>:SYMBol?..1465
BUS<m>:CAN:FRAMe<n>:TYPE?..1466
BUS<m>:CAN:FRAMe<n>:DATA?..1466
BUS<m>:CAN:FRAMe<n>:ACKState?..1466
BUS<m>:CAN:FRAMe<n>:CSSTate?..1466
BUS<m>:CAN:FRAMe<n>:DLCState?..1466
BUS<m>:CAN:FRAMe<n>:IDSTate?..1466
BUS<m>:CAN:FRAMe<n>:ACKValue?..1467
BUS<m>:CAN:FRAMe<n>:BITRate?..1467
BUS<m>:CAN:FRAMe<n>:CSValue?..1468
BUS<m>:CAN:FRAMe<n>:DLCValue?..1468
BUS<m>:CAN:FRAMe<n>:IDTYpe?..1468
BUS<m>:CAN:FRAMe<n>:IDValue?..1469
BUS<m>:CAN:FRAMe<n>:BSEPosition?..1469
BUS<m>:CAN:FRAMe<n>:FERCause?..1469
BUS<m>:CAN:FRAMe<n>:SDEXport?..1470
BUS<m>:CAN:FDATa:FRAMe<n>:SCValue?..1470
BUS<m>:CAN:FRAMe<n>:BYTE<o>:STATe?..1470
BUS<m>:CAN:FRAMe<n>:BYTE<o>:VALue?..1471

BUS<m>:CAN:FCOunt?

Returns the number of decoded frames of the acquisition.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Total number of decoded frames.

Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CAN:FDATa:FRAMe<n>:STANdard?

Returns the CAN standard.

The setting is available in CAN FD option R&S RTE-K9.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
The frame suffix is irrelevant.

Return values:

<Standard> CAN | CANFd
*RST: CAN

Usage: Query only

BUS<m>:CAN:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | FORM | BTST | CRC | CRCD | NOACk | ACKD | EOFD | CAERror | FCERor | INSufficient | SERRror | SFERror | SCERror | SAERror | SCAE | SCFE

OK: the frame is valid.
FORM: Fixed-bit form error
BTST: Bit stuffing error occured.
CRC: Cyclic redundancy check failed.
CRCD: Wrong CRC delimiter occured.
NOACK: Acknowlegde is missing.
ACKD: Wrong ACK delimiter occured.
EOFD: Wrong end of frame.
CAERror: CRC error followed by an acknowledgement error (missing acknowledge)
FCERor: CRC error followed by a form error (wrong CRC delimiter or wrong ACK delimiter)
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.
SERRror: Stuff count error (CAN-FD ISO only)
SFER: Stuff count error and FORM error (CAN-FD ISO only)
SCER: Stuff count error and CRC error (CAN-FD ISO only)
SAER: Stuff count error and ACK error (CAN-FD ISO only)
SCAE: Stuff count error and CRC error and ACK error (CAN-FD ISO only)
SCFE: Stuff count error and CRC error and FORM error (CAN-FD ISO only)

	*RST: OK
Usage:	Query only

BUS<m>:CAN:FRAMe<n>:NDBYtes?

REturns the number of data bytes.

Suffix:

<m>	1..4
<n>	*

Return values:

<NDBytes>	Range: 1 to 64 Increment: 1 *RST: 1
Usage:	Query only
Firmware/Software:	FW 3.35

BUS<m>:CAN:FRAMe<n>:STARt?
BUS<m>:CAN:FRAMe<n>:STOP?

Return the start time and stop time of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<Start>, <Stop>	Time Range: -100E+24 to 100E+24 *RST: 0 Default unit: s
Usage:	Query only

BUS<m>:CAN:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Label>	String with symbolic label of the identifier

Example: `BUS:CAN:FRAMe:SYMBol?`
Response: `Temperature`

Usage: Query only

BUS<m>:CAN:FRAMe<n>:TYPE?

Returns the frame type of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> DATA | REMote | ERR | OVLD
Data, remote, error or overload frame.
*RST: DATA

Usage: Query only

BUS<m>:CAN:FRAMe<n>:DATA?

Returns the data of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data> Comma-separated list of values. The first value is the number of bytes, followed by the values of the data bytes.

Example: `BUS1:CAN:FRAMe2:DATA?`
`--> 3,208,231,32`
Returns the data of the second frame: the number of bytes is 3 data (first value).

Usage: Query only

BUS<m>:CAN:FRAMe<n>:ACKState?
BUS<m>:CAN:FRAMe<n>:CSSTate?
BUS<m>:CAN:FRAMe<n>:DLCState?
BUS<m>:CAN:FRAMe<n>:IDSTate?

Return the states of following parts of a message

- ACKState: state of acknowledgement field
- CSSTate: state of checksum field (CRC)

- DLCState: state of data length code
- IDSTate: identifier state

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<State> OK | ERRor | UNDF
UNDF: Undefined
*RST: OK

Usage: Query only

BUS<m>:CAN:FRAMe<n>:ACKValue?

Returns the value of the acknowledge slot for the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AckValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.
Range: 0 to 1
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:CAN:FRAMe<n>:CSValue?

Returns the CRC sequence value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.

Range: 0 to 2097151
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:DLCValue?

Returns the data length code of the selected frame - the number of data bytes in the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameDLCValue> Number of data bytes in decimal values.

Range: 0 to 15
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:IDTYpe?

Returns the identifier type of the selected frame, the identifier format of data and remote frames.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierType> ANY | B11 | B29

B11: standard format, 11 bit
B29: extended format, 29 bit

*RST: B11

Usage: Query only

BUS<m>:CAN:FRAMe<n>:IDValue?

Returns the identifier value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 536870911
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BSEPosition?

Returns the location of a bit stuffing error.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<BitStuffErrorPos> Time when the error occured

Range: 0 to 5000
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:FERCause?

Returns information on a form error, if the frame status query returned a FORM error.

See also: `BUS<m>:CAN:FRAMe<n>:STATus?` on page 1464

Suffix:

<m>	1..4
<n>	*

Return values:

<FormErrorCause> NONE | CRCDerror | ACKDerror | FSBE | RESerror

CRCD = CRC delimiter error
ACKD = ACK delimiter error
FSBE = Fixed stuff bit error (CAN-FD ISO only)
RESE = Reserved bit error

*RST: NONE

Usage: Query only

BUS<m>:CAN:FRAMe<n>:SDEXport?

Returns the symbolic data of the frame in export format.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:CAN:FDATa:FRAMe<n>:SCValue?

Returns the stuff bit count modulo 8 value.

Suffix:

<m>	1..4
<n>	*

Return values:

<StuffCount>

Range: 0 to 7
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BYTE<o>:STATe?

Returns the state of the specified byte.

Suffix:

<m>	1..4 Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<State> OK | ERRor | UNDF
UNDF: Undefined
*RST: OK

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BYTE<o>:VALue?

Returns the value of the specified byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<Value> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.
Range: 0 to 255
*RST: 0

Usage: Query only

17.17.6.4 Search Settings

SEARch:TRIGger:CAN[:SSOFrame]....1472
SEARch:TRIGger:CAN:SFTYpe....1472
SEARch:TRIGger:CAN:SFIDentifier....1472
SEARch:TRIGger:CAN:SIDData....1473
SEARch:TRIGger:CAN:SERRor....1473
SEARch:TRIGger:CAN:FDATa:STANdard....1473
SEARch:TRIGger:CAN:FTYPe....1473
SEARch:TRIGger:CAN:ITYPe....1474
SEARch:TRIGger:CAN:ICONdition....1474
SEARch:TRIGger:CAN:IMAX....1474
SEARch:TRIGger:CAN:IMIN....1475
SEARch:TRIGger:CAN:DCONdition....1475
SEARch:TRIGger:CAN:DMIN....1475
SEARch:TRIGger:CAN:DMAX....1476

SEARch:TRIGger:CAN:DLCCondition....1476
SEARch:TRIGger:CAN:DLC....1476
SEARch:RESult:CAN:FRAMe<m>:NDBYtes?....1476
SEARch:TRIGger:CAN:FDATa:DPOPerator....1477
SEARch:TRIGger:CAN:FDATa:DPOSition....1477
SEARch:TRIGger:CAN:FDATa:DPTO....1478
SEARch:TRIGger:CAN:ACKerror....1478
SEARch:TRIGger:CAN:BITSterror....1478
SEARch:TRIGger:CAN:CRCerror....1478
SEARch:TRIGger:CAN:FORMerror....1479
SEARch:TRIGger:CAN:FDATa:SCERror....1479
SEARch:TRIGger:CAN:FDATa[:FDF]....1479
SEARch:TRIGger:CAN:FDATa:BRS....1479
SEARch:TRIGger:CAN:FDATa:ESI....1480

SEARch:TRIGger:CAN[:SSOFrame] <SearchName>,<FrameStart>
SEARch:TRIGger:CAN[:SSOFrame]? <SearchName>

Enables the search for a start of frame.

Parameters:

<FrameStart> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:SFTYpe <SearchName>,<FrameType>
SEARch:TRIGger:CAN:SFTYpe? <SearchName>

Enables the search for a specified frame type.

Parameters:

<FrameType> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:SFIDentifier <SearchName>,<Identifier>
SEARch:TRIGger:CAN:SFIDentifier? <SearchName>

Enables the search for frame indentifier.

Parameters:

<Identifier> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:SIDData <SearchName>,<IdentifierData>
SEARch:TRIGger:CAN:SIDData? <SearchName>

Enables the search for identifier and data.

Parameters:

<IdentifierData>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:CAN:SERRor? <SearchName>

Enables the search for a specified error.

Parameters:

<ErrorCondition>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FDATa:STANdard <SearchName>,<Standard>
SEARch:TRIGger:CAN:FDATa:STANdard? <SearchName>

Selects the CAN standard: CAN, CAN FD, or Any.

Use "Any" to search on either CAN or CAN-FD frame. In this case, the search configuration provides all possible settings, for CAN as well as for CAN FD.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:

<Standard>	ANY | CAN | CANFd	
	*RST:	CAN

Parameters for setting and query:

<SearchName>	String with the search name

SEARch:TRIGger:CAN:FTYPe <SearchName>,<FrameType>
SEARch:TRIGger:CAN:FTYPe? <SearchName>

Selects the CAN frame type to be searched for.

For data and remote frames, the identifier format has to be set with `SEARch:TRIGger:CAN:ITYPe` on page 1474.

Parameters:

<FrameType>	ANY | DATA | REMote | ERRor | OVERload	
	*RST:	ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:ITYPe <SearchName>,<IdentifierType>
SEARch:TRIGger:CAN:ITYPe? <SearchName>

Selects the format of data and remote frames: 11 bit for CAN base frames, or 29 bits for CAN extended frames.

Parameters:

<IdentifierType> ANY | B11 | B29

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:ICONdition <SearchName>,<IdOperator>
SEARch:TRIGger:CAN:ICONdition? <SearchName>

Sets the operator to set a specific identifier or an identifier range.

Parameters:

<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `SEARch:TRIGger:CAN:IMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:CAN:IMIN` and `SEARch:TRIGger:CAN:IMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:IMAX <SearchName>,<IdPatternTo>
SEARch:TRIGger:CAN:IMAX? <SearchName>

Sets the the end value of an identifier range if `SEARch:TRIGger:CAN:ICONdition` is set to `INRange` or `OORange`.

Parameters:

<IdPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:IMIN <SearchName>,<IdPattern>
SEARch:TRIGger:CAN:IMIN? <SearchName>

Specifies a message identifier pattern, or sets the the start value of an identifier range.

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:CAN:DCONdition? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:CAN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:CAN:DMIN` and `SEARch:TRIGger:CAN:DMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:CAN:DMIN? <SearchName>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:CAN:DMAX? <SearchName>

Sets the the end value of an data range if `SEARch:TRIGger:CAN:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:DLCCondition <SearchName>,<DLCOperator>
SEARch:TRIGger:CAN:DLCCondition? <SearchName>

Operator to set the data length code for search.

Parameters:

<DLCOperator> EQUal | GETHan

*RST: GETHan

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:DLC <SearchName>,<WordCount>
SEARch:TRIGger:CAN:DLC? <SearchName>

Sets the Data Length Code, the number of data bytes to be found. For complete definition, set also the operator with `SEARch:TRIGger:CAN:DLCCondition`.

Parameters:

<WordCount> Range: 1 to 8
Increment: 1
*RST: 1

Parameters for setting and query:

<SearchName>

SEARch:RESult:CAN:FRAMe<m>:NDBYtes? <SearchName>

Returns the number of data bytes defined by DLC. DLC and NDB are different in CAN FD for DLCs > 8.

See also: "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 532.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:
<NDBytes> Range: 1 to 64
Increment: 1
*RST: 1

Usage: Query only

SEARch:TRIGger:CAN:FDATa:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:CAN:FDATa:DPOPerator? <SearchName>

Sets the operator for the data position if DLC ≥ 9. You can define an exact position, or a position range.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:
<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The data position is not relevant for the search.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `SEARch:TRIGger:CAN:FDATa:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `SEARch:TRIGger:CAN:FDATa:DPOSition` and `SEARch:TRIGger:CAN:FDATa:DPTO`.

*RST: ANY

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:CAN:FDATa:DPOSition? <SearchName>

Defines the first possible start position of the data pattern.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:
<DataPosition> Range: 1 to 57
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:CAN:FDATa:DPTO? <SearchName>

Defines the last possible start position of the data pattern if the position operator `SEARch:TRIGger:CAN:FDATa:DPOPerator` defines a range.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:

<DataPositionTo>	Range:	8 to 64
	Increment:	1
	*RST:	8

Parameters for setting and query:

<SearchName> String with the search name

SEARch:TRIGger:CAN:ACKerror <SearchName>,<AckError>
SEARch:TRIGger:CAN:ACKerror? <SearchName>

Searches for acknowledgement errors. An acknowledgement error occurs when the transmitter does not receive an acknowledgment - a dominant bit during the Ack Slot.

Parameters:

<AckError>	ON | OFF	
	*RST:	ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:BITSterror <SearchName>,<BitStuffError>
SEARch:TRIGger:CAN:BITSterror? <SearchName>

Searches for bit stuffing errors.

Parameters:

<BitStuffError>	ON | OFF	
	*RST:	ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:CRCerror <SearchName>,<ChecksumError>
SEARch:TRIGger:CAN:CRCerror? <SearchName>

Searches for errors in the Cyclic Redundancy Check.

Parameters:

<ChecksumError>	ON | OFF	
	*RST:	ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CAN:FORMerror <SearchName>,<FormError>
SEARch:TRIGger:CAN:FORMerror? <SearchName>

Searches for form errors. A form error occurs when a fixed-form bit field contains one or more illegal bits.

Parameters:
<FormError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FDATa:SCERror <SearchName>,<StuffCountError>
SEARch:TRIGger:CAN:FDATa:SCERror? <SearchName>

Triggers on stuff count errors. A stuff count error occurs if th received stuff count value does not match the value calculated from the own stuff bit count.

Only relevant for CAN FD signals in ISO standard.

Parameters:
<StuffCountError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FDATa[:FDF] <SearchName>,<FDFBit>
SEARch:TRIGger:CAN:FDATa[:FDF]? <SearchName>

Sets the EDL bit (extended data length), which determines whether a frame is CAN or CAN-FD.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:
<FDFBit> ONE | ZERO | DC
ONE: CAN FD.
ZERO: CAN.
DC: don't care, the format is not relevant.
*RST: DC

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:BRS <SearchName>,<BRSBit>
SEARch:TRIGger:CAN:FDATa:BRS? <SearchName>

Sets the bit rate switching bit for identifier and identifier + data searches.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:

<BRSBit> ONE | ZERO | DC

DC: Don't care

*RST: ONE

Parameters for setting and query:

<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:ESI <SearchName>,<ESIBit>
SEARch:TRIGger:CAN:FDATa:ESI? <SearchName>

Sets the error state indicator bit for identifier and identifier + data searches.

The setting is available in CAN FD option R&S RTE-K9.

Parameters:

<ESIBit> ONE | ZERO | DC

*RST: DC

Parameters for setting and query:

<SearchName> String with the search name

17.17.6.5 Search Results

The search on decoded CAN data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.6.3, "Decode Results", on page 1463.

SEARch:RESult:CAN:FCOunt?.......................................1481
SEARch:RESult:CAN:FRAMe<m>:ACKValue?.......................................1481
SEARch:RESult:CAN:FRAMe<m>:BSEPosition?.......................................1481
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe?.......................................1481
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue?.......................................1482
SEARch:RESult:CAN:FRAMe<m>:ACKState?.......................................1482
SEARch:RESult:CAN:FRAMe<m>:CSSTate?.......................................1482
SEARch:RESult:CAN:FRAMe<m>:DLCState?.......................................1482
SEARch:RESult:CAN:FRAMe<m>:IDSTate?.......................................1482
SEARch:RESult:CAN:FRAMe<m>:CSValue?.......................................1482
SEARch:RESult:CAN:FRAMe<m>:DATA?.......................................1483
SEARch:RESult:CAN:FRAMe<m>:DLCValue?.......................................1483
SEARch:RESult:CAN:FRAMe<m>:FERCause?.......................................1483
SEARch:RESult:CAN:FRAMe<m>:IDTYpe?.......................................1484
SEARch:RESult:CAN:FRAMe<m>:IDValue?.......................................1484
SEARch:RESult:CAN:FDATa:FRAMe<m>:SCValue?.......................................1484
SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard?.......................................1484
SEARch:RESult:CAN:FRAMe<m>:STARt?.......................................1485
SEARch:RESult:CAN:FRAMe<m>:STATus?.......................................1485

SEARch:RESult:CAN:FRAMe<m>:STOP?... 1485
SEARch:RESult:CAN:FRAMe<m>:SYMBol?...1486
SEARch:RESult:CAN:FRAMe<m>:TYPE?...1486

SEARch:RESult:CAN:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:ACKValue? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<AckValue>	Range:	0 to 1
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:BSEPosition? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<BitStuffErrorPos>	Range:	0 to 5000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:

<State>	OK | ERRor | UNDF	
	*RST:	OK

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<Value>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:ACKState? <SearchName>
SEARch:RESult:CAN:FRAMe<m>:CSSTate? <SearchName>
SEARch:RESult:CAN:FRAMe<m>:DLCState? <SearchName>
SEARch:RESult:CAN:FRAMe<m>:IDSTate? <SearchName>

Suffix:

<m>	*

Query parameters:
<SearchName>

Return values:

<IdentifierState>	OK | ERRor | UNDF	
	*RST:	OK

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:CSValue? <SearchName>

Suffix:

<m>	*

Query parameters:
<SearchName>

Return values:

<ChecksumValue> Range: 0 to 2097151
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:DATA? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Data>

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:DLCValue? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameDLCValue> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:FERCause? <SearchName>

Returns information on a form error, if the frame status query returned a FORM error.

See also: `SEARch:RESult:CAN:FRAMe<m>:STATus?` on page 1485.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FormErrorCause> NONE | CRCDerror | ACKDerror | FSBE | RESerror

See `BUS<m>:CAN:FRAMe<n>:FERCause?` on page 1469.

*RST: NONE

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:IDTYpe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierType> ANY | B11 | B29
*RST: B11

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:IDValue? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierValue> Range: 0 to 536870911
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FDATa:FRAMe<m>:SCValue? <SearchName>

Returns the stuff bit count modulo 8.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<StuffCount> Range: 0 to 7
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard? <SearchName>

Returns the CAN protocol standard: CAN or CAN FD.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Standard>	CAN | CANFd *RST: CAN

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:STARt? <SearchName>

Suffix:

<m>	*

Query parameters:
<SearchName>

Return values:

<FrameStart>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:STATus? <SearchName>

Suffix:

<m>	*

Query parameters:
<SearchName>

Return values:

<FrameState>	OK | FORM | BTST | CRC | CRCD | NOACk | ACKD | EOFD | CAERror | FCERor | INSufficient | SERRror | SFERror | SCERror | SAERror | SCAE | SCFE See `BUS<m>:CAN:FRAMe<n>:STATus?` on page 1464. *RST: OK

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:STOP? <SearchName>

Suffix:

<m>	*

Query parameters:
<SearchName>

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:SYMBol? <SearchName>

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> *
Selects the number of the frame in the current acquisition, 1...n.

Query parameters:

<SearchName> String parameter that contains the search definition name

Return values:

<Label> Symbolic label (string)

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:TYPE? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameType> DATA | REMote | ERR | OVLD
*RST: DATA

Usage: Query only

17.17.6.6 Symbolic Trigger, Decode and Search

- Symbolic Trigger..1486
- Symbolic Decode Results..1488
- Symbolic Search..1488

Symbolic Trigger

TRIGger<m>:CAN:SYMBolic:MSGValue..1487
TRIGger<m>:CAN:SYMBolic:TSIGnals..1487
TRIGger<m>:CAN:SYMBolic:SIGValue..1487
TRIGger<m>:CAN:SYMBolic:DMAX..1487
TRIGger<m>:CAN:SYMBolic:DMIN..1487
TRIGger<m>:CAN:SYMBolic:SGEValue..1488

TRIGger<m>:CAN:SYMBolic:MSGValue <MessageName>

Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

See also: `SEARch:TRIGger:CAN:SYMBolic:MSGValue` on page 1489

Parameters:

<MessageName>	String with the symbolic message name

TRIGger<m>:CAN:SYMBolic:TSIGnals <TriggerOnSignal>

Enables the trigger on a specific signal value that is part of the selected message.

Parameters:

<TriggerOnSignal>	ON | OFF *RST: OFF

TRIGger<m>:CAN:SYMBolic:SIGValue <SignalName>

Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

See also: `SEARch:TRIGger:CAN:SYMBolic:SIGValue` on page 1490

Parameters:

<SignalName>	String with the signal name as defined in the DBC file.

TRIGger<m>:CAN:SYMBolic:DMAX <DataPatternTo>

Defines the maximum data value of the signal.

This value is required to specify a range if condition `INRange | OORange` is set with `TRIGger<m>:CAN:DCONdition`.

Parameters:

<DataPatternTo>	Decimal representation of the data pattern Range: -100E+24 to 100E+24 Increment: 0.5 *RST: 1

TRIGger<m>:CAN:SYMBolic:DMIN <DataPattern>

Defines the minimum data value of the signal.

To set the condition, use `TRIGger<m>:CAN:DCONdition`.

Parameters:

<DataPattern> Decimal representation of the data pattern

Range: -100E+24 to 100E+24
Increment: 0.5
*RST: 0

TRIGger<m>:CAN:SYMBolic:SGEValue <SignalEnumValue>

Sets a symbolic data value for signals with enumerated values.

Parameters:

<SignalEnumValue> Numeric value according to the value definition in the DBC file

Example: Definition line in DBC file:

```
VAL_ 2175091489 Gear 0 "Idle" 1 "Gear_1" 2 "Gear_2" 3 "Gear_3"
4 "Gear_4" 5 "Gear_5" ;
```

Search for "Gear_4"

```
TRIGger:CAN:SYMBolic:SGEValue 4
```

Symbolic Decode Results

BUS<m>:CAN:FRAMe<n>:SDATa?

Returns the complete symbolic data of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData> String with comma-separated list of symbolic data

Example:

```
BUS:CAN:FRAMe9:SDATa?
<-- [sym] 325 kW, 0x0A, 423 N, 174 l, Running, 90 degC, 0x06, 437 rpm
```

Returns the symbolic results of the 9th frame.

Usage: Query only

Symbolic Search

SEARch:RESult:CAN:FRAMe<m>:SDATa? 1489
SEARch:TRIGger:CAN:SSYMbolic 1489
SEARch:TRIGger:CAN:SYMBolic:MSGValue 1489
SEARch:TRIGger:CAN:SYMBolic:SSIGnals 1490
SEARch:TRIGger:CAN:SYMBolic:SIGValue 1490
SEARch:TRIGger:CAN:SYMBolic:DMIN 1490
SEARch:TRIGger:CAN:SYMBolic:DMAX 1490
SEARch:TRIGger:CAN:SYMBolic:SGEValue 1491

SEARch:RESult:CAN:FRAMe<m>:SDATa? <SearchName>

Returns the symbolic data of the selected result frame.

Suffix:

<m> *

Sets the index of the search result frame.

Query parameters:

<SearchName> String that contains the search definition name

Return values:

<SymbolicData> String with comma-separated list of symbolic data

Example:

```
SEARch:RESult:CAN:FRAMe:SDATa? 'Search1'
<-- [sym] 325 kW, 0x0A, 423 N, 174 l, Running, 90 degC, 0x06, 437 rpm
```

Returns the symbolic results of the first search result.

Usage: Query only

SEARch:TRIGger:CAN:SSYMbolic <SearchName>,<CheckSymbolic>
SEARch:TRIGger:CAN:SSYMbolic? <SearchName>

Enables the symbolic search and disables all other search criteria.

Parameters:

<CheckSymbolic> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:MSGValue <SearchName>, <MessageName>
SEARch:TRIGger:CAN:SYMBolic:MSGValue? <SearchName>

Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

See also: `TRIGger<m>:CAN:SYMBolic:MSGValue` on page 1487

Parameters:

<MessageName> String that contains the symbolic message name

Parameters for setting and query:

<SearchName> String that contains the search definition name

Example:

```
SEARch:TRIGger:CAN:SYMBolic:MSGValue "Search1","EngineData"
```

SEARch:TRIGger:CAN:SYMBolic:SSIGnals <SearchName>,<SymbolicSearch>
SEARch:TRIGger:CAN:SYMBolic:SSIGnals? <SearchName>

Enables the search for symbolic values if DBC label list file is loaded and applied. Symbolic search disables all other search criteria.

Parameters:

<SymbolicSearch> ON | OFF
*RST: OFF

Parameters for setting and query:

<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:SIGValue <SearchName>, <SignalName>
SEARch:TRIGger:CAN:SYMBolic:SIGValue? <SearchName>

Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

See also: TRIGger<m>:CAN:SYMBolic:SIGValue on page 1487

Parameters:

<SignalName> String that contains the symbolic signal name

Parameters for setting and query:

<SearchName> String that contains the search definition name

Example: `SEARch:TRIGger:CAN:SYMBolic:SIGValue "Search1","EngForce"`

SEARch:TRIGger:CAN:SYMBolic:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:CAN:SYMBolic:DMIN? <SearchName>

Defines the minimum data pattern in a symbolic search.

To set the condition, use SEARch:TRIGger:CAN:DCONdition.

Parameters:

<DataPattern> Range: -100E+24 to 100E+24
Increment: 0.5
*RST: 0

Parameters for setting and query:

<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:CAN:SYMBolic:DMAX? <SearchName>

Defines the maximum data pattern of the signal in a symbolic search.

This value is required to specify a range if condition `INRange | OORange` is set with SEARch:TRIGger:CAN:DCONdition on page 1475.

Parameters:

<DataPatternTo>	Range: -100E+24 to 100E+24 Increment: 0.5 *RST: 1

Parameters for setting and query:

<SearchName>	String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:SGEValue <SearchName>, <SignalEnumValue>
SEARch:TRIGger:CAN:SYMBolic:SGEValue? <SearchName>

Sets a symbolic data value for signals with enumerated values.

Parameters:

<SignalEnumValue>	Numeric value according to the value definition in the DBC file

Parameters for setting and query:

<SearchName>	String that contains the search definition name

Example: Definition line in DBC file:

```
VAL_ 2175091489 Gear 0 "Idle" 1 "Gear_1" 2 "Gear_2"
3 "Gear_3" 4 "Gear_4" 5 "Gear_5" ;
```

Search for "Gear_3"

```
SEARch:TRIGger:CAN:SYMBolic:SGEValue "Search1",3
```

17.17.7 LIN (Option R&S RTE-K3)

- Configuration....1491
- Trigger....1493
- Decode Results....1500
- LIN Search Settings....1507
- LIN Search Results....1514

17.17.7.1 Configuration

BUS<m>:LIN:DATA:SOURce....1491
BUS<m>:LIN:DATA:THReshold....1492
BUS<m>:LIN:TECHnology....1492
BUS<m>:LIN:BITRate....1493
BUS<m>:LIN:POLarity....1493
BUS<m>:LIN:STANdard....1493

BUS<m>:LIN:DATA:SOURce <DataSource>

Sets the waveform of the data line.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

Usage: Asynchronous command

BUS<m>:LIN:DATA:THReshold <Threshold>

Sets a user-defined threshold value.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:LIN:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Threshold> Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:LIN:TECHnology <Technology>

Sets the threshold voltage as defined for various signal technologies.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Technology> V15 | V25 | V35 | V6 | V9 | MANual

V15
1.5 Volt (TTL)

V25
2.5 Volt (CMOS 5.0 V)

V35 | V6 | V9
3.5 V (7 V supply), 6.0 V (12 V supply), 9.0 V (18 V supply) respectively

MANual
Manual setting of user-defined values with `BUS<m>:LIN:DATA:THReshold`.

*RST: V35

BUS<m>:LIN:BITRate <Bitrate>

Sets the number of transmitted bits per second.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<Bitrate>	Range: 1000 to 20000 Increment: 1 *RST: 9600 Default unit: bps

BUS<m>:LIN:POLarity <Polarity>

Defines the idle state of the bus. The idle state is the rezessive state and corresponds to a logic 1.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<Polarity>	IDLLow | IDLHigh *RST: IDLHigh

BUS<m>:LIN:STANdard <Standard>

Selects the version of the LIN standard.

See also: "LIN standard" on page 565

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<Standard>	V1X | V2X | J2602 | AUTO *RST: AUTO

17.17.7.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:LIN:TYPE....1494
TRIGger<m>:LIN:ICONdition....1495
TRIGger<m>:LIN:IMIN....1495
TRIGger<m>:LIN:IMAX....1495
TRIGger<m>:LIN:DCONdition....1495
TRIGger<m>:LIN:DMIN....1496
TRIGger<m>:LIN:DMAX....1496
TRIGger<m>:LIN:BORDer....1496
TRIGger<m>:LIN:DLECondition....1497
TRIGger<m>:LIN:DLENgth....1497
TRIGger<m>:LIN:IDOR<n>:ENABle....1497
TRIGger<m>:LIN:IDOR<n>[:VALue]....1498
TRIGger<m>:LIN:SYERror....1498
TRIGger<m>:LIN:IPERror....1498
TRIGger<m>:LIN:CHKSerror....1498
TRIGger<m>:LIN:ERRPattern....1499
TRIGger<m>:LIN:CRCDatalen....1499
TRIGger<m>:LIN:STANdard....1499

TRIGger<m>:LIN:TYPE <Type>

Selects the trigger type for LIN analysis.

See: "Trigger type" on page 566.

Parameters:

<Type> SYNC | ID | IDOR | IDDT | WKFR | ERRC

SYNC
Start of the frame, triggers on the stop bit of the sync field.

ID
Sets the trigger to one specific identifier or an identifier range.
To set the identifier, use `TRIGger<m>:LIN:ICONdition`, `TRIGger<m>:LIN:IMIN` on page 1495, and `TRIGger<m>:LIN:IMAX` on page 1495.

IDOR
Triggers on an OR combination with up to four identifier conditions. For each identifier condition, enable it with `TRIGger<m>:LIN:IDOR<n>:ENABle` and set the value with `TRIGger<m>:LIN:IDOR<n>[:VALue]`

IDDT
Combination of identifier and data conditions
To set the identifier condition, use `TRIGger<m>:LIN:ICONdition`, `TRIGger<m>:LIN:IMIN`, and `TRIGger<m>:LIN:IMAX`.
To set the data condition, use `TRIGger<m>:LIN:DCONdition`, `TRIGger<m>:LIN:DMIN`, and `TRIGger<m>:LIN:DMAX`.

WKFR
Wakeup frame

ERRC
Error condition. Define the error types with TRIGger<m>:LIN:CHKSerror on page 1498, TRIGger<m>:LIN:IPERror, and TRIGger<m>:LIN:SYERror

*RST: SYNC

TRIGger<m>:LIN:ICONdition <IdOperator>

Sets the operator to set a specific identifier or an identifier range.

Parameters:

<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with TRIGger<m>:LIN:IMIN

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with TRIGger<m>:LIN:IMIN and TRIGger<m>:LIN:IMAX

*RST: EQUal

TRIGger<m>:LIN:IMIN <IdPattern>

Specifies a slave identifier pattern, or sets the the start value of an identifier range.

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:LIN:IMAX <IdPatternTo>

Sets the the end value of an identifier range if TRIGger<m>:LIN:ICONdition is set to INRange or OORange.

Parameters:

<IdPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:LIN:DCONdition <DataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:LIN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:LIN:DMIN` and `TRIGger<m>:LIN:DMAX`

*RST: EQUal

TRIGger<m>:LIN:DMIN <DataPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:LIN:DMAX <DataPatternTo>

Sets the the end value of an data range if `TRIGger<m>:LIN:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:LIN:BORDer <Endianness>

Sets the byte order (endianness) of the data transfer.

According to the standard, LIN data is transmitted in little endian transfer order.

Parameters:

<Endianness> BENDian | LENDian

BENDian
Big endian, data is analyzed and evaluated in the order of reception.

LENDian
Little endian, the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

*RST: BENDian

TRIGger<m>:LIN:DLECondition <DLCOperator>

Operator to set the data length for triggering on LIN data.

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.

Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.

The number of data bytes to be found is set with `TRIGger<m>:LIN:DLENgth` on page 1497.

See also:`TRIGger<m>:LIN:BORDer` on page 1496 .

Parameters:

<DLCOperator>	EQUal | GETHan For little endian transfer direction, EQUal must be set. *RST: GETHan

TRIGger<m>:LIN:DLENgth <WordCount>

Sets the length of the bit pattern to be found, in bytes. For "Big Endian" transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the beginning of the data pattern. For "Little Endian" transfer direction, the exact number of data bytes in the frame must be set.

For complete definition, set also the operator with `TRIGger<m>:LIN:DLECondition` on page 1497.

Parameters:

<WordCount>	Range: 1 to 8 Increment: 1 *RST: 1

TRIGger<m>:LIN:IDOR<n>:ENABle <UseIdentifier>

Includes the indicated IDOR address in the "identifier OR" trigger condition.

Suffix:

<n>	1..4 Index of the identifier in an "identifier OR" condition

Parameters:

<UseIdentifier>	ON | OFF *RST: OFF

TRIGger<m>:LIN:IDOR<n>[:VALue] <IdPattern>

Defines the pattern of the indicated IDOR identifier in the "identifier OR" trigger condition.

Suffix:

<n>	1..4 Index of the identifier in an "identifier OR" condition

Parameters:

<IdPattern>	Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The parameter accepts the bit value X (don't care).

TRIGger<m>:LIN:SYERror <SyncError>

Triggers if a synchronization error occurs.

The trigger type has to be set before: `TRIGger<m>:LIN:TYPE` to `ERRC`.

Parameters:

<SyncError>	ON | OFF *RST: ON

TRIGger<m>:LIN:IPERror <IdParityError>

Triggers if an error occurs in the identifier parity bits. These are the bits 6 and 7 of the identifier.

The trigger type has to be set before: `TRIGger<m>:LIN:TYPE` to `ERRC`.

Parameters:

<IdParityError>	ON | OFF *RST: ON

TRIGger<m>:LIN:CHKSerror <ChecksumError>

Triggers on checksum errors according to the LIN standard set with `BUS<m>:LIN:STANdard`.

The trigger type has to be set before: `TRIGger<m>:LIN:TYPE` to `ERRC`.

The frame identifier must be set with `TRIGger<m>:LIN:ERRPattern` on page 1499 and the data length with `TRIGger<m>:LIN:CRCDatalen` on page 1499.

Parameters:

<ChecksumError>	ON | OFF *RST: ON

TRIGger<m>:LIN:ERRPattern <ErrorPattern>

Sets the frame identifier to trigger on a checksum error with `TRIGger<m>:LIN:CHKSerror` on page 1498.

Parameters:

<ErrorPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.
Possible values depend on `TRIGger<m>:LIN:CRCDatalen`.
Defining don't care bits 'X' in the ERRP bit string resets CRCDatalen to 0. When CRCDatalen is different than 0, then all the bits in ERRP must be 1 or 0, and X bits are set to 0.

Example:

```
FORM:BPAT STRG
TRIG1:LIN:ERRP '1X0'
TRIG1:LIN:ERRP?
<-- 1X0XXX
TRIGger:LIN:CRCDatalen?
<-- 0
TRIGger:LIN:CRCDatalen 4
TRIG1:LIN:ERRP?
<-- 100000
TRIG1:LIN:ERRP '00x11'
TRIG1:LIN:ERRP?
<-- 00X11X
TRIGger:LIN:CRCDatalen?
<-- 0
```

TRIGger<m>:LIN:CRCDatalen <CRCDataLength>

Sets the number of data bytes to trigger on CRC errors (`TRIGger<m>:LIN:TYPE` is set to ERRC and `TRIGger<m>:LIN:CHKSerror` is set ON.)

Parameters:

<CRCDataLength> Values ≠0 restrict allowed bit values in `TRIGger<m>:LIN:ERRPattern` to 0 and 1.

Range: 0 to 8
Increment: 1
*RST: 0

TRIGger<m>:LIN:STANdard <LINStandard>

Sets the LIN standard to trigger on CRC errors (`TRIGger<m>:LIN:TYPE` is set to ERRC and `TRIGger<m>:LIN:CHKSerror` is set ON.)

See also: "LIN standard" on page 565.

Parameters:

<LINStandard> V1X | V2X | J2602 | AUTO

*RST: AUTO

17.17.7.3 Decode Results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1384
- `BUS<m>:SYMBols` on page 1385

BUS<m>:LIN:FCOunt?...1500
BUS<m>:LIN:FRAMe<n>:STATus?..1500
BUS<m>:LIN:FRAMe<n>:STARt?...1501
BUS<m>:LIN:FRAMe<n>:STOP?..1501
BUS<m>:LIN:FRAMe<n>:SDATa?...1501
BUS<m>:LIN:FRAMe<n>:SDEXport?..1502
BUS<m>:LIN:FRAMe<n>:SYMBol?..1502
BUS<m>:LIN:FRAMe<n>:VERSion?...1502
BUS<m>:LIN:FRAMe<n>:DATA?..1503
BUS<m>:LIN:FRAMe<n>:IDSTate?...1503
BUS<m>:LIN:FRAMe<n>:IDValue?...1503
BUS<m>:LIN:FRAMe<n>:IDPValue?..1504
BUS<m>:LIN:FRAMe<n>:SYSTate?...1504
BUS<m>:LIN:FRAMe<n>:CSSTate?...1505
BUS<m>:LIN:FRAMe<n>:CSValue?...1505
BUS<m>:LIN:FRAMe<n>:BITRate?...1505
BUS<m>:LIN:FRAMe<n>:BYTE<o>:STATe?...................................1506
BUS<m>:LIN:FRAMe<n>:BYTE<o>:VALue?...................................1506

BUS<m>:LIN:FCOunt?

Returns the number of decoded frames.

Suffix:

<m>	1..4 Selects the serial bus.

Return values:

<Count>	Total number of decoded frames.

Usage: Query only

BUS<m>:LIN:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<FrameState>	OK | UART | CHCKsum | VERS | LENer | SPERror | PRERror | SYERror | WAKeup | CPERror | INSufficient | INComplete

UART: at least one UART error occured. LIN uses UART words without parity bit.
CHCKsum: checksum error
VERS: the version of the LIN standard is not valid
LENer: unexpected length
SPERror: stop error
PRERror: parity error in identifier
SYERror: synchronization error
WAKeup: the frame is a wakeup frame
CPERror: parity error and checksum error
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.
INComplete: the frame is missing some parts

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:STARt?
BUS<m>:LIN:FRAMe<n>:STOP?

Returns the start time and stop time of the selected frame, respectively.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Start>, <Stop> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SDATa?

Returns the complete symbolic data of the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SDEXport?

Returns the symbolic data of the frame in export format.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Label> String with symbolic name of the identifier

Example: `BUS:LIN:FRAMe2:SYMBol?`
Response: `Temperature`

Usage: Query only

BUS<m>:LIN:FRAMe<n>:VERSion?

Returns the version of the LIN standard for the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameVersion> V1X | V2X | UNK
UNK: Unknown
*RST: UNK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:DATA?

Returns the data bytes of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data> Comma-separated list of integer values (N, D1, D2,…, DN). N is the number of bytes in the frame, and D1…DN are the values of the bytes.

Example:
```
BUS:LIN:FRAMe4:DATA?
<-- 4,118,39,71,123
```

Usage: Query only

BUS<m>:LIN:FRAMe<n>:IDSTate?

Returns the identifier state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

STERror: start error
SPERror: stop error
PRERror: parity error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:IDValue?

Returns the identifier value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<IdentifierValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.
Range: 0 to 63
*RST: 0

Usage: Query only

BUS<m>:LIN:FRAMe<n>:IDPValue?

Returns the value of the identifier parity bits of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<IdtfParityValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.
Range: 0 to 3
*RST: 0

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SYSTate?

Returns the state of the sync field for the specified frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<SyncState> OK | STERror | SPERror | UVAL | NOEXists | INSufficient
STERror: start error
SPERror: stop error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.
*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:CSSTate?

Returns the checksum state of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumState> OK | STERror | SPERror | UVAL | NOEXists | INSufficient

STERror: start error
SPERror: stop error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:CSValue?

Returns the checksum value of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
*RST: 0

Usage: Query only

BUS<m>:LIN:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:LIN:FRAMe<n>:BYTE<o>:STATe?

Returns the state of the specified byte.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<ByteState> OK | STERror | SPERror | UVAL | NOEXists | INSufficient

STERror: start error
SPERror: stop error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:BYTE<o>:VALue?

Returns the value of the specified byte.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte.

Return values:

<ByteValue> To set the value format, use FORMat:BPATtern.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
*RST: 0

Usage: Query only

17.17.7.4 LIN Search Settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to LIN trigger commands.

SEARch:TRIGger:LIN:SSOFrame....1507
SEARch:TRIGger:LIN:SFIDentifier....1508
SEARch:TRIGger:LIN:IDENtifieror....1508
SEARch:TRIGger:LIN:SIDData....1508
SEARch:TRIGger:LIN:SERRor....1508
SEARch:TRIGger:LIN:WUFRame....1509
SEARch:TRIGger:LIN:ICONdition....1509
SEARch:TRIGger:LIN:IMIN....1509
SEARch:TRIGger:LIN:IMAX....1509
SEARch:TRIGger:LIN:IDOR<m>:ENABle....1510
SEARch:TRIGger:LIN:IDOR<m>[:VALue]....1510
SEARch:TRIGger:LIN:DCONdition....1510
SEARch:TRIGger:LIN:DMIN....1511
SEARch:TRIGger:LIN:DMAX....1511
SEARch:TRIGger:LIN:BORDer....1511
SEARch:TRIGger:LIN:DLECondition....1512
SEARch:TRIGger:LIN:DLENgth....1512
SEARch:TRIGger:LIN:IPERror....1512
SEARch:TRIGger:LIN:SYERror....1513
SEARch:TRIGger:LIN:CHKSerror....1513
SEARch:TRIGger:LIN:ERRPattern....1513
SEARch:TRIGger:LIN:CRCDatalen....1514
SEARch:TRIGger:LIN:STANdard....1514

SEARch:TRIGger:LIN:SSOFrame <SearchName>,<FrameStart>
SEARch:TRIGger:LIN:SSOFrame? <SearchName>

Enables the search for the stop bit of the sync field, which marks the frame start.

Parameters:

<FrameStart> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:SFIDentifier <SearchName>,<Identifier>
SEARch:TRIGger:LIN:SFIDentifier? <SearchName>

Enables the search for one specific identifier or an identifier range.

Parameters:
<Identifier> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IDENtifieror <SearchName>,<IdentifierOr>
SEARch:TRIGger:LIN:IDENtifieror? <SearchName>

Enables the search for one to four address conditions.

Parameters:
<IdentifierOr> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SIDData <SearchName>,<IdentifierData>
SEARch:TRIGger:LIN:SIDData? <SearchName>

Enables the search for a combination of identifier and data conditions.

Parameters:
<IdentifierData> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:LIN:SERRor? <SearchName>

Enables the search for various errors in the frame.

Parameters:
<ErrorCondition> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:WUFRame <SearchName>,<WakeUpFrame>
SEARch:TRIGger:LIN:WUFRame? <SearchName>

Enables the search for wakeup frames.

Parameters:

<WakeUpFrame> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:ICONdition <SearchName>,<IdOperator>
SEARch:TRIGger:LIN:ICONdition? <SearchName>

Sets the operator to define a specific identifier or an identifier range.

Parameters:

<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These operators require one identifier pattern to be set with `SEARch:TRIGger:LIN:IMIN`

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:LIN:IMIN` and `SEARch:TRIGger:LIN:IMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:IMIN <SearchName>,<IdPattern>
SEARch:TRIGger:LIN:IMIN? <SearchName>

Specifies a slave identifier pattern, or sets the the start value of an identifier range.

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:IMAX <SearchName>,<IdPatternTo>
SEARch:TRIGger:LIN:IMAX? <SearchName>

Sets the the end value of an identifier range if `SEARch:TRIGger:LIN:ICONdition` is set to `INRange` or `OORange`.

Parameters:
<IdPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IDOR<m>:ENABle <SearchName>,<UseIdentifier>
SEARch:TRIGger:LIN:IDOR<m>:ENABle? <SearchName>

Includes the indicated IDOR address in the "identifier OR" search.

Suffix:
<m> 1..4
Index of the identifier in an "identifier OR" condition

Parameters:
<UseIdentifier> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IDOR<m>[:VALue] <SearchName>,<IdPattern>
SEARch:TRIGger:LIN:IDOR<m>[:VALue]? <SearchName>

Defines the pattern of the indicated IDOR identifier in the "identifier OR" trigger condition.

Suffix:
<m> 1..4
Index of the identifier in an "identifier OR" condition

Parameters:
<IdPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:LIN:DCONdition? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:LIN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with SEARch:TRIGger:LIN:DMIN and SEARch:TRIGger:LIN:DMAX.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:LIN:DMIN? <SearchName>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:
<DataPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:LIN:DMAX? <SearchName>

Sets the the end value of an identifier range if SEARch:TRIGger:LIN:DCONdition is set to INRange or OORange.

Parameters:
<DataPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:BORDer <SearchName>,<Endianness>
SEARch:TRIGger:LIN:BORDer? <SearchName>

Sets the byte order (endianness) of the data transfer.

Parameters:
<Endianness> BENDian | LENDian

BENDian
Big endian, data is analyzed and evaluated in the order of reception.

LENDian
Little endian, the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

*RST: BENDian

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DLECondition <SearchName>,<DLCOperator>
SEARch:TRIGger:LIN:DLECondition? <SearchName>

Operator to set the data length for search on LIN data.

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.

Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.

The number of data bytes to be found is set with `SEARch:TRIGger:LIN:DLENgth`.

See also: `SEARch:TRIGger:LIN:BORDer` on page 1511.

Parameters:

<DLCOperator>	EQUal | GETHan	
	For little endian transfer direction, EQUal must be set.	
	*RST:	GETHan

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DLENgth <SearchName>,<WordCount>
SEARch:TRIGger:LIN:DLENgth? <SearchName>

Sets the length of the bit pattern to be found, in bytes.

For complete definition, set the operator using `SEARch:TRIGger:LIN:DLECondition`, and the transfer direction with `SEARch:TRIGger:LIN:BORDer`.

Parameters:

<WordCount>	Range:	1 to 8
	Increment:	1
	*RST:	1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IPERror <SearchName>,<IdParityError>
SEARch:TRIGger:LIN:IPERror? <SearchName>

Searches for errors in the identifier parity bits. These are the bits 6 and 7 of the identifier.

Parameters:
<IdParityError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SYERror <SearchName>,<SyncError>
SEARch:TRIGger:LIN:SYERror? <SearchName>

Searches for synchronization errors.

Parameters:
<SyncError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:CHKSerror <SearchName>,<ChecksumError>
SEARch:TRIGger:LIN:CHKSerror? <SearchName>

Searches for checksum errors according to the LIN standard.

Use the following commands to configure the checksum error search:
- `SEARch:TRIGger:LIN:ERRPattern` on page 1513
- `SEARch:TRIGger:LIN:CRCDatalen` on page 1514
- `SEARch:TRIGger:LIN:STANdard` on page 1514

Parameters:
<ChecksumError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:ERRPattern <SearchName>,<ErrorPattern>
SEARch:TRIGger:LIN:ERRPattern? <SearchName>

Sets the frame identifier to search for a checksum error.

Parameters:
<ErrorPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:CRCDatalen <SearchName>,<CRCDataLength>
SEARch:TRIGger:LIN:CRCDatalen? <SearchName>

Sets the number of data bytes search for CRC errors.

Parameters:

<CRCDataLength>	Range:	0 to 8
	Increment:	1
	*RST:	0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:STANdard <SearchName>,<LINStandard>
SEARch:TRIGger:LIN:STANdard? <SearchName>

Sets the LIN standard to search for CRC errors.

Parameters:

<LINStandard>	V1X | V2X | J2602 | AUTO	
	*RST:	AUTO

Parameters for setting and query:
<SearchName>

17.17.7.5 LIN Search Results

The search on decoded LIN data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.7.3, "Decode Results", on page 1500.

SEARch:RESult:LIN:FCOunt?.. 1515
SEARch:RESult:LIN:FRAMe<m>:STATus?.. 1515
SEARch:RESult:LIN:FRAMe<m>:STARt?.. 1515
SEARch:RESult:LIN:FRAMe<m>:STOP?.. 1515
SEARch:RESult:LIN:FRAMe<m>:DATA?.. 1516
SEARch:RESult:LIN:FRAMe<m>:CSSTate?.. 1516
SEARch:RESult:LIN:FRAMe<m>:CSValue?.. 1516
SEARch:RESult:LIN:FRAMe<m>:IDSTate?.. 1517
SEARch:RESult:LIN:FRAMe<m>:IDValue?.. 1517
SEARch:RESult:LIN:FRAMe<m>:IDPValue?.. 1517
SEARch:RESult:LIN:FRAMe<m>:SYMBol?.. 1518
SEARch:RESult:LIN:FRAMe<m>:SYSTate?.. 1518
SEARch:RESult:LIN:FRAMe<m>:VERSion?.. 1518
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe?.. 1519
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue?.. 1519

SEARch:RESult:LIN:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | UART | CHCKsum | VERS | LENer | SPERror | PRERror | SYERror | WAKeup | CPERror | INSufficient | INComplete

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data> Data bytes in the Frame

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:CSSTate? <SearchName>

Returns the status of the frame checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value
NOEXists: byte does not exist

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:CSValue? <SearchName>

Returns the checksum value.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<ChecksumValue>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:IDSTate? <SearchName>

Returns the status of the identifier.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value
NOEXists: byte does not exist

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:IDValue? <SearchName>

Returns the value of the identifier.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<IdentifierValue>	Range:	0 to 63
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:IDPValue? <SearchName>

Returns the value of the identifier parity bit.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<IdParityValue>	Range:	0 to 3
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Label>

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:SYSTate? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SyncState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:VERSion? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Version> V1X | V2X | UNK

*RST: UNK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteState>	OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient *RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteValue>	Range: 0 to 255 Increment: 1 *RST: 0

Usage: Query only

17.17.8 FlexRay (Option R&S RTE-K4)

- Configuration....................1519
- Trigger....................1524
- Decode Results....................1532
- Search Settings....................1538
- Search Results....................1548

17.17.8.1 Configuration

BUS<m>:FLXRay:SRCType....................1520
BUS<m>:FLXRay:SOURce<n>....................1520
BUS<m>:FLXRay:THReshold<n>....................1521
BUS<m>:FLXRay:THENable....................1521
BUS<m>:FLXRay:THData....................1521
BUS<m>:FLXRay:PRSingle....................1522
BUS<m>:FLXRay:PRDiff....................1522
BUS<m>:FLXRay:PRLogic....................1522

BUS<m>:FLXRay:POLarity....1523
BUS<m>:FLXRay:BITRate....1523
BUS<m>:FLXRay:CHTYpe....1523
BUS<m>:FLXRay:SEHB....1524

BUS<m>:FLXRay:SRCType <SourceType>

Sets the type of measurement.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SourceType> SINGle | DIFFerential | LOGic

SINGle
Used for measurements with single-ended probes or single-ended voltage measurements with differential probes on the FlexRay bus. Two thresholds have to be defined as absolute voltage levels, see `BUS<m>:FLXRay:THReshold<n>` on page 1521.

DIFFerential
Used for differential measurements on the FlexRay bus. This is the most common measurement. Two thresholds have to be defined as differential voltages.

LOGic
Used for measurements of the logic signal inside the FlexRay node, between the communication controller and the bus driver. It is possible to measure simultaneously on a data line and on the "enable" line. Each line requires its own threshold.

*RST: SINGle

BUS<m>:FLXRay:SOURce<n> <Sources>

Sets the input channel of the bus signal, or of the data and enable lines in case of a LOGic source type.

Suffix:

<m> 1..4
Selects the serial bus.

<n> 1 | 2
Selects the source: 1 = bus signal or data line, 2 = enable line

Parameters:

<Sources> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

C1W1 | C2W1 | C3W1 | C4W1
Always available

NONE
Only available for SOURce2 (enable line)

M1 | M2 | M3 | M4 | R1 | R2 | R3 | R4
Only available if the trigger source is one of the input channels but not the serial bus.

D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
Only available if BUS<m>:FLXRay:SOURce<n> is set to LOGic.

*RST: SOURce1: C1W1, SOURce2: C2W1

BUS<m>:FLXRay:THReshold<n> <THResholds>

Sets the thresholds for the bus signal if the source type is SINGle or DIFFerential.

For LOGic source type, use BUS<m>:FLXRay:THData on page 1521 and BUS<m>:FLXRay:THENable on page 1521.

Suffix:

<m> 1..4
Selects the serial bus.

<n> 1 | 2
1 = threshold high, 2 = threshold low

Parameters:

<THResholds> Differential or absolute voltage level, depending on the source type. See BUS<m>:FLXRay:SRCType on page 1520.

BUS<m>:FLXRay:THENable <ThresholdEnable>

Sets the threshold for the enable line if the source type is LOGic.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ThresholdEnable> Range: -12 to 12
Increment: 0.1
*RST: 2.65
Default unit: V

BUS<m>:FLXRay:THData <ThresholdData>

Sets the threshold for the data line if the source type is LOGic.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ThresholdData>
Range: -12 to 12
Increment: 0.1
*RST: 2.35
Default unit: V

BUS<m>:FLXRay:PRSingle <PresetSingleEnd>

Selects a default threshold voltage if BUS<m>:FLXRay:SRCType is set to SINGle.

Suffix:

<m>
1..4
Selects the serial bus.

Parameters:

<PresetSingleEnd>
MV150 | MV200 | MV250 | MV300 | MANual

MV150 | MV200 | MV250 | MV300
2.5 ± 0.15 V; 2.5 ± 0.2 V; 2.5 ± 0.25 V; 2.5 ± 0.3 V, respectively

MANual
Manual setting of user-defined values with BUS<m>:FLXRay:THResold<n> on page 1521.

*RST: MV150

BUS<m>:FLXRay:PRDiff <PresetDiff>

Selects a default threshold voltage if BUS<m>:FLXRay:SRCType is set to DIFFerential.

Suffix:

<m>
1..4
Selects the serial bus.

Parameters:

<PresetDiff>
MV150 | MV200 | MV250 | MV300 | MANual

MV150 | MV200 | MV250 | MV300
±150 mV, ±200 mV, ±250 mV, ±300 mV respectively

MANual
Manual setting of user-defined values with BUS<m>:FLXRay:THResold<n> on page 1521.

*RST: MV150

BUS<m>:FLXRay:PRLogic <PresetLogic>

Selects a default threshold voltage if BUS<m>:FLXRay:SRCType is set to LOGic.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<PresetLogic> V25 | V165 | V125 | V09 | V0 | MANual

V25 | V165 | V125 | V09 | V0
2.5 V (CMOS 5.0 V); 1.65 V (CMOS 3.5V), 1.25 V (CMOS 2.5V), 0.9 V (CMOS 1.8V), 0 V (ground)

MANual
Manual setting of user-defined values with `BUS<m>:FLXRay:THReshold<n>` on page 1521.

*RST: V25

BUS<m>:FLXRay:POLarity <Polarity>

Selects the wire on which the bus signal is measured in case of SINGle source type. The setting affects the digitization of the signal.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Polarity> BPLus | BMINus

*RST: BPLus

BUS<m>:FLXRay:BITRate <Bitrate>

Selects the number of transmitted bits per second.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Bitrate> M10 | M5 | M2_5

10, 5, or 2.5 Mbit/s.
The return value of 2.5 Mbit/s is M25.

*RST: M10

Example:

```
BUS:FLXRay:BITRate M2_5
BUS:FLXRay:BITRate?
M25
```

BUS<m>:FLXRay:CHTYpe <Channel>

Selects the channel on which the signal is measured. The setting is considered in the calculation of the frame CRC.

Suffix:

<m> 1..4

Selects the serial bus.

Parameters:

<Channel> CHA | CHB

Channel A or channel B

*RST: CHA

BUS<m>:FLXRay:SEHB <SeparateHdrBts>

The command affects the decoding and its display. If ON, the leading five indicator bits of the header are decoded as five single bits. Otherwise, the indicator bits are shown as one word with word length five bit.

Suffix:

<m> 1..4

Selects the serial bus.

Parameters:

<SeparateHdrBts> ON | OFF

*RST: OFF

17.17.8.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:FLXRay:TYPE....1525
TRIGger<m>:FLXRay:PLPReamble....1526
TRIGger<m>:FLXRay:NUFRame....1526
TRIGger<m>:FLXRay:SYFRame....1526
TRIGger<m>:FLXRay:STFRame....1526
TRIGger<m>:FLXRay:FCONdition....1527
TRIGger<m>:FLXRay:FMIN....1527
TRIGger<m>:FLXRay:FMAX....1527
TRIGger<m>:FLXRay:PCONdition....1528
TRIGger<m>:FLXRay:PMIN....1528
TRIGger<m>:FLXRay:PMAX....1528
TRIGger<m>:FLXRay:CENable....1528
TRIGger<m>:FLXRay:CMIN....1529
TRIGger<m>:FLXRay:CMAX....1529
TRIGger<m>:FLXRay:CSTep....1529
TRIGger<m>:FLXRay:DPOPerator....1530

TRIGger<m>:FLXRay:DPOSition....1530
TRIGger<m>:FLXRay:DPTO....1530
TRIGger<m>:FLXRay:DCONdition....1530
TRIGger<m>:FLXRay:DMIN....1531
TRIGger<m>:FLXRay:DMAX....1531
TRIGger<m>:FLXRay:SYMBol....1531
TRIGger<m>:FLXRay:BSSerror....1531
TRIGger<m>:FLXRay:FESerror....1532
TRIGger<m>:FLXRay:FSSerror....1532
TRIGger<m>:FLXRay:HCRCerror....1532
TRIGger<m>:FLXRay:PCRCerror....1532

TRIGger<m>:FLXRay:TYPE <Type>

Selects the trigger type for FlexRay analysis.

Parameters:

<Type> STOF | IDDT | SYMBol | ERRC

STOF
STart Of Frame: triggers on the first rising edge after the transmission start sequence (TSS).

IDDT
IDentifier and DaTa: triggers on the decoded frame content, on header and payload data.
For all settings that are not needed for the trigger condition, make sure to set its condition to `OFF`.
Indicator bits: see `TRIGger<m>:FLXRay:NUFRame`
Frame identifier: sets the trigger to one specific frame ID or an identifier range. To set the identifier, use `TRIGger<m>:FLXRay:FCONdition`, `TRIGger<m>:FLXRay:FMIN`, and `TRIGger<m>:FLXRay:FMAX`.
Payload length: trigger on the number of words in the payload segment. To set the payload length, use `TRIGger<m>:FLXRay:PCONdition`, `TRIGger<m>:FLXRay:PMIN`, and `TRIGger<m>:FLXRay:PMAX`.
Cycle count: trigger on the number of the current FlexRay cycle. To set the cycle count, use `TRIGger<m>:FLXRay:CENable`, `TRIGger<m>:FLXRay:CMIN`, `TRIGger<m>:FLXRay:CMAX`, and `TRIGger<m>:FLXRay:CSTep`.
Data position: sets the position of the data bit pattern within the payload segment. To set the data position, use `TRIGger<m>:FLXRay:DPOPerator`, `TRIGger<m>:FLXRay:DPOSition`, and `TRIGger<m>:FLXRay:DPTO`.
Data bit pattern: sets the data bit pattern to be found in the payload segment. The starting point of the pattern is defined by the data position. To set the bit pattern, use `TRIGger<m>:FLXRay:DCONdition`, `TRIGger<m>:FLXRay:DMIN`, and `TRIGger<m>:FLXRay:DMAX`.

SYMBol
Triggers on a symbol or wakeup pattern. Set the required symbol with TRIGger<m>:FLXRay:SYMBol

ERRC
ERRor Condition: triggers on one or more errors that are detected in the decoded data. Use TRIGger<m>:FLXRay:BSSerror, TRIGger<m>:FLXRay:FESerror, TRIGger<m>:FLXRay:FSSerror, and TRIGger<m>:FLXRay:PCRCerror.

*RST: STOF

TRIGger<m>:FLXRay:PLPReamble <PayloadPreamble>

Triggers on the payload preamble indicator bit that indicates a Network Management Vector in the payload segment.

Parameters:
<PayloadPreamble> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:NUFRame <NullFrame>

Triggers on the null frame indicator bit, a frame without usable data.

Parameters:
<NullFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:SYFRame <SyncFrame>

Triggers on the sync frame used for synchronization of the FlexRay system. Only sync nodes can send this frame type.

Parameters:
<SyncFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:STFRame <StartupFrame>

Triggers on startup frames used for startup of the network. Only specific start nodes can send this frame type.

Parameters:

<StartupFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:FCONdition <IdOperator>

Sets the operator to set a frame ID or a frame ID range.

Parameters:

<IdOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The frame ID is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one frame ID to be set with `TRIGger<m>:FLXRay:FMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:FMIN` and `TRIGger<m>:FLXRay:FMAX`.

*RST: EQUal

TRIGger<m>:FLXRay:FMIN <IdPattern>

Specifies a frame identifier pattern - the number of the slot - or sets the the start value of an identifier range.

Parameters:

IdPattern Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

TRIGger<m>:FLXRay:FMAX <IdPatternTo>

Sets the the end value of an identifier range if the condition `TRIGger<m>:FLXRay:FCONdition` is set to `INRange` or `OORange`.

Parameters:

IdPatternTo Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.
FMAX must be greater or equal than FMIN, and the position of the X bits are common to FMIN and FMAX.

TRIGger<m>:FLXRay:PCONdition <PayloadLengthOp>

Sets the operator for the payload length trigger setting. You can defined an exact value, or a range.

Parameters:

<PayloadLengthOp> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The payload length is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one payload length to be set with `TRIGger<m>:FLXRay:PMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:PMIN` and `TRIGger<m>:FLXRay:PMAX`.

*RST: OFF

TRIGger<m>:FLXRay:PMIN <PayloadLength>

Specifies a payload length - the number of words in the payload segment - or sets the the start value of an payload length range. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Parameters:

<PayloadLength> Range: 0 to 127
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:PMAX <PayloadLengthTo>

Sets the the end value of a payload length range if the condition `TRIGger<m>:FLXRay:PCONdition` is set to `INRange` or `OORange`.

Parameters:

<PayloadLengthTo> Range: 0 to 127
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:CENable <CycleCount>

Sets the operator to define a cycle count or a cycle count range.

Parameters:

<CycleCount> OFF | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF
The cycle count is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one cycle count number to be set with `TRIGger<m>:FLXRay:CMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:CMIN` and `TRIGger<m>:FLXRay:CMAX`.

*RST: OFF

TRIGger<m>:FLXRay:CMIN <CycleCountMin>

Specifies a cycle count - the number of the current FlexRay cycle - or sets the the start value of an cycle count range.

Parameters:

<CycleCountMin> Range: 0 to 63
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:CMAX <CycleCountMax>

Sets the the end value of a cycle count range if the condition `TRIGger<m>:FLXRay:CENable` on page 1528 is set to `INRange` or `OORange`.

Parameters:

<CycleCountMax> Range: 0 to 63
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:CSTep <CycleCountStep>

Specifies a step to trigger on each n-th cycle inside the given range. This allows for specific triggering if slot multiplexing is used.

The condition `TRIGger<m>:FLXRay:CENable` on page 1528 must be set to `INRange` or `OORange`.

Parameters:

<CycleCountStep> Range: 1 to 63
Increment: 1
*RST: 1

TRIGger<m>:FLXRay:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

OFF = ANY
The data position is not relevant for the trigger condition.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:FLXRay:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:DPOSition` and `TRIGger<m>:FLXRay:DPTO`.

*RST: EQUal

TRIGger<m>:FLXRay:DPOSition <DataPosition>

Sets the number of data bytes to be skipped after start of the payload segment

Parameters:

<DataPosition> Range: 0 to 255
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:DPTO <DataPositionTo>

Defines the last byte of interest, if the position operator `TRIGger<m>:FLXRay:DPOPerator` defines a range.

Parameters:

<DataPositionTo> Range: 0 to 255
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:DCONdition <DataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The data position is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:FLXRay:DMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:DMIN` and `TRIGger<m>:FLXRay:DMAX`.

*RST: EQUal

TRIGger<m>:FLXRay:DMIN <DataPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:FLXRay:DMAX <DataPatternTo>

Sets the the end value of an data range if the operator `TRIGger<m>:FLXRay:DCONdition` is set to INRange or OORange.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:FLXRay:SYMBol <Symbol>

Triggers on a symbol or on a wakeup pattern.

Parameters:

<Symbol> CASMts | WAKeup

CASMts
Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.

WAKeup
The wakeup pattern is sent to activate the nodes of the system.

*RST: CASMts

TRIGger<m>:FLXRay:BSSerror <BSSError>

Triggers on error in SyteStart Sequence. The BSS is transmitted before each byte.

Parameters:
<BSSError> ON | OFF
*RST: ON

TRIGger<m>:FLXRay:FESerror <FESError>

Triggers on error in Frame End Sequence. FES indicates the end of each frame.

Parameters:
<FESError> ON | OFF
*RST: ON

TRIGger<m>:FLXRay:FSSerror <FSSError>

Triggers on Error in a Frame Start Sequence. FSS follows the Transmission Start Sequence TSS at the beginning of each frame.

Parameters:
<FSSError> ON | OFF
*RST: ON

TRIGger<m>:FLXRay:HCRCerror <CRCHeaderError>

Triggers on error in the Cyclic Redundancy Check of the header data (mainly frame ID and payload length).

Parameters:
<CRCHeaderError> ON | OFF
*RST: ON

TRIGger<m>:FLXRay:PCRCerror <CRCPayloadError>

Triggers on error in the Cyclic Redundancy Check of the payload data.

Parameters:
<CRCPayloadError> ON | OFF
*RST: ON

17.17.8.3 Decode Results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1384
- `BUS<m>:SYMBols` on page 1385

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1383
- `BUS<m>:RESDetail` on page 1384

BUS<m>:FLXRay:FCOunt?............ 1533
BUS<m>:FLXRay:FRAMe<n>:STATus?............ 1533
BUS<m>:FLXRay:FRAMe<n>:STARt?............ 1534
BUS<m>:FLXRay:FRAMe<n>:STOP?............1534
BUS<m>:FLXRay:FRAMe<n>:SYMBol?............1534
BUS<m>:FLXRay:FRAMe<n>:TYPE?............1534
BUS<m>:FLXRay:FRAMe<n>:DATA?............ 1535
BUS<m>:FLXRay:FRAMe<n>:FLAGs?............ 1535
BUS<m>:FLXRay:FRAMe<n>:ADID?............1536
BUS<m>:FLXRay:FRAMe<n>:PAYLength?............ 1536
BUS<m>:FLXRay:FRAMe<n>:CYCount?............ 1536
BUS<m>:FLXRay:FRAMe<n>:CSSTate?............1537
BUS<m>:FLXRay:FRAMe<n>:CSValue?............1537
BUS<m>:FLXRay:FRAMe<n>:FCSTate?............1537
BUS<m>:FLXRay:FRAMe<n>:FCValue?............ 1538

BUS<m>:FLXRay:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Returns the number of decoded frames.

Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | FSS | BSS | FES | INDicator | HCRCerror | CRCerr | LENER | LENer | HCFCerror | INSufficient

OK: the frame is valid.
FSS: Frame Start Sequence after TSS is missing.
BSS: Byte Start Sequence is missing.
FES: error in the Frame End Sequence.
INDicator: Error in indicator bits.
HCRCerror: Header CRC is not valid.

CRCerr: Payload CRC is not valid.
LENER = LENer: Unexpected length of the frame.
HCFCerror: Header CRC error and frame CRC error
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:STARt?
BUS<m>:FLXRay:FRAMe<n>:STOP?

Return the start time and stop time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStop> Time

Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Label> String with symbolic name of the identifier

Example: `BUS:FLXRay:FRAMe2:SYMBol?`
Response: `Temperature`

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:TYPE?

Returns the frame type of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> UNKNown | STATic | DYNamic | WAKE | SYMBol

STATic: frame of the static segment
DYNamic: frame of the dynamic segment
WAKE: frame contains wakeup pattern
SYMBol: frame contains a MTS or CAS symbol

*RST: STATic

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:DATA?

Returns the data of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data> Comma-separated list of integer values (N, D1, D2,…, DN). N is the number of bytes in the frame, and D1…DN are the values of the bytes.

Example:

```
BUS:FLXRay:FRAMe4:DATA?
<-- 4,17,85,170,85
```

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:FLAGs?

Returns the value of the indicator bits at the beginning of the header segment. The five bits are read as one word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Flags> Range: 0 to 31
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:ADID?

Returns the frame identifier, the number of the slot in which the frame is transmitted.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddressID> To set the value format, use FORMat:BPATtern on page 1046.
The values below – range, increment and default – are decimal values.
Range: 0 to 2047
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:PAYLength?

Returns the payload length, the number of data words in the payload segment. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PayLen> Range: 0 to 127
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:CYCount?

Returns the number of the current FlexRay cycle.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<CycleCount> Range: 0 to 63
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:CSSTate?

Returns the state of the cyclic redundancy check code of the header data.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<HdrCksumSt> OK | UVAL | INSufficient

OK: the CRC is valid.
UVAL: unexpected value
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:CSValue?

Returns the checksum value of the header CRC.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<HdrCksumVal> To set the value format, use `FORMat:BPATtern` on page 1046.
The values below – range, increment and default – are decimal values.

Range: 0 to 2047
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:FCSTate?

Returns the state of the cyclic redundancy check code of the frame data.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumState> OK | UVAL | INSufficient

OK: the CRC is valid.
UVAL: unexpected value
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:FCValue?

Returns the cyclic redundancy check code of the frame CRC.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern` on page 1046.
The values below – range, increment and default – are decimal values.

Range: 0 to 16777215
*RST: 0

Usage: Query only

17.17.8.4 Search Settings

SEARch:TRIGger:FLXRay:SERRor ... 1539
SEARch:TRIGger:FLXRay:SIDData ... 1539
SEARch:TRIGger:FLXRay[:SSOFrame] ... 1539
SEARch:TRIGger:FLXRay:SSYMbol ... 1540
SEARch:TRIGger:FLXRay:CENable ... 1540
SEARch:TRIGger:FLXRay:CMAX ... 1540
SEARch:TRIGger:FLXRay:CMIN ... 1541
SEARch:TRIGger:FLXRay:CSTep ... 1541
SEARch:TRIGger:FLXRay:DCONdition ... 1541
SEARch:TRIGger:FLXRay:DMAX ... 1542
SEARch:TRIGger:FLXRay:DMIN ... 1542
SEARch:TRIGger:FLXRay:DPOPerator ... 1542
SEARch:TRIGger:FLXRay:DPOSition ... 1543

SEARch:TRIGger:FLXRay:DPTO... 1543
SEARch:TRIGger:FLXRay:FCONdition.. 1543
SEARch:TRIGger:FLXRay:FMAX.. 1544
SEARch:TRIGger:FLXRay:FMIN... 1544
SEARch:TRIGger:FLXRay:NUFRame.. 1544
SEARch:TRIGger:FLXRay:PLPReamble.. 1545
SEARch:TRIGger:FLXRay:PCONdition... 1545
SEARch:TRIGger:FLXRay:PMAX.. 1545
SEARch:TRIGger:FLXRay:PMIN.. 1546
SEARch:TRIGger:FLXRay:STFRame.. 1546
SEARch:TRIGger:FLXRay:SYFRame.. 1546
SEARch:TRIGger:FLXRay:SYMBol.. 1547
SEARch:TRIGger:FLXRay:BSSerror.. 1547
SEARch:TRIGger:FLXRay:FESerror.. 1547
SEARch:TRIGger:FLXRay:FSSerror.. 1547
SEARch:TRIGger:FLXRay:HCRCerror... 1548
SEARch:TRIGger:FLXRay:PCRCerror... 1548

SEARch:TRIGger:FLXRay:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:FLXRay:SERRor? <SearchName>

Enables the search for specified error or error combination.

Parameters:

<ErrorCondition> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:FLXRay:SIDData <SearchName>,<IdentifierData>
SEARch:TRIGger:FLXRay:SIDData? <SearchName>

Enables the search for indentifier and data.

Parameters:

<IdentifierData> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:FLXRay[:SSOFrame] <SearchName>,<FrameStart>
SEARch:TRIGger:FLXRay[:SSOFrame]? <SearchName>

Enables the search for a start of frame.

Parameters:

<FrameStart> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SSYMbol <SearchName>,<CheckSymbol>
SEARch:TRIGger:FLXRay:SSYMbol? <SearchName>

Enables the search for specified symbol.

Parameters:

<CheckSymbol> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CENable <SearchName>,<CycleCount>
SEARch:TRIGger:FLXRay:CENable? <SearchName>

Sets the operator to define a cycle count or a cycle count range.

Parameters:

<CycleCount> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The cycle count is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one cycle count number to be set with `SEARch:TRIGger:FLXRay:CMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:CMIN` and `SEARch:TRIGger:FLXRay:CMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CMAX <SearchName>,<CycleCountMax>
SEARch:TRIGger:FLXRay:CMAX? <SearchName>

Sets the the end value of a cycle count range if the condition `SEARch:TRIGger:FLXRay:CENable` is set to `INRange` or `OORange`.

Parameters:

<CycleCountMax> Range: 0 to 63
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CMIN <SearchName>,<CycleCountMin>
SEARch:TRIGger:FLXRay:CMIN? <SearchName>

Specifies a cycle count - the number of the current FlexRay cycle - or sets the the start value of an cycle count range.

Parameters:

<CycleCountMin>	Range:	0 to 63
	Increment:	1
	*RST:	0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CSTep <SearchName>,<CycleCountStep>
SEARch:TRIGger:FLXRay:CSTep? <SearchName>

Specifies a step to search for each n-th cycle inside the given range. This allows for a specific search if slot multiplexing is used.

The condition `SEARch:TRIGger:FLXRay:CENable` must be set to `INRange` or `OORange`.

Parameters:

<CycleCountStep>	Range:	1 to 63
	Increment:	1
	*RST:	1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:FLXRay:DCONdition? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The data pattern is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:FLXRay:DMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with SEARch:TRIGger:FLXRay:DMIN and SEARch:TRIGger:FLXRay:DMAX.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:FLXRay:DMAX? <SearchName>

Sets the the end value of an data range if SEARch:TRIGger:FLXRay:DCONdition is set to INRange or OORange.

Parameters:
<DataPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:FLXRay:DMIN? <SearchName>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:
<DataPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:FLXRay:DPOPerator? <SearchName>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:
<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

OFF = ANY
The data position is not relevant for the search condition.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with SEARch:TRIGger:FLXRay:DPOSition

INRange = RANGe
In range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:DPOSition` and `SEARch:TRIGger:FLXRay:DPTO`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:FLXRay:DPOSition? <SearchName>

Sets the number of data bytes to be skipped after start of the payload segment.

Parameters:

<DataPosition> Range: 0 to 255
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:FLXRay:DPTO? <SearchName>

Defines the last byte of interest, if the position operator `SEARch:TRIGger:FLXRay:DPOPerator` defines a range.

Parameters:

<DataPositionTo> Range: 0 to 255
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FCONdition <SearchName>,<IdOperator>
SEARch:TRIGger:FLXRay:FCONdition? <SearchName>

Sets the operator to set a frame ID or a frame ID range.

Parameters:

<IdOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The frame ID is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one frame ID to be set with `SEARch:TRIGger:FLXRay:FMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with SEARch:TRIGger:FLXRay:FMIN and SEARch:TRIGger:FLXRay:FMAX.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FMAX <SearchName>,<IdPatternTo>
SEARch:TRIGger:FLXRay:FMAX? <SearchName>

Sets the the end value of an identifier range if the condition SEARch:TRIGger:FLXRay:FCONdition is set to INRange or OORange.

Parameters:
<IdPatternTo> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FMIN <SearchName>,<IdPattern>
SEARch:TRIGger:FLXRay:FMIN? <SearchName>

Specifies a frame identifier pattern - the number of the slot - or sets the the start value of an identifier range.

Parameters:
<IdPattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:NUFRame <SearchName>,<NullFrame>
SEARch:TRIGger:FLXRay:NUFRame? <SearchName>

Searches for the null frame indicator bit, a frame without usable data.

Parameters:
<NullFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PLPReamble <SearchName>,<PayloadPreamble>
SEARch:TRIGger:FLXRay:PLPReamble? <SearchName>

Searches for the payload preamble indicator bit that indicates a Network Management Vector in the payload segment.

Parameters:

<PayloadPreamble> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:FLXRay:PCONdition <SearchName>,<PayloadLengthOp>
SEARch:TRIGger:FLXRay:PCONdition? <SearchName>

Sets the operator for the payload length search setting. You can defined an exact value, or a range.

Parameters:

<PayloadLengthOp> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The payload length is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one payload length to be set with `SEARch:TRIGger:FLXRay:PMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:PMIN` and `SEARch:TRIGger:FLXRay:PMAX`.

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:FLXRay:PMAX <SearchName>,<PayloadLengthTo>
SEARch:TRIGger:FLXRay:PMAX? <SearchName>

Sets the the end value of a payload length range if the condition `SEARch:TRIGger:FLXRay:PCONdition` is set to `INRange` or `OORange`.

Parameters:

<PayloadLengthTo> Range: 0 to 127
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PMIN <SearchName>,<PayloadLength>
SEARch:TRIGger:FLXRay:PMIN? <SearchName>

Specifies a payload length - the number of words in the payload segment - or sets the the start value of an payload length range. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Parameters:

<PayloadLength>	Range:	0 to 127
	Increment:	1
	*RST:	0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:STFRame <SearchName>,<StartupFrame>
SEARch:TRIGger:FLXRay:STFRame? <SearchName>

Searches for startup frames used for startup of the network. Only specific start nodes can send this frame type.

Parameters:
<StartupFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SYFRame <SearchName>,<SyncFrame>
SEARch:TRIGger:FLXRay:SYFRame? <SearchName>

Searches for the sync frame used for synchronization of the FlexRay system. Only sync nodes can send this frame type.

Parameters:
<SyncFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SYMBol <SearchName>,<Symbol>
SEARch:TRIGger:FLXRay:SYMBol? <SearchName>

Searches for a symbol or for a wakeup pattern.

Parameters:

<Symbol> CASMts | WAKeup

CASMts
Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.

WAKeup
The wakeup pattern is sent to activate the nodes of the system.

*RST: CASMts

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:BSSerror <SearchName>,<BSSError>
SEARch:TRIGger:FLXRay:BSSerror? <SearchName>

Searches for error in SyteStart Sequence. The BSS is transmitted before each byte.

Parameters:

<BSSError> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FESerror <SearchName>,<FESError>
SEARch:TRIGger:FLXRay:FESerror? <SearchName>

Searches for error in Frame End Sequence. FES indicates the end of each frame.

Parameters:

<FESError> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FSSerror <SearchName>,<FSSError>
SEARch:TRIGger:FLXRay:FSSerror? <SearchName>

Searches for an error in a Frame Start Sequence(FSS). FSS follows the Transmission Start Sequence (TSS) at the beginning of each frame.

Parameters:
<FSSError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:HCRCerror <SearchName>,<CRCHeaderError>
SEARch:TRIGger:FLXRay:HCRCerror? <SearchName>

Searches for an error in the Cyclic Redundancy Check of the header data (mainly frame ID and payload length).

Parameters:
<CRCHeaderError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PCRCerror <SearchName>,<CRCPayloadError>
SEARch:TRIGger:FLXRay:PCRCerror? <SearchName>

Searches for error in the Cyclic Redundancy Check of the payload data.

Parameters:
<CRCPayloadError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

17.17.8.5 Search Results

The search on decoded FlexRay data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.8.3, "Decode Results", on page 1532.

SEARch:RESult:FLXRay:FCOunt?...1549
SEARch:RESult:FLXRay:FRAMe<m>:ADID?...1549
SEARch:RESult:FLXRay:FRAMe<m>:CSSTate?...1549
SEARch:RESult:FLXRay:FRAMe<m>:CSValue?...1550
SEARch:RESult:FLXRay:FRAMe<m>:CYCount?...1550
SEARch:RESult:FLXRay:FRAMe<m>:DATA?...1550
SEARch:RESult:FLXRay:FRAMe<m>:FCSTate?...1550
SEARch:RESult:FLXRay:FRAMe<m>:FCValue?...1551
SEARch:RESult:FLXRay:FRAMe<m>:FLAGs?...1551

SEARch:RESult:FLXRay:FRAMe<m>:PAYLength?.. 1551
SEARch:RESult:FLXRay:FRAMe<m>:STATus?... 1552
SEARch:RESult:FLXRay:FRAMe<m>:STARt?.. 1552
SEARch:RESult:FLXRay:FRAMe<m>:STOP?..1553
SEARch:RESult:FLXRay:FRAMe<m>:TYPE?..1553
SEARch:RESult:FLXRay:FRAMe<m>:SYMBol?..1553

SEARch:RESult:FLXRay:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:ADID? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<AddressID>	Range:	0 to 2047
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:CSSTate? <SearchName>

Returns the status of the header checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<HeadCksumState> OK | STERror | SPERror | PRERror | UVAL | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value

*RST: OK

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:CSValue? <SearchName>

Returns the value of the header checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<HeadCksumValue> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:CYCount? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CycleCount> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:FCSTate? <SearchName>

Returns the status of the frame checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<ChecksumState> OK | STERror | SPERror | PRERror | UVAL | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value

*RST: OK

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:FCValue? <SearchName>

Returns the value of the frame checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<ChecksumValue> Range: 0 to 16777215
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:FLAGs? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Flags> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:PAYLength? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<PayLen> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:STATus? <SearchName>

Returns the overall state of the selected frame.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<State> OK | FSS | BSS | FES | INDicator | HCRCerror | CRCerr | LENER | LENer | HCFCerror | INSufficient

OK: the frame is valid.
FSS: Frame Start Sequence after TSS is missing.
BSS: Byte Start Sequence is missing.
FES: error in the Frame End Sequence.
INDicator: Error in indicator bits.
HCRCerror: Header CRC is not valid.
CRCerr: Payload CRC is not valid.
LENER = LENer: Unexpected length of the frame.
HCFCerror: Header CRC error and frame CRC error
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:STARt? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> UNKNown | STATic | DYNamic | WAKE | SYMBol
*RST: STATic

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:SYMBol? <SearchName>

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:
<m> *
Selects the number of the frame in the current acquisition, 1...n.

Query parameters:
<SearchName> String parameter that contains the search definition name

Return values:
<Translation> Symbolic label (string)

Usage: Query only

17.17.9 Audio Signals (Option R&S RTE-K5)

- Configuration.....1554
- Trigger.....1560
- Decode Results.....1563
- Track and Trend.....1566

17.17.9.1 Configuration

BUS<m>:I2S:AVARiant....1554
BUS<m>:I2S:CLOCk:SOURce....1554
BUS<m>:I2S:CLOCk:POLarity....1555
BUS<m>:I2S:WSELect:SOURce....1555
BUS<m>:I2S:WSELect:POLarity....1555
BUS<m>:I2S:DATA:SOURce....1556
BUS<m>:I2S:DATA:POLarity....1556
BUS<m>:I2S:TCOupling....1556
BUS<m>:I2S:CLOCk:THReshold....1557
BUS<m>:I2S:WSELect:THReshold....1557
BUS<m>:I2S:DATA:THReshold....1557
BUS<m>:I2S:CHANnel:ORDer....1557
BUS<m>:I2S:WLENgth....1558
BUS<m>:I2S:BORDer....1558
BUS<m>:I2S:CHANnel:OFFSet....1558
BUS<m>:I2S:CHANnel:TDMCount....1559
BUS<m>:I2S:FOFFset....1559
BUS<m>:I2S:CHANnel:LENGth....1559

BUS<m>:I2S:AVARiant <AudioVariant>

Selects the audio signal type.

For details, see "Audio Variant" on page 601.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<AudioVariant> I2S | LJ | RJ | TDM

I2S: Inter-IC Sound standard audio format.
LJ: left-justified data format
RJ: right-justified data format
TDM: Time Division Multiplexed audio format

*RST: I2S

BUS<m>:I2S:CLOCk:SOURce <ClockSource>

Selects the source of the clock line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:
<ClockSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital channels require installation of R&S RTE-B1. Digital and analog channels cannot be used at the same time. For triggering on a serial bus, analog or digital input channels are required.

*RST: C1W1

BUS<m>:I2S:CLOCk:POLarity <BitClockEdge>

Sets the polarity of the clock signal, that is the edge at which the instrument samples the data on the data line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<BitClockEdge> FALLing | RISing

*RST: RISing

BUS<m>:I2S:WSELect:SOURce <WSsource>

Selects the source of the word select line for I²S standard, left- und right-justified data formats, or the source of the frame synchronization pulse for TDM audio signals.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<WSsource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital channels require installation of R&S RTE-B1. Digital and analog channels cannot be used at the same time. For triggering on a serial bus, analog or digital input channels are required.

*RST: C2W1

BUS<m>:I2S:WSELect:POLarity <WSPolarity>

For a word select line, the polarity defines the signal values assigned to the left and right channels.

For an FSYNC line (TDM), the polarity defines the edge of the FSYNC pulse that identifies the beginning of a frame.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<WSPolarity> NORMal | INVert

NORMal: 0 = left, 1 = right channel; or rising edge for TDM
INVert: 1= left, 0 = right channel; or falling edge for TDM

*RST: NORMal

BUS<m>:I2S:DATA:SOURce <DataSource>

Selects the source of the audio data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital channels require installation of R&S RTE-B1.Digital and analog channels cannot be used at the same time. For triggering on a serial bus, analog or digital input channels are required.

*RST: C3W1

BUS<m>:I2S:DATA:POLarity <SDataPolarity>

Defines the interpretation of high and low signal states on the data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SDataPolarity> ACTLow | ACTHigh

ACTHigh: HIGH = 1 and LOW = 0
ACTLow: HIGH = 0 and LOW = 1

*RST: ACTHigh

BUS<m>:I2S:TCOupling <Coupling>

Sets all thresholds to the value of the clock threshold `BUS<m>:I2S:CLOCk:THReshold`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:
<Coupling> ON | OFF
*RST: ON

BUS<m>:I2S:CLOCk:THReshold <SCLKThreshold>

Sets the threshold value for the clock line SCLK.

If `BUS<m>:I2S:TCOupling` is `ON`, the command sets the threshold for all lines.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SCLKThreshold> Range: -10 to 10
Increment: 1E-3
*RST: 1.6
Default unit: V

BUS<m>:I2S:WSELect:THReshold <WSThreshold>

Sets the threshold value for the word select and FSYNC lines.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<WSThreshold> Range: -10 to 10
Increment: 1E-3
*RST: 1.6
Default unit: V

BUS<m>:I2S:DATA:THReshold <SDATAThreshold>

Sets the threshold value for the data line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SDATAThreshold> Range: -10 to 10
Increment: 1E-3
*RST: 1.6
Default unit: V

BUS<m>:I2S:CHANnel:ORDer <ChannelOrder>

Defines if the left or the right channel is the first channel in the frame.

The setting is not available for TDM audio signals.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ChannelOrder> LFIRst | RFIRst

Left channel first or right first

*RST: LFIRst

BUS<m>:I2S:WLENgth <WordLength>

Defines the number of bits in an audio data word.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<WordLength> Range: 4 to 32
Increment: 4
*RST: 8
Default unit: bit

BUS<m>:I2S:BORDer <BitOrder>

Sets the bit order of the audio data words.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<BitOrder> LSBF | MSBF

LSB first or MSB first

*RST: MSBF

BUS<m>:I2S:CHANnel:OFFSet <ChannelOffset>

Sets the number of bits between the channel start and the start of the audio word.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ChannelOffset> Range: 0 to 32 (left-justified). TDM: maximum delay is Channel length - Word length
Increment: 1
*RST: 0
Default unit: bit

BUS<m>:I2S:CHANnel:TDMCount <ChannelsTDM>

Sets the number of channels transmitted on the TDM audio line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ChannelsTDM> Range: 1 to 8
Increment: 1
*RST: 1

BUS<m>:I2S:FOFFset <FrameOffsetTDM>

Sets the number of bits between the frame start and the start of the first channel of a TDM audio line. Each FSYNC edge restarts the offset count.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<FrameOffsetTDM> Range: 0 to 256
Increment: 1
*RST: 0
Default unit: bit

BUS<m>:I2S:CHANnel:LENGth <ChlgthTDM>

Sets the number of bits in a TDM channel block.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ChlgthTDM> Range: 4 to 32
Increment: 4
*RST: 8
Default unit: bit

17.17.9.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:I2S:TYPE... 1560
TRIGger<m>:I2S:TCONdition<n>:CHANnel..1561
TRIGger<m>:I2S:TCONdition<n>:CONDtion...1561
TRIGger<m>:I2S:TCONdition<n>:DMIN...1562
TRIGger<m>:I2S:TCONdition<n>:DMAX...1562
TRIGger<m>:I2S:SOWords..1563
TRIGger<m>:I2S:WSSLope... 1563

TRIGger<m>:I2S:TYPE <Type>

Selects the trigger type for audio signal analysis.

Parameters:

<Type> DATA | WINDow | CONDition | WSELect | ECONdition

DATA
Triggers on a data word or data range on a specified channel or on any channel.
To set the channel, use `TRIGger<m>:I2S:TCONdition<n>:CHANnel`.
To set the data condition, use:
`TRIGger<m>:I2S:TCONdition<n>:CONDtion`,
`TRIGger<m>:I2S:TCONdition<n>:DMIN` and `TRIGger<m>:I2S:TCONdition<n>:DMAX`.

WINDow
Triggers if the decoded data values stay inside a "window" that is formed by a data range and a time specified by a number of subsequent words. It considers a selected channel or all channels.
To set up a window trigger, you define the channel and data condition in the same way as for DATA trigger type. Additionally, you set the time limit with `TRIGger<m>:I2S:SOWords`.

CONDition
The frame condition trigger sets the trigger on an AND combination of data conditions on different channels. The instrument triggers if all conditions are met inside one frame.
To set up a CONDition trigger, you define up to four channel and data conditions in the same way as for DATA trigger type.

WSELect
WordSELect: Triggers on the selected edge of the WS line (I²S standard, left- and right-justified). For TDM signals, it triggers on the selected edge of the FSYNC line. Set the edge with `TRIGger<m>:I2S:WSSLope`.

ECONdition
ErrorCONdition: Triggers on irregularities between the WS or FSYNC edges.

*RST: DATA

Usage: Asynchronous command

TRIGger<m>:I2S:TCONdition<n>:CHANnel <Channel>

Selects the audio channel on which the instrument looks for the specified data condition.

Suffix:

<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:

<Channel> ANY | TDMC1 | TDMCh1 | TDMC2 | TDMCh2 | TDMC3 | TDMCh3 | TDMC4 | TDMCh4 | TDMC5 | TDMCh5 | TDMC6 | TDMCh6 | TDMC7 | TDMCh7 | TDMC8 | TDMCh8 | LEFT | RIGHT | RIGHt

ANY
The instrument triggers on any channel on which the specified data is found.

LEFT | RIGHT = RIGHt
Available for I²S Standard, left- und right-justified data formats.

TDMCh1 | TDMCh2 | TDMCh3 | TDMCh4 | TDMCh5 | TDMCh6 | TDMCh7 | TDMCh8
Available for TDM audio signals
TDMC1 = TDMCh1, TDMC2 = TDMCh2, TDMC3 = TDMCh3, TDMC4 = TDMCh4, TDMC5 = TDMCh5, TDMC6 = TDMCh6, TDMC7 = TDMCh7, TDMC8 = TDMCh8. Query returns short form.
Note: Available audio channels depend on the configuration of the audio bus. The command `BUS<m>:I2S:CHANnel:TDMCount` specifies the number of channels in a TDM frame.

*RST: ANY

TRIGger<m>:I2S:TCONdition<n>:CONDtion <DataCondition>

Sets the operator to set a specific data pattern or a data pattern range.

Suffix:
<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:
<DataCondition> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
No range is defined.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:I2S:TCONdition<n>:DMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:I2S:TCONdition<n>:DMIN` and `TRIGger<m>:I2S:TCONdition<n>:DMAX`.

*RST: OFF

TRIGger<m>:I2S:TCONdition<n>:DMIN <DataMinPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Suffix:
<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:
<DataMinPattern> Numeric pattern in 2's complement format. See also: Chapter 17.4.6, "Bit Pattern Parameter", on page 1039
X bits are not allowed. If the bit string is shorter than the word length, the rightmost bit of the input bit string is aligned to the rightmost (LSB) bit of the word.

TRIGger<m>:I2S:TCONdition<n>:DMAX <DataMaxPattern>

Sets the the end value of an data range if the operator `TRIGger<m>:I2S:TCONdition<n>:CONDtion` is set to `INRange` or `OORange`.

Suffix:

<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:

<DataMaxPattern> Numeric pattern in 2's complement format. See also: Chapter 17.4.6, "Bit Pattern Parameter", on page 1039
DMAX must be greater or equal than DMIN.
X bits are not allowed. If the bit string is shorter than the word length, the rightmost bit of the input bit string is aligned to the rightmost (LSB) bit of the word.

TRIGger<m>:I2S:SOWords <SequenceOfWords>

Sets the number of words that is used as time limit for the "Window" trigger type. The instrument triggers if the data condition is fulfilled on the same channel for the given number of subsequent frames.

Parameters:

<SequenceOfWords> Range: 1 to 1000000
Increment: 1
*RST: 1
Default unit: word

Usage: Asynchronous command

TRIGger<m>:I2S:WSSLope <WSSlope>

Sets the edge of the WS or FSYNC signal as trigger condition. The instrument triggers on the first clock edge after the specified edge.

Parameters:

<WSSlope> POSitive | NEGative
*RST: POSitive

Usage: Asynchronous command

17.17.9.3 Decode Results

BUS<m>:I2S:FCOunt? ... 1564
BUS<m>:I2S:FRAMe<n>:STATe? ... 1564
BUS<m>:I2S:FRAMe<n>:STARt? ... 1564
BUS<m>:I2S:FRAMe<n>:STOP? ... 1564
BUS<m>:I2S:FRAMe<n>:LEFT:VALue? ... 1565
BUS<m>:I2S:FRAMe<n>:RIGHt:VALue? ... 1565
BUS<m>:I2S:FRAMe<n>:LEFT:STATe? ... 1565

BUS<m>:I2S:FRAMe<n>:RIGHt:STATe?....... 1565
BUS<m>:I2S:FRAMe<n>:TDM<o>:STATe?....... 1565
BUS<m>:I2S:FRAMe<n>:TDM<o>:VALue?....... 1566

BUS<m>:I2S:FCOunt?

Returns the number of decoded frames.

Suffix:

<m>	1..4 Selects the serial bus.

Return values:

<Count>	Number of decoded audio frames

Usage: Query only

BUS<m>:I2S:FRAMe<n>:STATe?

Returns the overall state of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<FrameState>	ERRor | OK | INSufficient OK: the frame is valid. ERRor: an error occured in the frame. INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

Usage: Query only

BUS<m>:I2S:FRAMe<n>:STARt?
BUS<m>:I2S:FRAMe<n>:STOP?

Return the start time and stop time of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<Start>, <Stop>	Range: -100E+24 to 100E+24 *RST: 0

Usage: Query only

BUS<m>:I2S:FRAMe<n>:LEFT:VALue?
BUS<m>:I2S:FRAMe<n>:RIGHt:VALue?

Return the data values of the left and right channel, respectively.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Value> Comma-separated list of values. To set the value format, use `FORMat:BPATtern`.

Range: 0 to 4294967295
*RST: 0

Usage: Query only

BUS<m>:I2S:FRAMe<n>:LEFT:STATe?
BUS<m>:I2S:FRAMe<n>:RIGHt:STATe?

Return the status of the left and right channel of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<WordState> ERRor | OK | INSufficient

OK: the channel data is valid.
ERRor: an error occured in the channel.
INSufficient: the channel is not completely contained in the acquisition.

Usage: Query only

BUS<m>:I2S:FRAMe<n>:TDM<o>:STATe?

Returns the state of the indicated channel of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> 1..8
Selects the TDM channel.

Return values:

<State> ERRor | OK | INSufficient

OK: the channel data is valid.
ERRor: an error occured in the channel.
INSufficient: the channel is not completely contained in the acquisition.

*RST: UNDefined

Usage: Query only

BUS<m>:I2S:FRAMe<n>:TDM<o>:VALue?

Returns the data value of the indicated TDM channel.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> 1..8
Selects the TDM channel.

Return values:

<Value> To set the value format, use `FORMat:BPATtern` on page 1046.
The stated values for range, increment and reset are decimal values.

Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

17.17.9.4 Track and Trend

BUS<m>:I2S:TRACk:LEFT....1567
BUS<m>:I2S:TRACk:RIGHt....1567
BUS<m>:I2S:TRACk:TD1Ch....1567
BUS<m>:I2S:TRACk:TD2Ch....1567
BUS<m>:I2S:TRACk:TD3Ch....1567
BUS<m>:I2S:TRACk:TD4Ch....1567
BUS<m>:I2S:TRACk:TD5Ch....1567
BUS<m>:I2S:TRACk:TD6Ch....1567
BUS<m>:I2S:TRACk:TD7Ch....1567
BUS<m>:I2S:TRACk:TD8Ch....1567
MEASurement<m>:TRACk[:STATe]....1567
MEASurement<m>:TRACk:DATA:HEADer?....1568
MEASurement<m>:TRACk:DATA:STYPe?....1568
MEASurement<m>:TRACk:DATA[:VALues]?....1568

BUS<m>:I2S:TRACk:LEFT <Left>
BUS<m>:I2S:TRACk:RIGHt <Right>

Enables or disables the track of the indicated channel. The commands are relevant for I²S standard, left-justified and right-justified audio data formats.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Left>, <Right> ON | OFF
*RST: OFF

BUS<m>:I2S:TRACk:TD1Ch <TDMCh1>
BUS<m>:I2S:TRACk:TD2Ch <TDMCh2>
BUS<m>:I2S:TRACk:TD3Ch <TDMCh3>
BUS<m>:I2S:TRACk:TD4Ch <TDMCh4>
BUS<m>:I2S:TRACk:TD5Ch <TDMCh5>
BUS<m>:I2S:TRACk:TD6Ch <TDMCh6>
BUS<m>:I2S:TRACk:TD7Ch <TDMCh7>
BUS<m>:I2S:TRACk:TD8Ch <TDMCh8>

Enables or disables the track of the indicated channel. The commands are relevant for TDM audio data.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<TDMChX> ON | OFF
*RST: OFF

MEASurement<m>:TRACk[:STATe] <State>

Enables the track functionality and displays the track.

The track functionality requires at least one option, see "Enable (Track)" on page 350.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:TRACk:DATA:HEADer?

Returns the header of the track.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Query only

MEASurement<m>:TRACk:DATA:STYPe?

Returns the data type: TRK (track).

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Query only

MEASurement<m>:TRACk:DATA[:VALues]?

Returns the data of track points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

Suffix:

<m> 1..10
See "Selection of the measurement: MEASurement<m>" on page 1212.

Usage: Query only

17.17.10 MIL-1553 (Option R&S RTE-K6)

- Configuration.....1568
- Trigger.....1571
- Decode Results.....1581
- Search Settings.....1585
- Search Results.....1591

17.17.10.1 Configuration

BUS<m>:MILStd:SOURce.....1569
BUS<m>:MILStd:MAXResponse:BITS.....1569
BUS<m>:MILStd:MAXResponse:SELect.....1569
BUS<m>:MILStd:MINGap:SELect.....1569
BUS<m>:MILStd:MINGap:BITS.....1570

BUS<m>:MILStd:POLarity....1570
BUS<m>:MILStd:PRESet....1570
BUS<m>:MILStd:THReshold:HIGH....1570
BUS<m>:MILStd:THReshold:LOW....1571

BUS<m>:MILStd:SOURce <SourceData>

Sets the channel for the signal source.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<SourceData>	C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 *RST: C1W1

BUS<m>:MILStd:MAXResponse:BITS <MaxResponseTime>

Sets the value for the maximum response time.

See also: `BUS<m>:MILStd:MAXResponse:SELect`.

Suffix:

<m>	1..4

Parameters:

<MaxResponseTime>	Range: 2E-6 to 262E-6 Increment: 100E-9 *RST: 14E-6 Default unit: s

---

BUS<m>:MILStd:MAXResponse:SELect <MaxRespSel>

Enables the detection of the maximum response time between the last bit of a word and the following status word sync during decoding.

To specify the maximum response time, use `BUS<m>:MILStd:MAXResponse:BITS` on page 1569.

Suffix:

<m>	1..4

Parameters:

<MaxRespSelect>	ON | OFF *RST: ON

BUS<m>:MILStd:MINGap:SELect <MinGapSelect>

Enables the detection of the minimum idle time between the last bit of a message and the following command word sync during decoding.

To specify the minimum gap, use BUS<m>:MILStd:MINGap:BITS.

Suffix:
<m> 1..4

Parameters:
<MinGapSelect> ON | OFF
*RST: ON

BUS<m>:MILStd:MINGap:BITS <MinGapTime>

Sets a value for the minimum gap.

See also: BUS<m>:MILStd:MINGap:SELect.

Suffix:
<m> 1..4

Parameters:
<MinGapTime> Range: 2E-6 to 262E-6
Increment: 100E-9
*RST: 4E-6
Default unit: s

BUS<m>:MILStd:POLarity <Polarity>

Selects the wire on which the bus signal is measured.

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVerted
*RST: NORMal

BUS<m>:MILStd:PRESet <Preset>

Sets the default threshold voltage.

Suffix:
<m> 1..4

Parameters:
<Preset> V05 | V2 | V5 | V7 | MANual
*RST: V5

BUS<m>:MILStd:THReshold:HIGH <ThresholdHigh>

Sets the lower threshold level of the signal.

Suffix:
<m> 1..4

Parameters:

<ThresholdHigh>	Range:	0 to 14
	Increment:	0.1
	*RST:	5
	Default unit:	V

BUS<m>:MILStd:THReshold:LOW <ThresholdLow>

Sets the lower threshold level of the signal.

Suffix:

<m>	1..4

Parameters:

<ThresholdLow>	Range:	-14 to 0
	Increment:	0.1
	*RST:	-5
	Default unit:	V

17.17.10.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:MILStd:TYPE....1572
TRIGger<m>:MILStd:DATA:RCONdition....1573
TRIGger<m>:MILStd:CMD:RCONdition....1573
TRIGger<m>:MILStd:CDST:RCONdition....1573
TRIGger<m>:MILStd:DATA:RMIN....1573
TRIGger<m>:MILStd:CMD:RMIN....1573
TRIGger<m>:MILStd:CDST:RMIN....1573
TRIGger<m>:MILStd:DATA:RMAX....1573
TRIGger<m>:MILStd:CMD:RMAX....1573
TRIGger<m>:MILStd:CDST:RMAX....1573
TRIGger<m>:MILStd:CMD:CCONdition....1574
TRIGger<m>:MILStd:CMD:CMAX....1574
TRIGger<m>:MILStd:CMD:CMIN....1574
TRIGger<m>:MILStd:CMD:SCONdition....1574
TRIGger<m>:MILStd:CMD:SMAX....1575
TRIGger<m>:MILStd:CMD:SMIN....1575
TRIGger<m>:MILStd:CMD:TR....1575
TRIGger<m>:MILStd:CDST:ICONdition....1576
TRIGger<m>:MILStd:CDST:IMAX....1576
TRIGger<m>:MILStd:CDST:IMIN....1576

TRIGger<m>:MILStd:DATA:DCONdition.. 1576
TRIGger<m>:MILStd:DATA:DMAX... 1577
TRIGger<m>:MILStd:DATA:DMIN.. 1577
TRIGger<m>:MILStd:DATA:ICONdition.. 1577
TRIGger<m>:MILStd:DATA:IMAX... 1577
TRIGger<m>:MILStd:DATA:IMIN.. 1578
TRIGger<m>:MILStd:ERRor:MANChester.. 1578
TRIGger<m>:MILStd:ERRor:PARity.. 1578
TRIGger<m>:MILStd:ERRor:SYNC... 1578
TRIGger<m>:MILStd:MAXResponse:BITS.. 1578
TRIGger<m>:MILStd:MAXResponse:SELect... 1579
TRIGger<m>:MILStd:MINGap:BITS.. 1579
TRIGger<m>:MILStd:MINGap:SELect... 1579
TRIGger<m>:MILStd:STATus:BCReceived... 1579
TRIGger<m>:MILStd:STATus:BUSY.. 1580
TRIGger<m>:MILStd:STATus:DBCaccept.. 1580
TRIGger<m>:MILStd:STATus:INSTrument.. 1580
TRIGger<m>:MILStd:STATus:MERRor.. 1580
TRIGger<m>:MILStd:STATus:SREQuest... 1580
TRIGger<m>:MILStd:STATus:SUBSystem.. 1581
TRIGger<m>:MILStd:STATus:TERMinal.. 1581
TRIGger<m>:MILStd:TPSPecifier.. 1581

TRIGger<m>:MILStd:TYPE <Type>

Sets the trigger type for MIL-1553 analysis.

Parameters:

<Type> STYPe | WTYPe | DATA | CDST | CMD | STATword | ERR

STYPe
SyncTYPe: triggers on a sync impulse.

WTYPe
WordTYPe: triggers on the selected word type.

DATA
Triggers on a data word that can be specified.

CDST
CommanDSTatus word: triggers on a command word or on a status word that can be specified.

CMD
CoMmanD word: triggers on a command word or on a status word that can be specified.

STATword
STATus word: triggers on a status word that can be specified.

ERR
ERRor: triggers on any combination of protocol errors.

*RST: STYPe

TRIGger<m>:MILStd:DATA:RCONdition <RTAOperator>
TRIGger<m>:MILStd:CMD:RCONdition <RTAOperator>
TRIGger<m>:MILStd:CDST:RCONdition <RTAOperator>

Sets the operator to define a remote terminal address:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:CDST:RMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CDST:RMIN` and `TRIGger<m>:MILStd:CDST:RMAX`.

*RST: EQUal

TRIGger<m>:MILStd:DATA:RMIN <RTAPatternMin>
TRIGger<m>:MILStd:CMD:RMIN <RTAPatternMin>
TRIGger<m>:MILStd:CDST:RMIN <RTAPatternMin>

Specify a remote terminal address or set the the start value of a remote terminal address range:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:RMAX <RTAPatternMax>
TRIGger<m>:MILStd:CMD:RMAX <RTAPatternMax>
TRIGger<m>:MILStd:CDST:RMAX <RTAPatternMax>

Set the end value of a data range if `TRIGger<m>:MILStd:CDST:RCONdition` is set to `INRange` or `OORange`:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:CCONdition <WordCntOperator>

Sets the operator to set a specific data word count or mode code pattern.

Parameters:

<WordCntOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:CMD:CMIN`.

INRange | OORange
In range/Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CMD:CMIN` and `TRIGger<m>:MILStd:CMD:CMAX`.

*RST: EQUal

TRIGger<m>:MILStd:CMD:CMAX <WordCntPattMax>

Sets the end value of a data word count/mode code pattern if `TRIGger<m>:MILStd:CMD:CCONdition` is set to `INRange` or `OORange`.

Parameters:

<WordCntPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:CMIN <WordCntPattMin>

Specifies a data word count/mode code pattern or sets the the start value of a pattern range.

Parameters:

<WordCntPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:SCONdition <SubaddrOperator>

Sets the operator to set a specific subaddress/mode pattern.

Parameters:

<SubaddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with TRIGger<m>:MILStd:CMD:SMIN.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with TRIGger<m>:MILStd:CMD:SMIN and TRIGger<m>:MILStd:CMD:SMAX.

*RST: EQUal

TRIGger<m>:MILStd:CMD:SMAX <SubaddrPattMax>

Sets the end value of the subaddress range if TRIGger<m>:MILStd:CMD:SCONdition is set to INRange or OORange.

Parameters:

<SubaddrPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:SMIN <SubaddrPattMin>

Specifies a subaddress or sets the the start value of a subaddress range.

Parameters:

<SubaddrPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:TR <TRFlag>

Triggers on a transmission mode.

Parameters:

<TRFlag> ONE | ZERO | DC

ONE
Transmit direction.

ZERO
Receive direction.

DC
Either directions.

*RST: DC

TRIGger<m>:MILStd:CDST:ICONdition <InfoOperator>

Sets the operator to set a specific info for the 9th to 19th bit of a command or status word.

Parameters:

<InfoOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:CDST:IMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CDST:IMIN` and `TRIGger<m>:MILStd:CDST:IMAX`.

*RST: EQUal

TRIGger<m>:MILStd:CDST:IMAX <InfoPatternMax>

Sets the end value of the info range if `TRIGger<m>:MILStd:CDST:ICONdition` is set to `INRange`.

Parameters:

<InfoPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CDST:IMIN <InfoPatternMin>

Specifies an info or sets the the start value of an info range.

Parameters:

<InfoPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:DCONdition <DataOperator>

Sets the operator to set a specific data pattern.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:DATA:DMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:DATA:DMIN` and `TRIGger<m>:MILStd:DATA:DMAX`.

*RST: EQUal

TRIGger<m>:MILStd:DATA:DMAX <DataPatternMax>

Sets the end value of a data pattern range if `TRIGger<m>:MILStd:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:DMIN <DataPatternMin>

Specifies a data pattern or sets the the start value of a data pattern range.

Parameters:

<DataPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:ICONdition <DataIdxOperator>

Sets the operator to set a specific range within this series of the data words that is considered for the analysis.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal| LTHan | LETHan | GTHan | GETHan
Equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:DATA:IMIN`.

INRange = RANGe
In range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:DATA:IMIN` and `TRIGger<m>:MILStd:DATA:IMAX`.

*RST: INRange

TRIGger<m>:MILStd:DATA:IMAX <DataIndexMax>

Sets the end value of a data word series index range if `TRIGger<m>:MILStd:DATA:ICONdition` is set to `INRange`.

Parameters:

<DataIndexMax>	Range:	1 to 32
	Increment:	1
	*RST:	32

TRIGger<m>:MILStd:DATA:IMIN <DataIndexMin>

Specifies an index or sets the the start value of a data word series index range.

Parameters:

<DataIndexMin>	Range:	1 to 32
	Increment:	1
	*RST:	1

TRIGger<m>:MILStd:ERRor:MANChester <ManCodingError>

Triggers on an error of the manchester coding if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:

<ManCodingError> ON | OFF

*RST: ON

TRIGger<m>:MILStd:ERRor:PARity <ParityError>

Triggers on even parity if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:

<ParityError> ON | OFF

*RST: ON

TRIGger<m>:MILStd:ERRor:SYNC <SyncError>

Triggers on an error of the synchronization if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:

<SyncError> ON | OFF

*RST: ON

TRIGger<m>:MILStd:MAXResponse:BITS <MaxResponseTime>

Sets the value for the maximum response time to be triggered on if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`

Enable the error trigger on maximum response time with `TRIGger<m>:MILStd:MAXResponse:SELect`.

Parameters:

<MaxResponseTime>	Range:	2E-6 to 262E-6
	Increment:	100E-9
	*RST:	14E-6
	Default unit:	s

TRIGger<m>:MILStd:MAXResponse:SELect <MaxRespSel>

Enables the trigger on exceeding the maximum response time if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`. You can set the maximum time with: `TRIGger<m>:MILStd:MAXResponse:BITS`.

Parameters:

<MaxRespSel>	ON | OFF	
	*RST:	ON

TRIGger<m>:MILStd:MINGap:BITS <MinGapTime>

Sets the value for the minimum gap to be triggered on if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:

<MinGapTime>	Range:	2E-6 to 262E-6
	Increment:	100E-9
	*RST:	4E-6
	Default unit:	s

TRIGger<m>:MILStd:MINGap:SELect <MinGapSelect>

Enables triggering when the minimum gap is out of range if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`. You can set the minimum gap with: `TRIGger<m>:MILStd:MINGap:BITS`.

Parameters:

<MinGapSelect>	ON | OFF	
	*RST:	ON

TRIGger<m>:MILStd:STATus:BCReceived <BroadcastCmd>

Triggers on the state of the broadcast command received bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<BroadcastCmd>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:BUSY <BusyFlag>

Triggers on the state of the busy bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<BusyFlag>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:DBCaccept <DynBusControl>

Triggers on the state of the dynamic bus control accept bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<DynBusControl>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:INSTrument <InstFlag>

Triggers on the state of the instrumentation bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<InstFlag>	ONE | ZERO | DC	
	*RST:	ZERO

TRIGger<m>:MILStd:STATus:MERRor <MessageError>

Triggers on the state of the message error bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<MessageError>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:SREQuest <ServiceRequest>

Triggers on the state of the the service request bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<ServiceRequest>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:SUBSystem <SubsystemFlag>

Triggers on the state of the subsystem flag bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<SubsystemFlag>	ONE | ZERO | DC
	*RST: DC

TRIGger<m>:MILStd:STATus:TERMinal <TerminalFlag>

Triggers on the state of theterminal flag bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<TerminalFlag>	ONE | ZERO | DC
	*RST: DC

TRIGger<m>:MILStd:TPSPecifier <TypeSpecifier>

Sets the sync impulse/ word type to be triggered on.

Parameters:

<TypeSpecifier>	CSTatus | DATA | ALL
	CSTatus: command/status word
	*RST: ALL

17.17.10.3 Decode Results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1384
- `BUS<m>:SYMBols` on page 1385

BUS<m>:MILStd:WCOunt? 1581
BUS<m>:MILStd:WORD<n>:DATA? 1582
BUS<m>:MILStd:WORD<n>:INFO? 1582
BUS<m>:MILStd:WORD<n>:RTADdress? 1582
BUS<m>:MILStd:WORD<n>:STARt? 1583
BUS<m>:MILStd:WORD<n>:STATus? 1583
BUS<m>:MILStd:WORD<n>:STOP? 1584
BUS<m>:MILStd:WORD<n>:SYMBol? 1584
BUS<m>:MILStd:WORD<n>:TYPE? 1584

BUS<m>:MILStd:WCOunt?

Returns the number of decoded words.

Suffix:

<m>	1..4

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:MILStd:WORD<n>:DATA?

Return the data bytes of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<Data> 16-bit data of the specified word as a 2-byte bit pattern (B1, B2). The first byte B1 is the most significant byte.

Example:
```
BUS:MILStd:WORD4:DATA?
<-- #H08,#H49
```

Usage: Query only

BUS<m>:MILStd:WORD<n>:INFO?

Returns the info value for the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameInfo>	Range:	0 to 2047
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:MILStd:WORD<n>:RTADdress?

Returns the RT address for the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameRta> Range: 0 to 31
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MILStd:WORD<n>:STARt?

Return the start time of the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MILStd:WORD<n>:STATus?

Returns the overall state of the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameState> OK | SYNC | MANC | PAR | GAP | RT

OK: the word is valid.
SYNC: synchronization error occured.
MANC: manchester coding error occured.
PAR: parity error occured.
GAP: timing gap error occured.
RT: remote terminal error occured.

*RST: OK

Usage: Query only

BUS<m>:MILStd:WORD<n>:STOP?

Return the stop time of the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MILStd:WORD<n>:SYMBol?

Returns the label name of the word ID.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<Translation>

Usage: Query only

BUS<m>:MILStd:WORD<n>:TYPE?

Returns the type of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameType> UNKNown | DATA | CMD | STAT | CMST | IM

CMD: command word
CMST: command/status word
IM: inter message. Shows if there are gap errors or response timeout.

*RST: DATA

Usage: Query only

17.17.10.4 Search Settings

The search remote commands are very similar to the trigger commands. Therefore, search coommands are described in short, for more details, see the corresponding trigger command in Chapter 17.17.10.2, "Trigger", on page 1571.

SEARch:TRIGger:MILStd:TYPE....1586
SEARch:TRIGger:MILStd:DATA:RCONdition....1586
SEARch:TRIGger:MILStd:CMD:RCONdition....1586
SEARch:TRIGger:MILStd:CDST:RCONdition....1586
SEARch:TRIGger:MILStd:DATA:RMIN....1587
SEARch:TRIGger:MILStd:CMD:RMIN....1587
SEARch:TRIGger:MILStd:CDST:RMIN....1587
SEARch:TRIGger:MILStd:DATA:RMAX....1587
SEARch:TRIGger:MILStd:CMD:RMAX....1587
SEARch:TRIGger:MILStd:CDST:RMAX....1587
SEARch:TRIGger:MILStd:CDST:ICONdition....1587
SEARch:TRIGger:MILStd:CMD:CCONdition....1587
SEARch:TRIGger:MILStd:CMD:SCONdition....1588
SEARch:TRIGger:MILStd:DATA:DCONdition....1588
SEARch:TRIGger:MILStd:CDST:IMIN....1588
SEARch:TRIGger:MILStd:CMD:CMIN....1588
SEARch:TRIGger:MILStd:CMD:SMIN....1588
SEARch:TRIGger:MILStd:DATA:DMIN....1588
SEARch:TRIGger:MILStd:CDST:IMAX....1588
SEARch:TRIGger:MILStd:CMD:CMAX....1588
SEARch:TRIGger:MILStd:CMD:SMAX....1589
SEARch:TRIGger:MILStd:DATA:DMAX....1589
SEARch:TRIGger:MILStd:DATA:ICONdition....1589
SEARch:TRIGger:MILStd:DATA:IMIN....1589
SEARch:TRIGger:MILStd:DATA:IMAX....1589
SEARch:TRIGger:MILStd:CMD:TR....1590
SEARch:TRIGger:MILStd:ERRor:MANChester....1590
SEARch:TRIGger:MILStd:ERRor:PARity....1590
SEARch:TRIGger:MILStd:ERRor:SYNC....1590
SEARch:TRIGger:MILStd:ERRor:TIMing....1590
SEARch:TRIGger:MILStd:STATus:BCReceived....1590
SEARch:TRIGger:MILStd:STATus:BUSY....1590
SEARch:TRIGger:MILStd:STATus:DBCaccept....1591
SEARch:TRIGger:MILStd:STATus:INSTrument....1591
SEARch:TRIGger:MILStd:STATus:MERRor....1591
SEARch:TRIGger:MILStd:STATus:SREQuest....1591
SEARch:TRIGger:MILStd:STATus:SUBSystem....1591
SEARch:TRIGger:MILStd:STATus:TERMinal....1591
SEARch:TRIGger:MILStd:TPSPecifier....1591

SEARch:TRIGger:MILStd:TYPE <SearchName>,<Type>
SEARch:TRIGger:MILStd:TYPE? <SearchName>

Sets the event to be searched for.

Parameters:

<Type> STYPe | WTYPe | DATA | CDST | CMD | STATword | ERR

See TRIGger<m>:MILStd:TYPE on page 1572

*RST: STYPe

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:DATA:RCONdition <SearchName>,<RTAOperator>
SEARch:TRIGger:MILStd:DATA:RCONdition? <SearchName>
SEARch:TRIGger:MILStd:CMD:RCONdition <SearchName>,<RTAOperator>
SEARch:TRIGger:MILStd:CMD:RCONdition? <SearchName>
SEARch:TRIGger:MILStd:CDST:RCONdition <SearchName>,<RTAOperator>
SEARch:TRIGger:MILStd:CDST:RCONdition? <SearchName>

Set the operator to define a remote terminal address:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with SEARch:TRIGger:MILStd:CDST:RMIN.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with SEARch:TRIGger:MILStd:CDST:RMIN and SEARch:TRIGger:MILStd:CDST:RMAX on page 1587.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:MILStd:DATA:RMIN <SearchName>,<RTAPatternMin>
SEARch:TRIGger:MILStd:DATA:RMIN? <SearchName>
SEARch:TRIGger:MILStd:CMD:RMIN <SearchName>,<RTAPatternMin>
SEARch:TRIGger:MILStd:CMD:RMIN? <SearchName>
SEARch:TRIGger:MILStd:CDST:RMIN <SearchName>,<RTAPatternMin>
SEARch:TRIGger:MILStd:CDST:RMIN? <SearchName>

Specify a remote terminal address or set the the start value of a remote terminal address range:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:DATA:RMAX <SearchName>,<RTAPatternMax>
SEARch:TRIGger:MILStd:DATA:RMAX? <SearchName>
SEARch:TRIGger:MILStd:CMD:RMAX <SearchName>,<RTAPatternMax>
SEARch:TRIGger:MILStd:CMD:RMAX? <SearchName>
SEARch:TRIGger:MILStd:CDST:RMAX <SearchName>,<RTAPatternMax>
SEARch:TRIGger:MILStd:CDST:RMAX? <SearchName>

Set the end value of a data range if `SEARch:TRIGger:MILStd:CDST:RCONdition` is set to `INRange` or `OORange`:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:CDST:ICONdition <SearchName>,<InfoOperator>
SEARch:TRIGger:MILStd:CDST:ICONdition? <SearchName>
SEARch:TRIGger:MILStd:CMD:CCONdition <SearchName>,<WordCntOperator>
SEARch:TRIGger:MILStd:CMD:CCONdition? <SearchName>

SEARch:TRIGger:MILStd:CMD:SCONdition <SearchName>,<SubaddrOperator>
SEARch:TRIGger:MILStd:CMD:SCONdition? <SearchName>
SEARch:TRIGger:MILStd:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:MILStd:DATA:DCONdition? <SearchName>

Sets the operator for the corresponding search:

- CDST:ICON - specific info for the 9th to 19th bit of a command or status word.
- CMD:CCON - specific data word count or mode code pattern in a command word
- CMD:SCON - specific subaddress/mode pattern in a command word
- DATA:DCON - data pattern in a data word

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange
*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:CDST:IMIN <SearchName>,<InfoPatternMin>
SEARch:TRIGger:MILStd:CDST:IMIN? <SearchName>
SEARch:TRIGger:MILStd:CMD:CMIN <SearchName>,<WordCntPattMin>
SEARch:TRIGger:MILStd:CMD:CMIN? <SearchName>
SEARch:TRIGger:MILStd:CMD:SMIN <SearchName>,<SubaddrPattMin>
SEARch:TRIGger:MILStd:CMD:SMIN? <SearchName>
SEARch:TRIGger:MILStd:DATA:DMIN <SearchName>,<DataPatternMin>
SEARch:TRIGger:MILStd:DATA:DMIN? <SearchName>

Sets the pattern or the start value of a pattern range for the corresponding search:

- CDST:IMIN - specific info for the 9th to 19th bit of a command or status word.
- CMD:CMIN - specific data word count or mode code pattern in a command word
- CMD:SMIN - specific subaddress/mode pattern in a command word
- DATA:DMIN - data pattern in a data word

Parameters:
<DataPatternMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:CDST:IMAX <SearchName>,<InfoPatternMax>
SEARch:TRIGger:MILStd:CDST:IMAX? <SearchName>
SEARch:TRIGger:MILStd:CMD:CMAX <SearchName>,<WordCntPattMax>
SEARch:TRIGger:MILStd:CMD:CMAX? <SearchName>

SEARch:TRIGger:MILStd:CMD:SMAX <SearchName>,<SubaddrPattMax>
SEARch:TRIGger:MILStd:CMD:SMAX? <SearchName>
SEARch:TRIGger:MILStd:DATA:DMAX <SearchName>,<DataPatternMax>
SEARch:TRIGger:MILStd:DATA:DMAX? <SearchName>

Sets the end value of a pattern range for the corresponding search:

- CDST:IMAX - specific info for the 9th to 19th bit of a command or status word.
- CMD:CMAX - specific data word count or mode code pattern in a command word
- CMD:SMAX - specific subaddress/mode pattern in a command word
- DATA:DMAX - data pattern in a data word

Parameters:
<DataPatternMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:MILStd:DATA:ICONdition? <SearchName>

Sets the operator to set a range within a series of the data words that is considered for the search.

Parameters:

<DataIdxOperator>	EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe INRange = RANGe *RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:MILStd:DATA:IMIN? <SearchName>

Specifies an index or sets the the start value of a data word series index range.

Parameters:

<DataIndexMin>	Range: 1 to 32 Increment: 1 *RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:MILStd:DATA:IMAX? <SearchName>

Sets the end value of a data word series index range if the operator is set to `INRange`.

Parameters:

<DataIndexMax>	Range:	1 to 32
	Increment:	1
	*RST:	32

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:CMD:TR <SearchName>,<TRFlag>
SEARch:TRIGger:MILStd:CMD:TR? <SearchName>

Searches for a transmission mode.

Parameters:

<TRFlag>	ONE | ZERO | DC	
	*RST:	DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:ERRor:MANChester <SearchName>,<ManCodingError>
SEARch:TRIGger:MILStd:ERRor:MANChester? <SearchName>
SEARch:TRIGger:MILStd:ERRor:PARity <SearchName>,<ParityError>
SEARch:TRIGger:MILStd:ERRor:PARity? <SearchName>
SEARch:TRIGger:MILStd:ERRor:SYNC <SearchName>,<SyncError>
SEARch:TRIGger:MILStd:ERRor:SYNC? <SearchName>
SEARch:TRIGger:MILStd:ERRor:TIMing <SearchName>,<MinGapSelect>
SEARch:TRIGger:MILStd:ERRor:TIMing? <SearchName>

Enables search for errors if `SEARch:TRIGger:MILStd:TYPE` is set to `ERRor`.

- MANChester: error of the manchester coding
- PARity: even parity (parity error)
- SYNC: error of the synchronization
- TIMing: Minimum gap is out of range

Parameters:

<SyncError>, <ParityError>, <ManCodingError>, <MinGapSelect>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:STATus:BCReceived <SearchName>,<BroadcastCmd>
SEARch:TRIGger:MILStd:STATus:BCReceived? <SearchName>
SEARch:TRIGger:MILStd:STATus:BUSY <SearchName>,<BusyFlag>
SEARch:TRIGger:MILStd:STATus:BUSY? <SearchName>

SEARch:TRIGger:MILStd:STATus:DBCaccept <SearchName>,<DynBusControl>
SEARch:TRIGger:MILStd:STATus:DBCaccept? <SearchName>
SEARch:TRIGger:MILStd:STATus:INSTrument <SearchName>,<InstFlag>
SEARch:TRIGger:MILStd:STATus:INSTrument? <SearchName>
SEARch:TRIGger:MILStd:STATus:MERRor <SearchName>,<MessageError>
SEARch:TRIGger:MILStd:STATus:MERRor? <SearchName>
SEARch:TRIGger:MILStd:STATus:SREQuest <SearchName>,<ServiceRequest>
SEARch:TRIGger:MILStd:STATus:SREQuest? <SearchName>
SEARch:TRIGger:MILStd:STATus:SUBSystem <SearchName>,<SubsystemFlag>
SEARch:TRIGger:MILStd:STATus:SUBSystem? <SearchName>
SEARch:TRIGger:MILStd:STATus:TERMinal <SearchName>,<TerminalFlag>
SEARch:TRIGger:MILStd:STATus:TERMinal? <SearchName>

Specifies the values (0, 1, X) of the status flags if `SEARch:TRIGger:MILStd:TYPE` is set to `STATword`.

Parameters:

<TerminalFlag> ONE | ZERO | DC

*RST: DC

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:TPSPecifier <SearchName>,<TypeSpecifier>
SEARch:TRIGger:MILStd:TPSPecifier? <SearchName>

Sets the sync impulse/ word type to be searched for.

Parameters:

<TypeSpecifier> CSTatus | DATA | ALL

CSTatus: command/status word

*RST: ALL

Parameters for setting and query:

<SearchName>

17.17.10.5 Search Results

The search on decoded MIL-1553 data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.10.3, "Decode Results", on page 1581.

SEARch:RESult:MILStd:WCOunt?....1592
SEARch:RESult:MILStd:WORD<m>:TYPE?....1592
SEARch:RESult:MILStd:WORD<m>:STATus?....1592
SEARch:RESult:MILStd:WORD<m>:STARt?....1592
SEARch:RESult:MILStd:WORD<m>:STOP?....1593

SEARch:RESult:MILStd:WORD<m>:SYMBol?..1593
SEARch:RESult:MILStd:WORD<m>:RTADdress?... 1593
SEARch:RESult:MILStd:WORD<m>:DATA?.. 1594
SEARch:RESult:MILStd:WORD<m>:INFO?...1594

SEARch:RESult:MILStd:WCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> UNKNown | DATA | CMD | STAT | CMST | IM
*RST: DATA

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | SYNC | MANC | PAR | GAP | RT
*RST: OK

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:RTADdress? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameRta> Range: 0 to 31
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:INFO? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameInfo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

17.17.11 ARINC 429 (Option R&S RTE-K7)

● Configuration....1594
● Trigger....1597
● Decode Results....1601
● Search Settings....1604
● Search Results....1607

17.17.11.1 Configuration

BUS<m>:ARINc:SOURce....1595
BUS<m>:ARINc:BRValue....1595
BUS<m>:ARINc:BRMode....1595
BUS<m>:ARINc:MAXGap:SELect....1595
BUS<m>:ARINc:MAXGap:BITS....1596
BUS<m>:ARINc:MINGap:SELect....1596
BUS<m>:ARINc:MINGap:BITS....1596
BUS<m>:ARINc:POLarity....1596
BUS<m>:ARINc:PRESet....1597
BUS<m>:ARINc:THReshold:HIGH....1597
BUS<m>:ARINc:THReshold:LOW....1597

BUS<m>:ARINc:SOURce <SourceData>

Sets the channel for the signal source.

Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<SourceData>	C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 *RST: C1W1

BUS<m>:ARINc:BRValue <BitRateValue>

Sets the number of transmitted bits per second.

Suffix:

<m>	1..4

Parameters:

<BitRateValue>	Range: 10000 to 110000 Increment: 100 *RST: 100000 Default unit: bps

BUS<m>:ARINc:BRMode <BitRateMode>

Sets the bit rate mode to high or low speed.

Suffix:

<m>	1..4

Parameters:

<BitRateMode>	HIGH | LOW *RST: HIGH

BUS<m>:ARINc:MAXGap:SELect <MaxGapSelect>

Enables the detection of the maximum gap time during decoding.

To specify the minimum gap time `BUS<m>:ARINc:MINGap:BITS`.

Suffix:

<m>	1..4

Parameters:

<MaxGapSelect>	ON | OFF *RST: OFF

BUS<m>:ARINc:MAXGap:BITS <MaxGapBits>

Sets the value for the maximum gap between two words.

See also: `BUS<m>:ARINc:MAXGap:SELect`

Suffix:

<m>	1..4

Parameters:

<MaxGapBits>	Range: 0 to 1000 Increment: 1 *RST: 100 Default unit: bit

BUS<m>:ARINc:MINGap:SELect <MinGapSelect>

Enables the detection of the minimum idle time between two words during decoding.

To specify the minimum gap, use`BUS<m>:ARINc:MINGap:BITS`.

Suffix:

<m>	1..4

Parameters:

<MinGapSelect>	ON | OFF *RST: ON

BUS<m>:ARINc:MINGap:BITS <MinGapBits>

Sets a value for the minimum timing gap between two words.

See also: `BUS<m>:ARINc:MINGap:SELect` on page 1596.

Suffix:

<m>	1..4

Parameters:

<MinGapBits>	Range: 0 to 100 Increment: 1 *RST: 4 Default unit: bit

BUS<m>:ARINc:POLarity <Polarity>

Sets the wire on which the bus signal is measured.

Suffix:

<m>	1..4

Parameters:

<Polarity>	ALEG | BLEG *RST: ALEG

BUS<m>:ARINc:PRESet <Preset>

Sets the default threshold voltage.

Suffix:

<m>	1..4

Parameters:

<Preset>	V25 | V5 | MANual	
	*RST:	V5

BUS<m>:ARINc:THReshold:HIGH <ThresholdHigh>

Sets the high threshold level of the signal.

Suffix:

<m>	1..4

Parameters:

<ThresholdHigh>	Range:	0 to 12
	Increment:	0.1
	*RST:	5
	Default unit:	V

BUS<m>:ARINc:THReshold:LOW <ThresholdLow>

Sets the low threshold level of the signal.

Suffix:

<m>	1..4

Parameters:

<ThresholdLow>	Range:	-12 to 0
	Increment:	0.1
	*RST:	-5
	Default unit:	V

17.17.11.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:ARINc:TYPE....1598
TRIGger<m>:ARINc:DATA:CONDition....1598
TRIGger<m>:ARINc:DATA:MIN....1598

TRIGger<m>:ARINc:DATA:MAX....................1599
TRIGger<m>:ARINc:ERRor:CODing....................1599
TRIGger<m>:ARINc:ERRor:PARity....................1599
TRIGger<m>:ARINc:LABel:CONDition....................1599
TRIGger<m>:ARINc:LABel:MIN....................1599
TRIGger<m>:ARINc:LABel:MAX....................1600
TRIGger<m>:ARINc:MINGap:SELect....................1600
TRIGger<m>:ARINc:MINGap:BITS....................1600
TRIGger<m>:ARINc:MAXGap:SELect....................1600
TRIGger<m>:ARINc:MAXGap:BITS....................1600
TRIGger<m>:ARINc:SDI....................1601
TRIGger<m>:ARINc:SSM....................1601

TRIGger<m>:ARINc:TYPE <Type>

Sets the trigger type for ARINC 429 analysis.

Parameters:

<Type> STARt | STOP | LABel | ERRor

*RST: STARt

TRIGger<m>:ARINc:DATA:CONDition <DataOperator>

Sets the condition for the data. You can define an exact data patterm or a data range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:ARINc:DATA:MIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:ARINc:DATA:MIN` and `TRIGger<m>:ARINc:DATA:MAX`.

*RST: EQUal

TRIGger<m>:ARINc:DATA:MIN <DataMin>

Defines the minimum bit pattern for the data.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:ARINc:DATA:MAX <DataMax>

Sets the end value of a data pattern if `TRIGger<m>:ARINc:DATA:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DataMax>

TRIGger<m>:ARINc:ERRor:CODing <CodingError>

Enables triggering when a coding error occurs.

Parameters:

<CodingError>	ON | OFF *RST: ON

TRIGger<m>:ARINc:ERRor:PARity <ParityError>

Enables triggering when a parity error occurs.

Parameters:

<ParityError>	ON | OFF *RST: ON

TRIGger<m>:ARINc:LABel:CONDition <LabelOperator>

Sets the condition for the label. You can define an exact label or a label range.

Parameters:

<LabelOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange **EQUal | NEQual | LTHan | LETHan | GTHan | GETHan** Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with`TRIGger<m>:ARINc:LABel:MIN`. **INRange | OORange** In range/Out of range: set the minimum and maximum value of the range with`TRIGger<m>:ARINc:LABel:MIN` and `TRIGger<m>:ARINc:LABel:MAX`. *RST: EQUal

TRIGger<m>:ARINc:LABel:MIN <LabelMin>

Defines the minimum bit pattern for the label.

Parameters:

<LabelMin>	Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:ARINc:LABel:MAX <LabelMax>

Sets the end value of a label pattern if `TRIGger<m>:ARINc:LABel:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LabelMax>	Numeric or string pattern, see `TRIGger<m>:ARINc:LABel:MIN`

TRIGger<m>:ARINc:MINGap:SELect <MinGapSelect>

Enables triggering when the minimum gap is out of range. You can set the minimum gap with: `TRIGger<m>:ARINc:MINGap:BITS`.

Parameters:

<MinGapSelect>	ON | OFF *RST: ON

TRIGger<m>:ARINc:MINGap:BITS <MinGapBits>

Sets the value for the minimum gap to be triggered on.

Parameters:

<MinGapBits>	Range: 0 to 100 Increment: 1 *RST: 4 Default unit: bit

TRIGger<m>:ARINc:MAXGap:SELect <MaxGapSelect>

Enables triggering when the maximum gap is out of range. You can set the maximum gap with: `TRIGger<m>:ARINc:MAXGap:BITS`.

Parameters:

<MaxGapSelect>	ON | OFF *RST: OFF

TRIGger<m>:ARINc:MAXGap:BITS <MaxGapBits>

Sets the value for the maximum gap to be triggered on.

Parameters:

<MaxGapBits>	Range: 0 to 1000 Increment: 1 *RST: 100 Default unit: bit

TRIGger<m>:ARINc:SDI <SDI>

Sets the source/destination identifier (SDI) bits.

Parameters:
<SDI>

TRIGger<m>:ARINc:SSM <SSM>

Sets the sign/status matrix (SSM) bits.

Parameters:
<SSM>

17.17.11.3 Decode Results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1384
- `BUS<m>:SYMBols` on page 1385

BUS<m>:ARINc:WCOunt? 1601
BUS<m>:ARINc:WORD<n>:DATA? 1601
BUS<m>:ARINc:WORD<n>:LABel? 1602
BUS<m>:ARINc:WORD<n>:PATTern? 1602
BUS<m>:ARINc:WORD<n>:SDI? 1602
BUS<m>:ARINc:WORD<n>:SSM? 1603
BUS<m>:ARINc:WORD<n>:STARt? 1603
BUS<m>:ARINc:WORD<n>:STATe? 1603
BUS<m>:ARINc:WORD<n>:STOP? 1604
BUS<m>:ARINc:WORD<n>:SYMBol? 1604

BUS<m>:ARINc:WCOunt?

Returns the number of decoded words.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:DATA?

Returns the data of the specified word.

Suffix:
<m> 1..4

<n> *

Return values:

<FrameData> 19-bit data field of the word as an integer

Range: 0 to 0
Increment: 1
*RST: 0

Example:
```
BUS:ARINc:WORD3:DATA?
<-- 148035
```

Usage: Query only

BUS<m>:ARINc:WORD<n>:LABel?

Returns the label of the specified word.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameLabel> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:PATTern?

Returns all 32 bits of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

Return values:

<Data> Comma-separated list of 4 bytes in big endian order. The format of each byte is defined by FORMat:BPATtern.

Example:
```
BUS2:ARINc:WORD3:PATTern?
--> #H75,#H11,#H55,#H82
FORMat:BPATtern DEC
BUS2:ARINc:WORD3:PATTern?
--> 117,17,85,130
```

Usage: Query only

BUS<m>:ARINc:WORD<n>:SDI?

Returns the source/destination identifier (SDI) bits of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<SDI> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:SSM?

Returns the sign/status matrix(SSM) bits of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<SSM> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:STARt?

Returns the start time of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:ARINc:WORD<n>:STATe?

Returns the overall state of the specified word.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the word.

Return values:

<FrameState> OK | CODE | GAP | PAR

CODE: coding error occured.
GAP: timing gap error occured.
PAR: parity error occured.

*RST: OK

Usage: Query only

BUS<m>:ARINc:WORD<n>:STOP?

Returns the end time of the specified word.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:ARINc:WORD<n>:SYMBol?

Returns the label name of the word ID.

Suffix:

<m> 1..4

<n> *

Return values:

<Translation>

Usage: Query only

17.17.11.4 Search Settings

SEARch:TRIGger:ARINc:TYPE....1605
SEARch:TRIGger:ARINc:LABel:CONDition....1605
SEARch:TRIGger:ARINc:DATA:CONDition....1605
SEARch:TRIGger:ARINc:LABel:MIN....1605
SEARch:TRIGger:ARINc:DATA:MIN....1605
SEARch:TRIGger:ARINc:LABel:MAX....1606
SEARch:TRIGger:ARINc:DATA:MAX....1606
SEARch:TRIGger:ARINc:SDI....1606
SEARch:TRIGger:ARINc:SSM....1606

SEARch:TRIGger:ARINc:ERRor:CODing.. 1606
SEARch:TRIGger:ARINc:ERRor:PARity.. 1607
SEARch:TRIGger:ARINc:ERRor:TIMing.. 1607

SEARch:TRIGger:ARINc:TYPE <SearchName>,<Type>
SEARch:TRIGger:ARINc:TYPE? <SearchName>

Sets the search type.

Parameters:

<Type> STARt | STOP | LABel | ERRor

*RST: STARt

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:LABel:CONDition <SearchName>,<LabelOperator>
SEARch:TRIGger:ARINc:LABel:CONDition? <SearchName>
SEARch:TRIGger:ARINc:DATA:CONDition <SearchName>,<DataOperator>
SEARch:TRIGger:ARINc:DATA:CONDition? <SearchName>

Set the condition for the label or data, respectively. You can define an exact value or a value range

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:ARINc:...:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range. with`TRIGger<m>:ARINc:LABel:MIN` and `TRIGger<m>:ARINc:LABel:MAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:LABel:MIN <SearchName>,<LabelMin>
SEARch:TRIGger:ARINc:LABel:MIN? <SearchName>
SEARch:TRIGger:ARINc:DATA:MIN <SearchName>,<DataMin>
SEARch:TRIGger:ARINc:DATA:MIN? <SearchName>

Specifies a label or data bit pattern, or sets the the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ARINc:LABel:MAX <SearchName>,<LabelMax>
SEARch:TRIGger:ARINc:LABel:MAX? <SearchName>
SEARch:TRIGger:ARINc:DATA:MAX <SearchName>,<DataMax>
SEARch:TRIGger:ARINc:DATA:MAX? <SearchName>

Set the end value of a label or data pattern if the condition is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see `SEARch:TRIGger:ARINc:LABel:MIN`

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:SDI <SearchName>,<SDI>
SEARch:TRIGger:ARINc:SDI? <SearchName>

Sets the source/destination identifier (SDI) bits.

Parameters:

<SDI>

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:SSM <SearchName>,<SSM>
SEARch:TRIGger:ARINc:SSM? <SearchName>

Sets the sign/status matrix (SSM) bits.

Parameters:

<SSM>

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:ERRor:CODing <SearchName>,<CodingError>
SEARch:TRIGger:ARINc:ERRor:CODing? <SearchName>

Enables the search for coding errors.

Parameters:

<CodingError> ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:ERRor:PARity <SearchName>,<ParityError>
SEARch:TRIGger:ARINc:ERRor:PARity? <SearchName>

Enables the search for parity errors.

Parameters:

<ParityError> ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:ARINc:ERRor:TIMing <SearchName>,<MinGapSelect>
SEARch:TRIGger:ARINc:ERRor:TIMing? <SearchName>

Enables the search for timing errors, when the minimum gap is out of range.

Parameters:

<MinGapSelect> ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

17.17.11.5 Search Results

The search on decoded ARINC 429 data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.11.3, "Decode Results", on page 1601.

SEARch:RESult:ARINc:WORD<m>:LABel?....1608
SEARch:RESult:ARINc:WORD<m>:PATTern?....1608
SEARch:RESult:ARINc:WORD<m>:DATA?....1608
SEARch:RESult:ARINc:WORD<m>:SSM?....1608
SEARch:RESult:ARINc:WORD<m>:SYMBol?....1609
SEARch:RESult:ARINc:WCOunt?....1609
SEARch:RESult:ARINc:WORD<m>:STOP?....1609
SEARch:RESult:ARINc:WORD<m>:SDI?....1609
SEARch:RESult:ARINc:WORD<m>:STATe?....1610
SEARch:RESult:ARINc:WORD<m>:STARt?....1610

SEARch:RESult:ARINc:WORD<m>:LABel? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Label> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:PATTern? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DataValues>

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:SSM? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SSM> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:ARINc:WCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:SDI? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<SDI>	Range:	0 to 3
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:STATe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | CODE | GAP | PAR

*RST: OK

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

17.17.12 Ethernet 10BASE-T and 100BASE-TX (Option R&S RTE-K8)

- Configuration....1610
- Trigger....1613
- Decode Results....1618
- Search Settings....1624
- Search Results....1629

17.17.12.1 Configuration

In all `BUS<m>:ETHernet` commands, the suffix <m> selects the serial bus.

BUS<m>:ETHernet:VARiant....1611
BUS<m>:ETHernet:SOURce....1611
BUS<m>:ETHernet:POLarity....1611
BUS<m>:ETHernet:THReshold:HIGH....1612
BUS<m>:ETHernet:THReshold:LOW....1612
BUS<m>:ETHernet:PRESet....1612
BUS<m>:ETHernet:BITRate....1613

BUS<m>:ETHernet:VARiant <Variant>

Selects the Ethernet protocol variant and transmission speed.

Suffix:

<m>	1..4

Parameters:

<Variant> B10T | B100TX | B100tx

B10T
Ethernet protocol variant 10BASE-T (10 Mbit/s)

B100TX = B100tx
Ethernet protocol variant 100BASE-TX (100 Mbit/s)

*RST: B10T

BUS<m>:ETHernet:SOURce <SourceData>

Selects the source channel for the data signal.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Suffix:

<m>	1..4

Parameters:

<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

BUS<m>:ETHernet:POLarity <Polarity>

Defines the polarity of the data signal. This setting is only available in 10BASE-T.

Suffix:

<m>	1..4

Parameters:

<Polarity> NORMal | INVert

NORMal
Normal (non-inverted) data signal polarity

INVert
Inverted data signal polarity

*RST: NORMal

BUS<m>:ETHernet:THReshold:HIGH <ThresholdUpper>

Sets the upper threshold value for the signal digitization. If the signal value is higher than the this threshold, the signal state is considered high.

Suffix:

<m>	1..4

Parameters:

<ThresholdUpper>
Range: 0 to 10
Increment: 0.01
*RST: 0.5
Default unit: V

BUS<m>:ETHernet:THReshold:LOW <ThresholdLower>

Sets the lower threshold value for the signal digitization. If the signal value is below this threshold, the signal state is considered low.

Suffix:

<m>	1..4

Parameters:

<ThresholdLower>
Range: -10 to 0
Increment: 0.01
*RST: -0.5
Default unit: V

BUS<m>:ETHernet:PRESet <ThresholdPreset>

Sets the thresholds to predefined or individually definable voltage levels.

Suffix:

<m>	1..4

Parameters:

<ThresholdPreset> T0 | T100 | TX0 | TX100 | MANual

T0
Sets the thresholds to the default values for 10BASE-T (0 meters): upper threshold to 1.25 V, lower threshold to -1.25 V

T100
Sets the thresholds to the default values for 10BASE-T (100 meters): upper threshold to 0.75 V, lower threshold to -0.75 V

TX0
Sets the thresholds to the default values for 100BASE-TX (0 meters): upper threshold to 0.5 V, lower threshold to -0.5 V

TX100
Sets the thresholds to the default values for 100BASE-TX (100 meters): upper threshold to 0.35 V, lower threshold to -0.35 V

MANual
Allows to set individual threshold voltage levels

*RST: T0

BUS<m>:ETHernet:BITRate <BitRateValue>

Sets the bit rate value that defines the transmission speed in bits per second.

Suffix:
<m> 1..4

Parameters:
<BitRateValue> Range: 10000 to 150000000
Increment: 1000
*RST: 10000000
Default unit: bps

17.17.12.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:ETHernet:TYPE.. 1614
TRIGger<m>:ETHernet:PATTern.. 1614
TRIGger<m>:ETHernet:ERRor:CRC.. 1614
TRIGger<m>:ETHernet:ERRor:LENGth.. 1615
TRIGger<m>:ETHernet:ERRor:PREamble.. 1615
TRIGger<m>:ETHernet:FRAMe:CCONdition.. 1615
TRIGger<m>:ETHernet:FRAMe:CMAX.. 1616
TRIGger<m>:ETHernet:FRAMe:CMIN.. 1616
TRIGger<m>:ETHernet:FRAMe:DCONdition.. 1616
TRIGger<m>:ETHernet:FRAMe:DMAX.. 1616

TRIGger<m>:ETHernet:FRAMe:DMIN....1617
TRIGger<m>:ETHernet:FRAMe:SCONdition....1617
TRIGger<m>:ETHernet:FRAMe:SMAX....1617
TRIGger<m>:ETHernet:FRAMe:SMIN....1617
TRIGger<m>:ETHernet:FRAMe:TCONdition....1618
TRIGger<m>:ETHernet:FRAMe:TMAX....1618
TRIGger<m>:ETHernet:FRAMe:TMIN....1618

TRIGger<m>:ETHernet:TYPE <Type>

Selects the type of frame to be triggered on.

Suffix:

<m> 1..3

Parameters:

<Type> STARt | PATTern | ADVFrame | ADVCerror

STARt
Selects to trigger for the start of frame: search for the preamble and set the trigger instant thereafter.

PATTern
Selects to trigger for a bit pattern (data) to be specified in `TRIGger<m>:ETHernet:PATTern` on page 1614.

ADVFrame
Sets the trigger to the advanced frame. You can further define the values of the frame.

ADVCerror
Sets the trigger to advanced error. You can further enable the error conditions you want to trigger on.

*RST: STARt

TRIGger<m>:ETHernet:PATTern <DataPattern>

Specifies the bit pattern (or data) that is to be triggered.

Suffix:

<m> 1..3

Parameters:

<DataPattern>

TRIGger<m>:ETHernet:ERRor:CRC <ErrorCRC>

Enables the trigger on a mismatch of the cyclic redundancy check (CRC) value between the transmitting and receiving device.

Suffix:

<m> 1..3

Parameters:

<ErrorCRC> ON | OFF

*RST: ON

TRIGger<m>:ETHernet:ERRor:LENGth <ErrorLength>

Enables the trigger on an incorrect length of the sequence - when additional or missing bits are detected and the sequence of bits is not as expected.

Suffix:

<m> 1..3

Parameters:

<ErrorLength> ON | OFF

*RST: ON

TRIGger<m>:ETHernet:ERRor:PREamble <ErrorPreamble>

Enables the trigger on a frame with invalid preamble.

Suffix:

<m> 1..3

Parameters:

<ErrorPreamble> ON | OFF

*RST: ON

TRIGger<m>:ETHernet:FRAMe:CCONdition <CRCOperator>

Sets the operator to trigger on a specific frame check sequence value.

Suffix:

<m> 1..3

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:CMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:CMIN` and `TRIGger<m>:ETHernet:FRAMe:CMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:CMAX <CRCPatternMax>

Sets the end value of the frame check sequence range, if `TRIGger<m>:ETHernet:FRAMe:CCONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMax>

TRIGger<m>:ETHernet:FRAMe:CMIN <CRCPatternMin>

Specifies a CRC pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMin>

TRIGger<m>:ETHernet:FRAMe:DCONdition <DestAddrOptor>

Sets the destination address operator to trigger on a specific pattern or a range.

Suffix:
<m> 1..3

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:DMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:DMIN` and `TRIGger<m>:ETHernet:FRAMe:DMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:DMAX <DestAddrPattMax>

Sets the end value of a destination address range, if `TRIGger<m>:ETHernet:FRAMe:DMAX` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMax>

TRIGger<m>:ETHernet:FRAMe:DMIN <DestAddrPattMin>

Specifies a destination address pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMin>

TRIGger<m>:ETHernet:FRAMe:SCONdition <SrcAddrOperator>

Sets source address operator to trigger on a specific pattern or a range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:SMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:SMIN` and `TRIGger<m>:ETHernet:FRAMe:SMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:SMAX <SrcAddrPattMax>

Sets the end value of a range, if the data condition is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMax>

TRIGger<m>:ETHernet:FRAMe:SMIN <SrcAddrPattMin>

Specifies a source address pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMin>

TRIGger<m>:ETHernet:FRAMe:TCONdition <TypeOperator>

Sets the type operator to trigger on a specific length/type value pattern or a range.

Suffix:
<m> 1..3

Parameters:
<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:TMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:TMIN` and `TRIGger<m>:ETHernet:FRAMe:TMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:TMAX <TypePatternMax>

Sets the end value of a length/type range, if `TRIGger<m>:ETHernet:FRAMe:TCONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<TypePatternMax>

TRIGger<m>:ETHernet:FRAMe:TMIN <TypePatternMin>

Specifies a length/type value, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<TypePatternMin>

17.17.12.3 Decode Results

In all `BUS<m>:ETHernet:WORD<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the word number in the decode table.

As an example, regarding Table 12-13, a set of query commands for bus #1 and word #1 is shown in the following, together with examples for results of these queries:

- `BUS1:ETH:WCOunt?` !2
- `BUS1:ETH:WORD1:STATe?` !OK

- `BUS1:ETH:WORD1:STARt?` !-0.000135
- `BUS1:ETH:WORD1:STOP?` !-6.62e-5
- `BUS1:ETH:WORD1:DEST?` !FF:FF:FF:FF:FF:FF
- `BUS1:ETH:WORD1:SRC?` !0F:0E:0D:0C:0B:0A
- `BUS1:ETH:WORD1:DATA?` ![60]45003c3e6210...
- `BUS1:ETH:WORD1:TYPE?` !2048
- `BUS1:ETH:WORD1:CRC?` !-1821935433
- `BUS1:ETH:WORD1:SSYM?` !
- `BUS1:ETH:WORD1:DSYM?` !BroadCast
- `BUS1:ETH:WORD1:BYTE1:VAL?` !69
- `BUS1:ETH:WORD1:BYTE2:VAL?` !0

BUS<m>:ETHernet:WCOunt?........ 1619
BUS<m>:ETHernet:WORD<n>:FTYPe?........ 1619
BUS<m>:ETHernet:WORD<n>:STATe?........ 1620
BUS<m>:ETHernet:WORD<n>:STARt?........ 1620
BUS<m>:ETHernet:WORD<n>:STOP?........ 1620
BUS<m>:ETHernet:WORD<n>:DESTaddress?........ 1621
BUS<m>:ETHernet:WORD<n>:SRCaddress?........ 1621
BUS<m>:ETHernet:WORD<n>:TYPE?........ 1621
BUS<m>:ETHernet:WORD<n>:DATA?........ 1622
BUS<m>:ETHernet:WORD<n>:CRC?........ 1622
BUS<m>:ETHernet:WORD<n>:DSYMbol?........ 1622
BUS<m>:ETHernet:WORD<n>:SSYMbol?........ 1623
BUS<m>:ETHernet:WORD<n>:BITRate?........ 1623
BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?........ 1623
BUS<m>:ETHernet:WORD<n>:NUMWords?........ 1624

BUS<m>:ETHernet:WCOunt?

Returns the frame count for the selected serial bus, i.e. the number of frames in the current acquisition. The result corresponds to the number of rows in the result table.

Suffix:

<m>	1..4

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:FTYPe?

Returns the frame type of the selected word in the current acquisition.

Suffix:

<m>	1..4

<n> *

Return values:

<FrameType> MAC | IDLE | SLEep | EOS | UNKNown

*RST: MAC

Usage: Query only

BUS<m>:ETHernet:WORD<n>:STATe?

Returns the frame state of the selected word in the current acquisition.

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | ERR_PREAMBLE | ERR_LENGTH | UNCorrelated | INComplete

OK
No error detected

ERR_PREAMBLE
Error in the preamble of the selected word

ERR_SFD
Error in the start frame delimiter (SFD). The value of a correct SFD byte is 171. The SFD is transmitted LSB first.

ERR_LENGTH
Error in the number of bits in the selected word

*RST: OK

Usage: Query only

BUS<m>:ETHernet:WORD<n>:STARt?

Returns the frame start time of the selected word in the current acquisition.

Suffix:

<m> 1..4

<n> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

BUS<m>:ETHernet:WORD<n>:STOP?

Returns the frame stop time of the selected word in the current acquisition.

Suffix:

<m>	1..4
<n>	*

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

BUS<m>:ETHernet:WORD<n>:DESTaddress?

Returns the destination address of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<DestAddress> String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:SRCaddress?

Returns the source address of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<SrcAddress> String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:TYPE?

The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of the TYPE field is ambivalent.

Suffix:

<m>	1..4
<n>	*

Return values:

<Type> Returns the word type (specific for the sub-protocol), or the length of the selected word.

Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:DATA?

Returns the number of word bytes in brackets [.] followed by the first six word bytes of data in hexadecimal format.

Use `BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?` to access the word bytes.

Suffix:

<m> 1..4

<n> *

Return values:

<Data> String parameter

Example:
```
BUS:ETHernet:WORD3:DATA?
<-- '[60]FF00FFFF1234'
```

Usage: Query only

BUS<m>:ETHernet:WORD<n>:CRC?

Returns the Cyclic Redundancy Code (CRC, or frame check) checksum of the selected word.

Suffix:

<m> 1..4

<n> *

Return values:

<CRC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:DSYMbol?

Returns the symbolic label (or translation) of the destination address of the specified word, if the label list is enabled.

Suffix:

<m> 1..4

<n> *

Return values:

<DestTranslation> String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:SSYMbol?

Returns the symbolic label (or translation) of the source address of the specified word, if the label list is enabled.

Suffix:

<m> 1..4

<n> *

Return values:

<SrcTranslation> String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4

<n> *

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?

BYTE returns all data of up to 1982 bytes (not just the first 5 or 6 bytes). This is also visible in the data table under "Show details".

Suffix:

<m> 1..4

<n> *

<o> *
Selects the byte number.

Return values:

<FrameByteValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:NUMWords?

Returns the number of words in the selected frame. The result corresponds to the "Number of Words" column in the results table.

Suffix:

<m> 1..4

<n> *
Frame index

Return values:

<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

17.17.12.4 Search Settings

SEARch:TRIGger:ETHernet:FRAMe:SELect....1624
SEARch:TRIGger:ETHernet:FRAMe:DCONdition....1625
SEARch:TRIGger:ETHernet:FRAMe:DMIN....1625
SEARch:TRIGger:ETHernet:FRAMe:DMAX....1625
SEARch:TRIGger:ETHernet:FRAMe:SCONdition....1626
SEARch:TRIGger:ETHernet:FRAMe:SMIN....1626
SEARch:TRIGger:ETHernet:FRAMe:SMAX....1626
SEARch:TRIGger:ETHernet:FRAMe:TCONdition....1627
SEARch:TRIGger:ETHernet:FRAMe:TMIN....1627
SEARch:TRIGger:ETHernet:FRAMe:TMAX....1627
SEARch:TRIGger:ETHernet:FRAMe:CCONdition....1628
SEARch:TRIGger:ETHernet:FRAMe:CMIN....1628
SEARch:TRIGger:ETHernet:FRAMe:CMAX....1628
SEARch:TRIGger:ETHernet:ERRor:SELect....1629
SEARch:TRIGger:ETHernet:ERRor:PREamble....1629
SEARch:TRIGger:ETHernet:ERRor:LENGth....1629

SEARch:TRIGger:ETHernet:FRAMe:SELect <SearchName>,<CheckFrame>
SEARch:TRIGger:ETHernet:FRAMe:SELect? <SearchName>

Defines, whether a search within a frame shall be activated or not.

Parameters:

<CheckFrame> ON | OFF
*RST: OFF

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:DCONdition <SearchName>,<DestAddrOptor>
SEARch:TRIGger:ETHernet:FRAMe:DCONdition? <SearchName>

Defines the operator to search a specific destination address within a frame.

Parameters:

<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `SEARch:TRIGger:ETHernet:FRAMe:DMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:DMIN` and `SEARch:TRIGger:ETHernet:FRAMe:DMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:DMIN <SearchName>,<DestAddrPattMin>
SEARch:TRIGger:ETHernet:FRAMe:DMIN? <SearchName>

Defines a destination address, or sets the start value of a destination address range.

Parameters:

<DestAddrPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:DMAX <SearchName>,<DestAddrPattMax>
SEARch:TRIGger:ETHernet:FRAMe:DMAX? <SearchName>

Sets the end value of a destination address range, if `SEARch:TRIGger:ETHernet:FRAMe:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DestAddrPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:SCONdition <SearchName>,<SrcAddrOperator>
SEARch:TRIGger:ETHernet:FRAMe:SCONdition? <SearchName>

Defines the operator to search a specific source address within a frame.

Parameters:

<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `SEARch:TRIGger:ETHernet:FRAMe:SMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:SMIN` and `SEARch:TRIGger:ETHernet:FRAMe:SMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:SMIN <SearchName>,<SrcAddrPattMin>
SEARch:TRIGger:ETHernet:FRAMe:SMIN? <SearchName>

Defines a source address, or sets the start value of a source address range.

Parameters:

<SrcAddrPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:SMAX <SearchName>,<SrcAddrPattMax>
SEARch:TRIGger:ETHernet:FRAMe:SMAX? <SearchName>

Sets the end value of a source address range, if `SEARch:TRIGger:ETHernet:FRAMe:SCONdition` is set to `INRange` or `OORange`.

Parameters:

<SrcAddrPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:TCONdition <SearchName>,<TypeOperator>
SEARch:TRIGger:ETHernet:FRAMe:TCONdition? <SearchName>

Defines the operator to search for a specific frame length or type.

Parameters:

<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a pattern to be set with `SEARch:TRIGger:ETHernet:FRAMe:TMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:TMIN` and `SEARch:TRIGger:ETHernet:FRAMe:TMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:TMIN <SearchName>,<TypePatternMin>
SEARch:TRIGger:ETHernet:FRAMe:TMIN? <SearchName>

Defines a frame length/type, or sets the start value for a range of frame lengths/types.

Parameters:

<TypePatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:TMAX <SearchName>,<TypePatternMax>
SEARch:TRIGger:ETHernet:FRAMe:TMAX? <SearchName>

Sets the end value of a range of frame lengths/types, if `SEARch:TRIGger:ETHernet:FRAMe:TCONdition` is set to `INRange` or `OORange`.

Parameters:

<TypePatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:CCONdition <SearchName>,<CRCOperator>
SEARch:TRIGger:ETHernet:FRAMe:CCONdition? <SearchName>

Defines the operator to search for a Cyclic Redundancy Code (CRC, or frame check) error condition within a frame.

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a CRC pattern to be set with SEARch:TRIGger:ETHernet:FRAMe:CMIN.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with SEARch:TRIGger:ETHernet:FRAMe:CMIN and SEARch:TRIGger:ETHernet:FRAMe:CMAX.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:CMIN <SearchName>,<CRCPatternMin>
SEARch:TRIGger:ETHernet:FRAMe:CMIN? <SearchName>

Defines a CRC error condition pattern, or sets the start value of such a pattern.

Parameters:

<CRCPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:CMAX <SearchName>,<CRCPatternMax>
SEARch:TRIGger:ETHernet:FRAMe:CMAX? <SearchName>

Sets the end value of a CRC error condition pattern, if SEARch:TRIGger:ETHernet:FRAMe:CCONdition is set to INRange or OORange.

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:ERRor:SELect <SearchName>,<ErrorCondition>
SEARch:TRIGger:ETHernet:ERRor:SELect? <SearchName>

Defines, whether a search for an error condition shall be activated or not.

Parameters:
<ErrorCondition> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> String parameter

SEARch:TRIGger:ETHernet:ERRor:PREamble <SearchName>,<ErrorPreamble>
SEARch:TRIGger:ETHernet:ERRor:PREamble? <SearchName>

Defines, whether a search for any preamble error shall be activated or not.

Parameters:
<ErrorPreamble> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

SEARch:TRIGger:ETHernet:ERRor:LENGth <SearchName>,<ErrorLength>
SEARch:TRIGger:ETHernet:ERRor:LENGth? <SearchName>

Defines, whether a search for any word length error (too few or too many bits per word) shall be activated or not.

Parameters:
<ErrorLength> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

17.17.12.5 Search Results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1383
- `BUS<m>:RESDetail` on page 1384

In all `SEARch:RESult:ETHernet:WORD<m>` commands, the suffix <m> selects the word number in the list of search results.

SEARch:RESult:ETHernet:WCOunt?.. 1630
SEARch:RESult:ETHernet:WORD<m>:STATe?.. 1630
SEARch:RESult:ETHernet:WORD<m>:STARt?.. 1630
SEARch:RESult:ETHernet:WORD<m>:STOP?... 1631
SEARch:RESult:ETHernet:WORD<m>:DESTaddress?.. 1631

SEARch:RESult:ETHernet:WORD<m>:SRCaddress?..1631
SEARch:RESult:ETHernet:WORD<m>:TYPE?... 1632
SEARch:RESult:ETHernet:WORD<m>:FTYPe?..1632
SEARch:RESult:ETHernet:WORD<m>:DATA?..1632
SEARch:RESult:ETHernet:WORD<m>:CRC?..1633
SEARch:RESult:ETHernet:WORD<m>:DSYMbol?... 1633
SEARch:RESult:ETHernet:WORD<m>:SSYMbol?.. 1633
SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue?... 1634

SEARch:RESult:ETHernet:WCOunt? <SearchName>

Returns the number of decoded words within the search result.

Query parameters:
<SearchName> String parameter

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:STATe? <SearchName>

Returns the frame state of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<State> OK | ERR_PREAMBLE | ERR_LENGTH

OK
No error detected

ERR_PREAMBLE
Error in the preamble of the selected word

ERR_SFD
Error in the start frame delimiter (SFD). The value of a correct SFD byte is 171. The SFD is transmitted LSB first.

ERR_LENGTH
Error in the number of bits in the selected word

*RST: OK

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:STARt? <SearchName>

Returns the frame start time of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:STOP? <SearchName>

Returns the frame stop time of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:DESTaddress? <SearchName>

Returns the destination address of the specified word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<DestAddress> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:SRCaddress? <SearchName>

Returns the source address of the specified word within the search result.

Suffix:
<m> *

Query parameters:

<SearchName> String parameter

Return values:

<SourceAddress> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:TYPE? <SearchName>

The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of the TYPE field is ambivalent. The query either returns the word type (specific for the sub-protocol), or the length of the selected word within the search result.

Suffix:

<m> *

Query parameters:

<SearchName> String parameter

Return values:

<Type> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:FTYPe? <SearchName>

Returns the frame type of the specified frame.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameType> MAC | IDLE | SLEep | EOS | UNKNown
*RST: MAC

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:DATA? <SearchName>

Returns the data bytes of the specified word within the search result.

Suffix:

<m> *

Query parameters:

<SearchName> String parameter

Return values:

<Data>	String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:CRC? <SearchName>

Returns the Cyclic Redundancy Code (CRC, or frame check) checksum of the selected word within the search result.

Suffix:

<m>	*

Query parameters:

<SearchName>	String parameter

Return values:

<CRC>	Range: 0 to 4294967295 Increment: 1 *RST: 0

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:DSYMbol? <SearchName>

Returns the symbolic label (or translation) of the destination address of the specified word within the search result, if the label list is enabled.

Suffix:

<m>	*

Query parameters:

<SearchName>	String parameter

Return values:

<DestTranslation>	String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:SSYMbol? <SearchName>

Returns the symbolic label (or translation) of the source address of the specified word within the search result, if the label list is enabled.

Suffix:

<m>	*

Query parameters:

<SearchName>	String parameter

Return values:

<SrcTranslation>	String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue? <SearchName>

BYTE returns all data of up to 1982 bytes (not just the first 5 or 6 bytes). This is also visible in the data table under "Show details".

Suffix:

<m>	*
<n>	* Selects the byte number.

Query parameters:

<SearchName>	String parameter

Return values:

<FrameByteValue>	Range: 0 to 255 Increment: 1 *RST: 0

Usage: Query only

17.17.13 Ethernet 100BASE-T1 (Option R&S RTE-K57)

- Configuration....1634
- Trigger....1638
- Decode Results....1645
- Search Settings....1651
- Search Results....1659

17.17.13.1 Configuration

In all `BUS<m>:HBTO` commands, the suffix <m> selects the serial bus.

BUS<m>:HBTO:ATTN....1634
BUS<m>:HBTO:FDAM....1635
BUS<m>:HBTO:FDAP....1635
BUS<m>:HBTO:FDIF....1635
BUS<m>:HBTO:FTYP....1636
BUS<m>:HBTO:RDAM....1636
BUS<m>:HBTO:RDAP....1636
BUS<m>:HBTO:RDIF....1636
BUS<m>:HBTO:RTYP....1636
BUS<m>:HBTO:THReshold....1637
BUS<m>:HBTO:POLarity....1637
BUS<m>:HBTO:MODE....1637
BUS<m>:HBTO:SYMRate....1637

BUS<m>:HBTO:ATTN <Attenuation>

Sets the attenuation factor. It is used to de-amplify the reverse signal before subtracting it from the forward signal.

Suffix:
<m> 1..4

Parameters:
<Attenuation> Range: -100 to 0
Increment: 0.1
*RST: -26
Default unit: dB

BUS<m>:HBTO:FDAM <SourceDAminus>

Selects the DA- source of the provided forward single ended signal, if `BUS<m>:HBTO:FTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceDAminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C2W1

BUS<m>:HBTO:FDAP <SourceDAplus>

Selects the DA+ source of the provided forward single ended signal, if `BUS<m>:HBTO:FTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceDAplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:HBTO:FDIF <SrcDiff>

Sets the source of the provided forward differential signal, if `BUS<m>:HBTO:FTYP` is set to `DIFFerential`.

Suffix:
<m> 1..4

Parameters:
<SrcDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:HBTO:FTYP <SignalType>

Sets the type of forward signal measurement.

Suffix:

<m> 1..4

Parameters:

<SignalType> DIFFerential | SINGle

*RST: DIFFerential

BUS<m>:HBTO:RDAM <SrcRevDAminus>

Selects the DA- source of the provided reversed single ended signal, if BUS<m>:HBTO:RTYP is set to SINGle.

Suffix:

<m> 1..4

Parameters:

<SrcRevDAminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:HBTO:RDAP <SourceRevDAplus>

Selects the DA+ source of the provided reversed single ended signal, if BUS<m>:HBTO:RTYP is set to SINGle.

Suffix:

<m> 1..4

Parameters:

<SourceRevDAplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:HBTO:RDIF <SrcRevDiff>

Sets the source of the provided reversed differential signal, if BUS<m>:HBTO:RTYP is set to DIFFerential.

Suffix:

<m> 1..4

Parameters:

<SrcRevDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:HBTO:RTYP <SignalRevType>

Sets the type of reverse signal measurement.

Suffix:

<m> 1..4

Parameters:

<SignalRevType> DISabled | DIFFerential | SINGle

*RST: DISabled

BUS<m>:HBTO:THReshold <ThresholdUpper>

Sets an user-defined threshold value.

Suffix:

<m> 1..4

Parameters:

<ThresholdUpper> Range: 0 to 2

Increment: 1E-3

*RST: 0.4

Default unit: V

BUS<m>:HBTO:POLarity <Polarity>

Selects the polarity of the data signal.

Suffix:

<m> 1..4

Parameters:

<Polarity> NORMal | INVert

*RST: NORMal

BUS<m>:HBTO:MODE <Mode>

Selects the direction of the full-duplex signal you want to look at.

Suffix:

<m> 1..4

Parameters:

<Mode> MASTer | SLAVe | AUTO

*RST: AUTO

BUS<m>:HBTO:SYMRate <SymbolRate>

Defines the rate of ternary symbols.

Suffix:

<m> 1..4

Parameters:

<SymbolRate>	Range:	10 to 150
	Increment:	1E-3
	*RST:	66.6667
	Default unit:	MSymb/s

17.17.13.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:HBTO:TYPE....1638
TRIGger<m>:HBTO:DADDress:CONDition....1639
TRIGger<m>:HBTO:DADDress:MIN....1639
TRIGger<m>:HBTO:DADDress:MAX....1640
TRIGger<m>:HBTO:SADDress:CONDition....1640
TRIGger<m>:HBTO:SADDress:MIN....1640
TRIGger<m>:HBTO:SADDress:MAX....1641
TRIGger<m>:HBTO:LENGth:CONDition....1641
TRIGger<m>:HBTO:LENGth:MIN....1641
TRIGger<m>:HBTO:LENGth:MAX....1641
TRIGger<m>:HBTO:CRC:CONDition....1642
TRIGger<m>:HBTO:CRC:MIN....1642
TRIGger<m>:HBTO:CRC:MAX....1642
TRIGger<m>:HBTO:DATA:DCONdition....1643
TRIGger<m>:HBTO:DATA:DMIN....1643
TRIGger<m>:HBTO:DATA:DMAX....1643
TRIGger<m>:HBTO:DATA:ICONdition....1644
TRIGger<m>:HBTO:DATA:IMIN....1644
TRIGger<m>:HBTO:DATA:IMAX....1644
TRIGger<m>:HBTO:ERRor:PREamble....1644
TRIGger<m>:HBTO:ERRor:CRC....1645
TRIGger<m>:HBTO:ERRor:SFD....1645

TRIGger<m>:HBTO:TYPE <Type>

Selects the type of frame to be triggered on.

Suffix:

<m>	1..3

Parameters:

<Type>	STARt | MAC | IDLE | ERRor

STARt
Start of frame.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

IDLE
IDLE frame. This frame is used for clock synchronization.

ERRor
Error frame. Thi frame contains erroneous bits.

*RST: STARt

TRIGger<m>:HBTO:DADDress:CONDition <DestAddrOptor>

Sets the condition for the destination address. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:DADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:DADDress:MIN` and `TRIGger<m>:HBTO:DADDress:MAX`.

*RST: EQUal

TRIGger<m>:HBTO:DADDress:MIN <DestAddrPattMin>

Specifies the destination address bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DADDress:MAX <DestAddrPattMax>

Sets the end value of the destination address range if TRIGger<m>:HBTO:DADDress:CONDition is set to INRange or OORange.

Suffix:

<m>	1..3

Parameters:

<DestAddrPattMax>	Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:SADDress:CONDition <SrcAddrOperator>

Sets the condition for the source address. You can define an exact value or a value range.

Suffix:

<m>	1..3

Parameters:

<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding TRIGger<m>:HBTO:SADDress:MIN command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with TRIGger<m>:HBTO:SADDress:MIN and TRIGger<m>:HBTO:SADDress:MAX.

*RST: EQUal

TRIGger<m>:HBTO:SADDress:MIN <SrcAddrPattMin>

Specifies the source address bit pattern, or sets the start value of a pattern range.

Suffix:

<m>	1..3

Parameters:

<SrcAddrPattMin>	Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:SADDress:MAX <SrcAddrPattMax>

Sets the end value of the source address range if TRIGger<m>:HBTO:SADDress:CONDition is set to INRange or OORange.

Suffix:

<m> 1..3

Parameters:

<SrcAddrPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:LENGth:CONDition <TypeOperator>

Sets the condition for the length / type. You can define an exact value or a value range.

Suffix:

<m> 1..3

Parameters:

<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding TRIGger<m>:HBTO:LENGth:MIN command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with TRIGger<m>:HBTO:LENGth:MIN and TRIGger<m>:HBTO:LENGth:MAX.

*RST: EQUal

TRIGger<m>:HBTO:LENGth:MIN <TypePatternMin>

Specifies the length / type bit pattern, or sets the start value of a pattern range.

Suffix:

<m> 1..3

Parameters:

<TypePatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:LENGth:MAX <TypePatternMax>

Sets the end value of the length / type range if TRIGger<m>:HBTO:LENGth:CONDition is set to INRange or OORange.

Suffix:
<m> 1..3

Parameters:
<TypePatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:CRC:CONDition <CRCOperator>

Sets the condition for the frame check. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:CRC:MIN` and `TRIGger<m>:HBTO:CRC:MAX`.

*RST: EQUal

TRIGger<m>:HBTO:CRC:MIN <CRCPatternMin>

Specifies the frame check bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:CRC:MAX <CRCPatternMax>

Sets the end value of the frame check range if `TRIGger<m>:HBTO:CRC:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DATA:DCONdition <DataOperator>

Sets the condition for the data. You can define an exact value or a value range.

Suffix:

<m> 1..3

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:DATA:DMIN` and `TRIGger<m>:HBTO:DATA:DMAX`.

*RST: EQUal

TRIGger<m>:HBTO:DATA:DMIN <DataMin>

Specifies the data bit pattern, or sets the start value of a pattern range.

Suffix:

<m> 1..3

Parameters:

<DataMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DATA:DMAX <DataMax>

Sets the end value of the data range if `TRIGger<m>:HBTO:DATA:DCONdition` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3

Parameters:

<DataMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DATA:ICONdition <DataIdxOperator>

Sets the condition for the index. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:DATA:IMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:DATA:IMIN` and `TRIGger<m>:HBTO:DATA:IMAX`.

*RST: INRange

TRIGger<m>:HBTO:DATA:IMIN <DataIndexMin>

Specifies the index bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataIndexMin> Range: 1 to 65535
Increment: 1
*RST: 1

TRIGger<m>:HBTO:DATA:IMAX <DataIndexMax>

Sets the end value of the index range if `TRIGger<m>:HBTO:DATA:ICONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataIndexMax> Range: 1 to 65535
Increment: 1
*RST: 0

TRIGger<m>:HBTO:ERRor:PREamble <ErrorPreamble>

Enables / disables trigger on preamble error.

Suffix:
<m> 1..3

Parameters:
<ErrorPreamble> ON | OFF
*RST: ON

TRIGger<m>:HBTO:ERRor:CRC <ErrorCRC>

Enables / disables trigger on Cyclic Redundancy Check (CRC) error.

Suffix:
<m> 1..3

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

TRIGger<m>:HBTO:ERRor:SFD <ErrorSFD>

Enables / disables trigger on start frame delimiter (SFD) error.

Suffix:
<m> 1..3

Parameters:
<ErrorSFD> ON | OFF
*RST: ON

17.17.13.3 Decode Results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1383
- `BUS<m>:RESDetail` on page 1384

In all `BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>` commands, the suffix <m> selects the serial bus, suffix <n> selects the frame in the decode table and suffix <o> selects the word in the selected frame.

BUS<m>:HBTO:RESult:FCOunt?....... 1646
BUS<m>:HBTO:RESult:FRAMe<n>:FTYPe?....... 1646
BUS<m>:HBTO:RESult:FRAMe<n>:STATe?....... 1646
BUS<m>:HBTO:RESult:FRAMe<n>:STARt?....... 1647
BUS<m>:HBTO:RESult:FRAMe<n>:STOP?....... 1647
BUS<m>:HBTO:RESult:FRAMe<n>:DESTaddress?....... 1648
BUS<m>:HBTO:RESult:FRAMe<n>:SRCaddress?....... 1648
BUS<m>:HBTO:RESult:FRAMe<n>:TYPE?....... 1648
BUS<m>:HBTO:RESult:FRAMe<n>:DATA?....... 1648
BUS<m>:HBTO:RESult:FRAMe<n>:CRC?....... 1649
BUS<m>:HBTO:RESult:FRAMe<n>:NUMWords?....... 1649
BUS<m>:HBTO:RESult:FRAMe<n>:DSYMbol?....... 1649

BUS<m>:HBTO:RESult:FRAMe<n>:SSYMbol?....1649
BUS<m>:HBTO:RESult:FRAMe<n>:BITRate?....1650
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:TYPE?....1650
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VALue?....1650
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VSTR?....1651

BUS<m>:HBTO:RESult:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:FTYPe?

Returns the type of frame for the selected frame.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameType> IDLE | MAC | FILLer | UNKNown

IDLE
IDLE frame. This frame is used for clock synchronization.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

FILLer
Filler frame. The frame is used to maintain transmission activity.

UNKNown
No meaningful frame can be determined.

*RST: MAC

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:STATe?

Returns the state of the frame.

Suffix:

<m> 1..4

<n> *

Return values:
<State> OK | ERR_PREAMBLE | ERR_SFD | ERR_LENGTH | ERR_CRC | UNCorrelated | INComplete

OK
Valid frame.

ERR_PREAMBLE
Erroneous frame due to preamble error.

ERR_SFD
Erroneous frame due to SFD error.

ERR_LENGTH
Erroneous frame due to length / type error.

ERR_CRC
Erroneous frame due to CRC error.

*RST: OK

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:STARt?

Returns the start time of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:STOP?

Returns the end time of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:DESTaddress?

Returns the destination address of the selected frame.

Suffix:

<m>	1..4
<n>	*

Return values:
<DestAddress>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:SRCaddress?

Returns the source address of the selected frame.

Suffix:

<m>	1..4
<n>	*

Return values:
<SrcAddress>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:TYPE?

Returns the value of length / type field of the selected frame.

Suffix:

<m>	1..4
<n>	*

Return values:

<TypeLen>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:DATA?

Returns the data for the selected frame, corresponds to the Data column in the decode results table.

Suffix:

<m>	1..4
<n>	*

Return values:
<Data>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:CRC?

Returns the CRC (frame check) checksum of the selected frame.

Suffix:

<m> 1..4

<n> *

Return values:

<CRC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:NUMWords?

Returns the number of decoded words for the selected frame.

Suffix:

<m> 1..4

<n> *

Return values:

<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:DSYMbol?

Returns the destination symbols of the selected frame.

Suffix:

<m> 1..4

<n> *

Return values:

<DestTranslation>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:SSYMbol?

Returns the source symbols of the selected frame.

Suffix:

<m> 1..4

<n>	*

Return values:
<SrcTranslation>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m>	1..4
<n>	*

Return values:

<PrimaryBitRate>	Range:	0 to 100000000000
	Increment:	1
	*RST:	0
	Default unit:	bps

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:TYPE?

Returns the data type for the selected word.

Suffix:

<m>	1..4
<n>	*
<o>	*

Return values:
<WordType>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VALue?

Returns the data value for the selected word.

Suffix:

<m>	1..4
<n>	*
<o>	*

Return values:

<WordValue>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VSTR?

Returns the string equivalent of data value for the selected cell.

Suffix:

<m>	1..4
<n>	*
<o>	*

Return values:

<WordValueString>	String parameter

Usage: Query only

17.17.13.4 Search Settings

SEARch:TRIGger:HBTO:TYPE........ 1651
SEARch:TRIGger:HBTO:DADDress:CONDition........ 1652
SEARch:TRIGger:HBTO:DADDress:MIN........ 1652
SEARch:TRIGger:HBTO:DADDress:MAX........ 1653
SEARch:TRIGger:HBTO:SADDress:CONDition........ 1653
SEARch:TRIGger:HBTO:SADDress:MIN........ 1653
SEARch:TRIGger:HBTO:SADDress:MAX........ 1654
SEARch:TRIGger:HBTO:LENGth:CONDition........ 1654
SEARch:TRIGger:HBTO:LENGth:MIN........ 1654
SEARch:TRIGger:HBTO:LENGth:MAX........ 1655
SEARch:TRIGger:HBTO:CRC:CONDition........ 1655
SEARch:TRIGger:HBTO:CRC:MIN........ 1655
SEARch:TRIGger:HBTO:CRC:MAX........ 1656
SEARch:TRIGger:HBTO:DATA:DCONdition........ 1656
SEARch:TRIGger:HBTO:DATA:DMIN........ 1656
SEARch:TRIGger:HBTO:DATA:DMAX........ 1657
SEARch:TRIGger:HBTO:DATA:ICONdition........ 1657
SEARch:TRIGger:HBTO:DATA:IMIN........ 1657
SEARch:TRIGger:HBTO:DATA:IMAX........ 1658
SEARch:TRIGger:HBTO:ERRor:PREamble........ 1658
SEARch:TRIGger:HBTO:ERRor:CRC........ 1658
SEARch:TRIGger:HBTO:ERRor:SFD........ 1658

SEARch:TRIGger:HBTO:TYPE <SearchName>,<Type>
SEARch:TRIGger:HBTO:TYPE? <SearchName>

Selects the type of frame to be searched for.

Parameters:

<Type> STARt | MAC | IDLE | ERRor

STARt
Start of frame.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

IDLE
IDLE frame. This frame is used for clock synchronization.

ERRor
Error frame. Thi frame contains erroneous bits.

*RST: STARt

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DADDress:CONDition <SearchName>,<DestAddrOptor>
SEARch:TRIGger:HBTO:DADDress:CONDition? <SearchName>

Sets the condition for the destination address. You can define an exact value or a value range.

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:DADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:DADDress:MIN` and `SEARch:TRIGger:HBTO:DADDress:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DADDress:MIN <SearchName>,<DestAddrPattMin>
SEARch:TRIGger:HBTO:DADDress:MIN? <SearchName>

Specifies the destination address bit pattern, or sets the start value of a pattern range.

Parameters:
<DestAddrPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DADDress:MAX <SearchName>,<DestAddrPattMax>
SEARch:TRIGger:HBTO:DADDress:MAX? <SearchName>

Sets the end value of the destination address range if `SEARch:TRIGger:HBTO:DADDress:CONDition` is set to `INRange` or `OORange`.

Parameters:

<DestAddrPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:SADDress:CONDition <SearchName>,<SrcAddrOperator>
SEARch:TRIGger:HBTO:SADDress:CONDition? <SearchName>

Sets the condition for the source address. You can define an exact value or a value range.

Parameters:

<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:SADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:SADDress:MIN` and `SEARch:TRIGger:HBTO:SADDress:MAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:SADDress:MIN <SearchName>,<SrcAddrPattMin>
SEARch:TRIGger:HBTO:SADDress:MIN? <SearchName>

Specifies the source address bit pattern, or sets the start value of a pattern range.

Parameters:

<SrcAddrPattMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:SADDress:MAX <SearchName>,<SrcAddrPattMax>
SEARch:TRIGger:HBTO:SADDress:MAX? <SearchName>

Sets the end value of the source address range if `SEARch:TRIGger:HBTO:SADDress:CONDition` is set to `INRange` or `OORange`.

Parameters:

<SrcAddrPattMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:LENGth:CONDition <SearchName>,<TypeOperator>
SEARch:TRIGger:HBTO:LENGth:CONDition? <SearchName>

Sets the condition for the length / type. You can define an exact value or a value range.

Parameters:

<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:LENGth:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:LENGth:MIN` and `SEARch:TRIGger:HBTO:LENGth:MAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:LENGth:MIN <SearchName>,<TypePatternMin>
SEARch:TRIGger:HBTO:LENGth:MIN? <SearchName>

Specifies the length / type bit pattern, or sets the start value of a pattern range.

Parameters:

<TypePatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:LENGth:MAX <SearchName>,<TypePatternMax>
SEARch:TRIGger:HBTO:LENGth:MAX? <SearchName>

Sets the end value of the length / type range if `SEARch:TRIGger:HBTO:LENGth:CONDition` is set to `INRange` or `OORange`.

Parameters:

<TypePatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:CRC:CONDition <SearchName>,<CRCOperator>
SEARch:TRIGger:HBTO:CRC:CONDition? <SearchName>

Sets the condition for the frame check. You can define an exact value or a value range.

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:CRC:MIN` and `SEARch:TRIGger:HBTO:CRC:MAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:CRC:MIN <SearchName>,<CRCPatternMin>
SEARch:TRIGger:HBTO:CRC:MIN? <SearchName>

Specifies the frame check bit pattern, or sets the start value of a pattern range.

Parameters:

<CRCPatternMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:CRC:MAX <SearchName>,<CRCPatternMax>
SEARch:TRIGger:HBTO:CRC:MAX? <SearchName>

Sets the end value of the frame check range if `SEARch:TRIGger:HBTO:CRC:CONDition` is set to `INRange` or `OORange`.

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:HBTO:DATA:DCONdition? <SearchName>

Sets the condition for the data. You can define an exact value or a value range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:DATA:DMIN` and `SEARch:TRIGger:HBTO:DATA:DMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:DATA:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:HBTO:DATA:DMIN? <SearchName>

Specifies the data bit pattern, or sets the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:DATA:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:HBTO:DATA:DMAX? <SearchName>

Sets the end value of the data range if `SEARch:TRIGger:HBTO:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:HBTO:DATA:ICONdition? <SearchName>

Sets the condition for the index. You can define an exact value or a value range.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:DATA:IMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:DATA:IMIN` and `SEARch:TRIGger:HBTO:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:HBTO:DATA:IMIN? <SearchName>

Specifies the index bit pattern, or sets the start value of a pattern range.

Parameters:

<DataIndexMin>

Range:	1 to 0
Increment:	1
*RST:	1

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:HBTO:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:HBTO:DATA:IMAX? <SearchName>

Sets the end value of the index range if `SEARch:TRIGger:HBTO:DATA:ICONdition` is set to `INRange` or `OORange`.

Parameters:

<DataIndexMax>	Range:	1 to 0
	Increment:	1
	*RST:	0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:ERRor:PREamble <SearchName>,<ErrorPreamble>
SEARch:TRIGger:HBTO:ERRor:PREamble? <SearchName>

Enables / disables search for preamble error.

Parameters:

<ErrorPreamble>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:ERRor:CRC <SearchName>,<ErrorCRC>
SEARch:TRIGger:HBTO:ERRor:CRC? <SearchName>

Enables / disables trigger on Cyclic Redundancy Check (CRC) error.

Parameters:

<ErrorCRC>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:ERRor:SFD <SearchName>,<ErrorSFD>
SEARch:TRIGger:HBTO:ERRor:SFD? <SearchName>

Enables / disables search for start frame delimiter (SFD) error.

Parameters:

<ErrorSFD>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

17.17.13.5 Search Results

In all `SEARch:RESult:HBTO:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:HBTO:FCOunt?....1659
SEARch:RESult:HBTO:FRAMe<m>:TYPE?....1659
SEARch:RESult:HBTO:FRAMe<m>:STATe?....1660
SEARch:RESult:HBTO:FRAMe<m>:STARt?....1660
SEARch:RESult:HBTO:FRAMe<m>:STOP?....1660
SEARch:RESult:HBTO:FRAMe<m>:DESTaddress?....1661
SEARch:RESult:HBTO:FRAMe<m>:SRCaddress?....1661
SEARch:RESult:HBTO:FRAMe<m>:DATA?....1661
SEARch:RESult:HBTO:FRAMe<m>:CRC?....1662
SEARch:RESult:HBTO:FRAMe<m>:NUMWords?....1662
SEARch:RESult:HBTO:FRAMe<m>:FTYPe?....1662
SEARch:RESult:HBTO:FRAMe<m>:DSYMbol?....1663
SEARch:RESult:HBTO:FRAMe<m>:SSYMbol?....1663
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE?....1663
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue?....1664

SEARch:RESult:HBTO:FCOunt? <SearchName>

Returns the number of decoded frames within the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:TYPE? <SearchName>

Returns the value of length / type field of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<TypeLen>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:STATe? <SearchName>

Returns the state of the frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | ERR_PREAMBLE | ERR_SFD | ERR_LENGTH | ERR_CRC | UNCorrelated | INComplete

OK
Valid frame.

ERR_PREAMBLE
Erroneous frame due to preamble error.

ERR_SFD
Erroneous frame due to SFD error.

ERR_LENGTH
Erroneous frame due to length / type error.

ERR_CRC
Erroneous frame due to CRC error.

*RST: OK

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:STARt? <SearchName>

Returns the start time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:STOP? <SearchName>

Returns the end time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:DESTaddress? <SearchName>

Returns the destination address of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DestAddress>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:SRCaddress? <SearchName>

Returns the source address of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SourceAddress>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:DATA? <SearchName>

Returns the data for the selected frame, corresponds to the Data column in the decode results table.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:CRC? <SearchName>

Returns the CRC checksum of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CRC>
Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:NUMWords? <SearchName>

Returns the number of decoded words for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<NumWords>
Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:FTYPe? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> IDLE | MAC | FILLer | UNKNown

IDLE
IDLE frame. This frame is used for clock synchronization.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

FILLer
Filler frame. The frame is used to maintain transmission activity.

UNKNown
No meaningful frame can be determined.

*RST: MAC

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:DSYMbol? <SearchName>

Returns the destination symbols of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DestTranslation>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:SSYMbol? <SearchName>

Returns the source symbols of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SrcTranslation>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE? <SearchName>

Returns the data type of the selected word within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordType> String parameter

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue? <SearchName>

Returns the data value of the selected word within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

17.17.14 SENT (Option R&S RTE-K10)

- Configuration....1664
- Trigger....1668
- Decode Results....1674
- SENT Search Settings....1681
- SENT Search Results....1689

17.17.14.1 Configuration

BUS<m>:SENT:DATA:SOURce....1664
BUS<m>:SENT:DATA:THReshold....1665
BUS<m>:SENT:TECHnology....1665
BUS<m>:SENT:CLKPeriod....1665
BUS<m>:SENT:CLKTolerance....1666
BUS<m>:SENT:DNIBbles....1666
BUS<m>:SENT:SFORmat....1666
BUS<m>:SENT:CRCVersion....1666
BUS<m>:SENT:CRCMethod....1667
BUS<m>:SENT:PPULse....1667
BUS<m>:SENT:PPFLength....1667

BUS<m>:SENT:DATA:SOURce <DataSource>

Selects the source of the data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:SENT:DATA:THReshold <Threshold>

Sets a user-defined threshold value. Alternatively, you can set the threshold according to the signal technology `BUS<m>:SENT:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Threshold> Range: -12 to 12
Increment: 0.1
*RST: 1.65
Default unit: V

BUS<m>:SENT:TECHnology <Technology>

Selects the threshold voltage.

Suffix:

<m> 1..4
Selects the serial data bus.

Parameters:

<Technology> V25 | MANual

V25
The threshold value is 2.5 V, according to CMOS technology.

MANual
Sets the threshold to the value set with `BUS<m>:SENT:DATA:THReshold`.

*RST: V25

BUS<m>:SENT:CLKPeriod <ClockPeriod>

Sets the nominal clock period (clock tick).

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ClockPeriod>
Range: 1E-6 to 100E-6
Increment: 1E-6
*RST: 6E-6
Default unit: s

BUS<m>:SENT:CLKTolerance <ClockTolerance>

Sets a tolerated deviation of the clock signal.

Suffix:

<m>
1..4
Selects the serial bus.

Parameters:

<ClockTolerance>
Range: 0 to 25
Increment: 1
*RST: 20
Default unit: %

BUS<m>:SENT:DNIBbles <DataNibbles>

Sets the number of data nibbles for a transmission sequence.

Suffix:

<m>
1..4
Selects the serial data bus.

Parameters:

<DataNibbles>
Range: 1 to 6
Increment: 1
*RST: 6

BUS<m>:SENT:SFORmat <Format>

Selects the serial message format.

Suffix:

<m>
1..4

Parameters:

<Format>
SHORt | ENHanced | NONE

Short serial message, Enhanced serial message, none = single transmission sequence.

*RST: NONE

BUS<m>:SENT:CRCVersion <CRCVersion>

Selects the calculaction method for the cyclic redundancy check (CRC).

Suffix:
<m> 1..4
Selects the serial data bus.

Parameters:
<CRCVersion> LEGA | V2010

Legacy: method used up to 2010
V2010: current method

*RST: V2010

BUS<m>:SENT:CRCMethod <CRCCalculation>

Selects the calculation method for the CRC checksum.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<CRCCalculation> SAEJ | TLE

SAEJ: according to the standard
TLE: according to the computing method for TLE_4998X sensors.

*RST: SAEJ

BUS<m>:SENT:PPULse <PausePulse>

Determines if a pause pulse is part of the SENT transmission sequence.

Suffix:
<m> 1..4
Selects the serial data bus.

Parameters:
<PausePulse> NPP | PP | PPFL

NPP
Transmits the SENT message without pause pulse.

PP
Transmits the message with a fixed pulse length, automatically calculated.

PPFL
Transmits the pause pulse with a user-defined frame length to obtain a transmission sequence with constant length.

*RST: PPFL

BUS<m>:SENT:PPFLength <FrameLength>

Defines a constant transmission sequence length. To select the fixed sequence length, set `BUS:SENT:PPUL PPFL`.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<FrameLength>	Range: 104 to 922 Increment: 1 *RST: 256

17.17.14.2 Trigger

Event in a trigger sequence: 1 = A-event only

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:SENT:TYPE....1668
TRIGger<m>:SENT:TTYPe....1669
TRIGger<m>:SENT:STATus....1669
TRIGger<m>:SENT:TDCN....1670
TRIGger<m>:SENT:TDMN....1670
TRIGger<m>:SENT:TDMX....1670
TRIGger<m>:SENT:STYPe....1670
TRIGger<m>:SENT:SIDType....1671
TRIGger<m>:SENT:SICN....1671
TRIGger<m>:SENT:SIMN....1671
TRIGger<m>:SENT:SIMX....1672
TRIGger<m>:SENT:SDCN....1672
TRIGger<m>:SENT:SDMN....1672
TRIGger<m>:SENT:SDMX....1672
TRIGger<m>:SENT:FORMerror....1672
TRIGger<m>:SENT:PULSeerror....1673
TRIGger<m>:SENT:PPERioderror....1673
TRIGger<m>:SENT:CRCerror....1673
TRIGger<m>:SENT:IRFLength....1674

TRIGger<m>:SENT:TYPE <Type>

Selects the trigger event for the SENT transmission type.

Parameters:

<Type>	CALI | TSEQ | SMSG | ERRC **CALI** CALIbration: triggers on the falling edge of the calibration/ synchronization pulse.

TSEQ
Transmission SEQuence: triggers either on the falling edge of the status nibble, or on the last data nibble.
To set the transmission sequence conditions, use `TRIGger<m>:SENT:TTYPe` and `TRIGger<m>:SENT:STATus`.
To set the data condition, use `TRIGger<m>:SENT:TDCN`, `BUS<m>:SENT:DNIBbles`, `TRIGger<m>:SENT:TDMN` and `TRIGger<m>:SENT:TDMX`.

SMSG
Serial Message: combination of identifier and data conditions.
To select the sequence condition, use `TRIGger<m>:SENT:STYPe`.
To select the message ID format for an enhanced serial message, use `TRIGger<m>:SENT:SIDType`.
To set the identifier condition, use `TRIGger<m>:SENT:SICN`, `TRIGger<m>:SENT:SIMN` and `TRIGger<m>:SENT:SIMX`.
To set the data condition, use `TRIGger<m>:SENT:SDCN`, `TRIGger<m>:SENT:SDMN` and `TRIGger<m>:SENT:SDMX`.

ERRC
ERRor Condition: triggers on an error event.
Define the error types with `TRIGger<m>:SENT:PULSeerror`, `TRIGger<m>:SENT:PPERioderror` or `TRIGger<m>:SENT:CRCerror`.

*RST: CALI

TRIGger<m>:SENT:TTYPe <TSFieldType>

Selects the trigger sequence type for `TRIGger<m>:SENT:TYPE TSEQ` (transmission sequence).

Parameters:

<TSFieldType> STAT | STDA

STAT
Triggers on the status nibble.

STDA
Triggers at the end of the combination of status and data nibble(s).
Define the data conditions with `TRIGger<m>:SENT:STATus`, `TRIGger<m>:SENT:TDCN`, `BUS<m>:SENT:DNIBbles`, `TRIGger<m>:SENT:TDMN` and `TRIGger<m>:SENT:TDMX`

*RST: STAT

TRIGger<m>:SENT:STATus <StatusBits>

Sets the status nibble data.

Parameters:

<StatusBits> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:TDCN <TSDataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<TSDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:SENT:TDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `TRIGger<m>:SENT:TDMN` and `TRIGger<m>:SENT:TDMX`.

*RST: EQUal

TRIGger<m>:SENT:TDMN <TSDataPattern>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:

<TSDataPattern> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:TDMX <TSDataPatternTo>

Sets the end value of an identifier range for `TRIGger:SENT:TDCN INRange or OORange`.

Parameters:

<TSDataPatternTo> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:STYPe <SMFieldType>

Selects the trigger sequence type for `TRIGger:SENT:TYPE SSMSg or ESMSg` (serial message).

Parameters:

<SMFieldType> ID | IDDT

ID
Triggers on the identifier.
To set the identifier condition for a serial message, use `TRIGger<m>:SENT:SICN`, `TRIGger<m>:SENT:SIMN` and `TRIGger<m>:SENT:SIMX`.

IDDT
Triggers at the end of the combination of identifier and data.
To set the identifier condition, use the commands shown above.
To set the data condition, use `TRIGger<m>:SENT:SDCN`, `TRIGger<m>:SENT:SDMN` and `TRIGger<m>:SENT:SDMX`.

*RST: ID

TRIGger<m>:SENT:SIDType <SMIDType>

Sets the message ID format (4 bit or 8 bit) of the enhanced serial message.

Parameters:
<SMIDType> B4 | B8

*RST: B4

TRIGger<m>:SENT:SICN <SMIDOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:
<SMIDOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:SENT:SIMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `TRIGger<m>:SENT:SIMN` and `TRIGger<m>:SENT:SIMX`.

*RST: EQUal

TRIGger<m>:SENT:SIMN <SMIDPattern>

Specifies a message identifier pattern, or sets the start value of an identifier range.

Parameters:
<SMIDPattern> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:SIMX <SMIDPatternTo>

Sets the end value of an identifier range for `TRIGger<m>:SENT:SICN INRange or OORange`.

Parameters:

<SMIDPatternTo> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:SDCN <SMDataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<SMDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:SENT:SDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `TRIGger<m>:SENT:SDMN` and `TRIGger<m>:SENT:SDMX`.

*RST: EQUal

TRIGger<m>:SENT:SDMN <SMDataPattern>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:

<SMDataPattern> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:SDMX <SMDataPatternTo>

Sets the end value of an identifier range for `TRIGger<m>:SENT:SCONdition INRange or OORange`.

Parameters:

<SMDataPatternTo> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:FORMerror <FormError>

Triggers on format errors in serial messages.

A form error occurs when at least one of the transmission sequences that form a serial message has an error.

To trigger on an error event, select the corresponding trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<FormError> ON | OFF

*RST: OFF

TRIGger<m>:SENT:PULSeerror <CalibPulseError>

Trigger son calibration pulse erors in transmission sequences.

An error occurs when

- the duration of the "Calibration/Sync" pulse (in ticks) is less than 56*(1-clock tolerance) or more than 56*(1+clock tolerance)
- the "Calibration/Sync" pulse duration of frame (n-1) varies by more than 1.5625% from the "Calibration/Sync" pulse duration of frame (n)

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<CalibPulseError> ON | OFF

*RST: OFF

TRIGger<m>:SENT:PPERioderror <PulsePeriodErr>

Triggers on pulse period errors.

An error occurs when a nibble has any of the following:

- number of ticks at low is less than 4 ticks.
- nibble value < 0 (less than 12 ticks) or > 15 (more than 27 ticks).

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<PulsePeriodErr> ON | OFF

*RST: OFF

TRIGger<m>:SENT:CRCerror <CRCError>

Triggers on CRC errors in both, the transmission sequences and serial messages.

A CRC error occurs when the CRC calculated by the receiver differs from the received value in the CRC sequence. The CRC length is 4 bits for transmission sequences and short serial messages, and 6 bit of enhanced serial messages.

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<CRCError> ON | OFF

*RST: ON

TRIGger<m>:SENT:IRFLength <IrregularFrmLen>

Triggers on frame length errors in transmission sequences when pause pulse for constant frame length is set, see `BUS<m>:SENT:PPULse PPFL`.

An error occurs when the total length of the transmission sequence (including pause pulse) does not match the frame length setting, see `BUS<m>:SENT:PPFLength` on page 1667.

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<IrregularFrmLen> ON | OFF

*RST: OFF

17.17.14.3 Decode Results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1384
- `BUS<m>:SYMBols` on page 1385

BUS<m>:SENT:FCOunt?....1675
BUS<m>:SENT:FRAMe<n>:STATus?....1675
BUS<m>:SENT:FRAMe<n>:STARt?....1675
BUS<m>:SENT:FRAMe<n>:STOP?....1675
BUS<m>:SENT:FRAMe<n>:CSValue?....1676
BUS<m>:SENT:FRAMe<n>:DATA?....1676
BUS<m>:SENT:FRAMe<n>:IDTYpe?....1676
BUS<m>:SENT:FRAMe<n>:IDValue?....1677
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:STATe?....1677
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:VALue?....1678
BUS<m>:SENT:FRAMe<n>:PAPTicks?....1678
BUS<m>:SENT:FRAMe<n>:SCOM?....1678
BUS<m>:SENT:FRAMe<n>:SDATa?....1679
BUS<m>:SENT:FRAMe<n>:SDEXport?....1679
BUS<m>:SENT:FRAMe<n>:SYMBol?....1679
BUS<m>:SENT:FRAMe<n>:SYNCduration?....1680
BUS<m>:SENT:FRAMe<n>:TYPE?....1680
BUS<m>:SENT:RDSL....1681

BUS<m>:SENT:FCOunt?

Returns the number of decoded frames of the acquisition.

Suffix:

<m>	1..4 Selects the serial data bus.

Return values:

<Count>	Total number of decoded frames. Range: 0 to 100000 Increment: 1 *RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m>	1..4 Selects the serial data bus.
<n>	* Selects the frame.

Return values:

<FrameState>	OK | SYNC | PULSe | CRC | IRFL | FORM | INSufficient OK: the frame is valid. SYNC: Synchronization error occurred. PULse: Pulse error occurred. CRC: Cyclic redundancy check failed. IRFL: Irregular frame length error occurred. FORM: Format error occurred. INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid. *RST: OK

Usage: Query only

BUS<m>:SENT:FRAMe<n>:STARt?
BUS<m>:SENT:FRAMe<n>:STOP?

Returns the start time and stop time of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<Start>, <Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SENT:FRAMe<n>:CSValue?

Returns the CRC sequence value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.
Range: 0 to 63
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:DATA?

Returns the data of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Parameters:

<Data> Comma-separated sequence of integer values (N, D1, D2,…, DN). N is the number of nibbles in the frame, and D1…DN are the values of the nibbles.

Example:

```
BUS:SENT:FRAMe4:DATA?
<-- 4,3,15,11,9
```

Usage: Query only

BUS<m>:SENT:FRAMe<n>:IDTYpe?

Returns the identifier type of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<IdentifierType>	B4 | B8 B4: standard format, 4 bit B8: extended format, 8 bit *RST: B4

Usage: Query only

BUS<m>:SENT:FRAMe<n>:IDValue?

Returns the identifier value of the selected frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<IdentifierValue>	To set the value format, use `FORMat:BPATtern`. The values below – range, increment and reset – are decimal values. Range: 0 to 255 Increment: 1 *RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:NIBBle<o>:STATe?

Returns the state of the specified nibble.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the nibble number.

Return values:

<State>	OK | UNDF UNDF: Undefined *RST: OK

Usage: Query only

BUS<m>:SENT:FRAMe<n>:NIBBle<o>:VALue?

Returns the value of the specified nibble.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the nibble number.

Return values:

<Value>	To set the value format, use FORMat:BPATtern. The values below – range, increment and reset – are decimal values. Range: 0 to 15 Increment: 1 *RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:PAPTicks?

Returns the number of the pulse pause clock ticks.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<PausePulseTicks>	Range: 12 to 768 Increment: 1 *RST: 12

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SCOM?

Returns the value of the status/communication pulse.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<StatusComm> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SDATa?

Returns the symbolic data of the frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<SymbolicData> Comma-separated list of values. The first value is the number of bytes, followed by the decoded data bytes.
To set the value format, use `FORMat:BPATtern`.

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SDEXport?

Returns the symbolic data of the frame in export format.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Translation>	String with symbolic label of the identifier.

Example: `BUS:SENT:FRAMe:SYMBol?`
Response: `Air Temperature`

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SYNCduration?

Returns the time of the synchronization pulse.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<SyncDuration>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:SENT:FRAMe<n>:TYPE?

Returns the type of SENT message.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<FrameType>	TRSQ | SMSG | EMSG Transmission sequence, short serial message or enhanced serial message. *RST: TRSQ

Usage: Query only

BUS<m>:SENT:RDSL <RessDispSel>

Selects the results to be displayed.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<RessDispSel> TRSQ | SMSG | ALL
Transmission sequence, serial messages or all.
*RST: ALL

17.17.14.4 SENT Search Settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to SENT trigger commands.

SEARch:TRIGger:SENT:CALibration....1681
SEARch:TRIGger:SENT:TRANsmission....1682
SEARch:TRIGger:SENT:SMSG....1682
SEARch:TRIGger:SENT:ERRor....1682
SEARch:TRIGger:SENT:TTYPe....1683
SEARch:TRIGger:SENT:STATus....1683
SEARch:TRIGger:SENT:TDCN....1684
SEARch:TRIGger:SENT:TDMN....1684
SEARch:TRIGger:SENT:TDMX....1684
SEARch:TRIGger:SENT:STYPe....1685
SEARch:TRIGger:SENT:SIDType....1685
SEARch:TRIGger:SENT:SICN....1685
SEARch:TRIGger:SENT:SIMN....1686
SEARch:TRIGger:SENT:SIMX....1686
SEARch:TRIGger:SENT:SDCN....1686
SEARch:TRIGger:SENT:SDMN....1687
SEARch:TRIGger:SENT:SDMX....1687
SEARch:TRIGger:SENT:PULSeerror....1687
SEARch:TRIGger:SENT:PPERioderror....1688
SEARch:TRIGger:SENT:IRFLength....1688
SEARch:TRIGger:SENT:FORMerror....1688
SEARch:TRIGger:SENT:CRCerror....1688

SEARch:TRIGger:SENT:CALibration <SearchName>,<CalSyncPulse>
SEARch:TRIGger:SENT:CALibration? <SearchName>

Enables the search for the Calibration/Synchronization pulse.

Parameters:

<CalSyncPulse> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TRANsmission <SearchName>,<TransmSequence>
SEARch:TRIGger:SENT:TRANsmission? <SearchName>

Enables the search for a transmission sequence, which is combination of status and data conditions.

To set the transmission sequence conditions, use `SEARch:TRIGger:SENT:TTYPe` and `SEARch:TRIGger:SENT:STATus`.

To set the data condition, use `SEARch:TRIGger:SENT:TDCN`, `TRIGger<m>:SENT:TDCN`, `SEARch:TRIGger:SENT:TDMN` and `SEARch:TRIGger:SENT:TDMX`.

Parameters:
<TransmSequence> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SMSG <SearchName>,<SerialMessage>
SEARch:TRIGger:SENT:SMSG? <SearchName>

Enables the search in a serial message, which is combination of identifier and data conditions.

To select the sequence condition, use `SEARch:TRIGger:SENT:STYPe`.

To set the identifier condition for the serial message, use `SEARch:TRIGger:SENT:SICN`, `SEARch:TRIGger:SENT:SIMN` and `SEARch:TRIGger:SENT:SIMX`.

To set the data condition, use `SEARch:TRIGger:SENT:SDCN`, `SEARch:TRIGger:SENT:SDMN` and `SEARch:TRIGger:SENT:SDMX`.

Parameters:
<SerialMessage> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:ERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:SENT:ERRor? <SearchName>

Enables the search for specified errors.

Define the error types with `SEARch:TRIGger:SENT:PULSeerror`, `SEARch:TRIGger:SENT:PPERioderror`, `SEARch:TRIGger:SENT:FORMerror` on page 1688 and `SEARch:TRIGger:SENT:CRCerror`.

Parameters:

<ErrorCondition> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:TTYPe <SearchName>,<TSFieldType>
SEARch:TRIGger:SENT:TTYPe? <SearchName>

Selects the SENT transmission sequence to be searched for.

To enable the search for the transmission sequence, use `SEARch:TRIGger:SENT:TRANsmission`.

Parameters:

<TSFieldType> STAT | STDA

STAT
Searches on the status nibble.

STDA
Searches for the end of the combination of status and data nibble(s).
Define the data conditions with `SEARch:TRIGger:SENT:STATus`, `SEARch:TRIGger:SENT:TDCN`, `BUS<m>:SENT:DNIBbles`, `SEARch:TRIGger:SENT:TDMN` and `SEARch:TRIGger:SENT:TDMX`.

*RST: STAT

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:STATus <SearchName>,<StatusBits>
SEARch:TRIGger:SENT:STATus? <SearchName>

Sets the status nibble data.

Parameters:

<StatusBits> Numeric or string pattern, Chapter 17.4.6, "Bit Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:TDCN <SearchName>,<TSDataOperator>
SEARch:TRIGger:SENT:TDCN? <SearchName>

Sets the operator for a specific data pattern or a data pattern range.

Parameters:
<TSDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:SENT:TDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `SEARch:TRIGger:SENT:TDMN` and `SEARch:TRIGger:SENT:TDMX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TDMN <SearchName>,<TSDataPattern>
SEARch:TRIGger:SENT:TDMN? <SearchName>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:
<TSDataPattern> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TDMX <SearchName>,<TSDataPatternTo>
SEARch:TRIGger:SENT:TDMX? <SearchName>

Sets the end value of an identifier range for `SEARch:TRIGger:SENT:DCONdition INRange or OORange`.

Parameters:
<TSDataPatternTo> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:STYPe <SearchName>,<SMFieldType>
SEARch:TRIGger:SENT:STYPe? <SearchName>

Selects the serial message sequence to be searched for.

To enable the search for one of the serial message types, use `BUS<m>:SENT:SFORmat` and enable with `SEARch:TRIGger:SENT:SMSG`.

Parameters:

<SMFieldType> ID | IDDT

ID
Searches for the identifier.
To set the identifier condition for the serial message, use `SEARch:TRIGger:SENT:SICN`, `SEARch:TRIGger:SENT:SIMN` and `SEARch:TRIGger:SENT:SIMX`.

IDDT
Searches for the combination of identifier and data.
To set the identifier condition, use the commands shown above.
To set the data condition, use `SEARch:TRIGger:SENT:SDCN`, `SEARch:TRIGger:SENT:SDMN` and `SEARch:TRIGger:SENT:SDMX`.

*RST: ID

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:SIDType <SearchName>,<SMIDType>
SEARch:TRIGger:SENT:SIDType? <SearchName>

Sets the message ID format (4 bit or 8 bit) of the enhanced serial message.

Parameters:

<SMIDType> B4 | B8

*RST: B4

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:SICN <SearchName>,<SMIDOperator>
SEARch:TRIGger:SENT:SICN? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<SMIDOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:SENT:SIMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with SEARch:TRIGger:SENT:SIMN and SEARch:TRIGger:SENT:SIMX.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SIMN <SearchName>,<SMIDPattern>
SEARch:TRIGger:SENT:SIMN? <SearchName>

Specifies a message identifier pattern, or sets the start value of an identifier range.

Parameters:
<SMIDPattern> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SIMX <SearchName>,<SMIDPatternTo>
SEARch:TRIGger:SENT:SIMX? <SearchName>

Sets the end value of an identifier range for SEARch:TRIGger:SENT:SICN INRange or OORange.

Parameters:
<SMIDPatternTo> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SDCN <SearchName>,<SMDataOperator>
SEARch:TRIGger:SENT:SDCN? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:
<SMDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with SEARch:TRIGger:SENT:SDMN.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `SEARch:TRIGger:SENT:SDMN` and `SEARch:TRIGger:SENT:SDMX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SDMN <SearchName>,<SMDataPattern>
SEARch:TRIGger:SENT:SDMN? <SearchName>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:
<SMDataPattern> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SDMX <SearchName>,<SMDataPatternTo>
SEARch:TRIGger:SENT:SDMX? <SearchName>

Sets the end value of an identifier range for `SEARch:TRIGger:SENT:SDCN INRange or OORange`.

Parameters:
<SMDataPatternTo> Numeric or string pattern, Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:PULSeerror <SearchName>,<CalibPulseError>
SEARch:TRIGger:SENT:PULSeerror? <SearchName>

Enables the search for calibration pulse errors.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:
<CalibPulseError> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:PPERioderror <SearchName>,<PulsePeriodErr>
SEARch:TRIGger:SENT:PPERioderror? <SearchName>

Enables the search for pulse period errors.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:

<PulsePeriodErr> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:IRFLength <SearchName>,<IrregularFrmLen>
SEARch:TRIGger:SENT:IRFLength? <SearchName>

Enables the search for irregular frame length errors in transmission sequences.

This error type is relevant if pause pulse mode is set to constant frame length.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:

<IrregularFrmLen> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:FORMerror <SearchName>,<FormError>
SEARch:TRIGger:SENT:FORMerror? <SearchName>

Enables the search for format errors in serial messages.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:

<FormError> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SENT:CRCerror <SearchName>,<CRCError>
SEARch:TRIGger:SENT:CRCerror? <SearchName>

Enables the search for errors in the Cyclic Redundancy Check.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:

<CRCError> ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

17.17.14.5 SENT Search Results

The search on decoded SENT data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.14.3, "Decode Results", on page 1674.

SEARch:RESult:SENT:FCOunt?....................1689
SEARch:RESult:SENT:FRAMe<m>:CSValue?....................1690
SEARch:RESult:SENT:FRAMe<m>:DATA?....................1690
SEARch:RESult:SENT:FRAMe<m>:IDTYpe?....................1690
SEARch:RESult:SENT:FRAMe<m>:IDValue?....................1690
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe?....................1691
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue?....................1691
SEARch:RESult:SENT:FRAMe<m>:PAPTicks?....................1691
SEARch:RESult:SENT:FRAMe<m>:SCOM?....................1692
SEARch:RESult:SENT:FRAMe<m>:SDATa?....................1692
SEARch:RESult:SENT:FRAMe<m>:STARt?....................1692
SEARch:RESult:SENT:FRAMe<m>:STATus?....................1693
SEARch:RESult:SENT:FRAMe<m>:STOP?....................1693
SEARch:RESult:SENT:FRAMe<m>:SYMBol?....................1693
SEARch:RESult:SENT:FRAMe<m>:SYNCduration?....................1693
SEARch:RESult:SENT:FRAMe<m>:TYPE?....................1694

SEARch:RESult:SENT:FCOunt? <SearchName>

Query parameters:

<SearchName>

Return values:

<Count> Range: 0 to 100000

Increment: 1

*RST: 0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:CSValue? <SearchName>

Returns the checksum value.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumValue> Range: 0 to 63
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:DATA? <SearchName>

Returns the data of the specified frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:IDTYpe? <SearchName>

Returns the identifier type of the selected frame.

Suffix:
<m> *
Selects the frame.

Query parameters:
<SearchName>

Return values:
<IdentifierType> B4 | B8
B4: standard format, 4 bit
B8: extended format, 8 bit
*RST: B4

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:IDValue? <SearchName>

Returns the identifier value of the selected frame.

Suffix:
<m> *
Selects the serial bus.

Query parameters:
<SearchName>

Return values:
<IdentifierValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<State> OK | UNDF
*RST: OK

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Value> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:PAPTicks? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<PausePulseTicks>	Range:	12 to 768
	Increment:	1
	*RST:	12

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SCOM? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<StatusComm>	Range:	0 to 15
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SDATa? <SearchName>

Returns the symbolic data of the specified frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SymbolicData> String that contain the symbolic data.

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | SYNC | PULSe | CRC | IRFL | FORM | INSufficient
*RST: OK

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SYNCduration? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<SyncDuration> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:TYPE? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameType> TRSQ | SMSG | EMSG
*RST: TRSQ

Usage: Query only

17.17.15 Custom: Manchester / NRZ (Option R&S RTE-K50)

- Configuration....1694
- Trigger....1710
- Filter....1716
- Decode Results....1720
- Search Settings....1727
- Search Results....1731

17.17.15.1 Configuration

In all `BUS<m>:CMSB` commands, the suffix <m> selects the serial bus.

In all `BUS<m>:CMSB:FRAMe<n>` commands, the suffix <n> selects a frame number, and the suffix <o> selects a cell number.

BUS<m>:CMSB:CODing....1695
BUS<m>:CMSB:MANChester:DATA....1696
BUS<m>:CMSB:MANChester:POLarity....1696
BUS<m>:CMSB:MANChester:THReshold:HIGH....1696
BUS<m>:CMSB:MANChester:THReshold:LOW....1697
BUS<m>:CMSB:MANChester:THReshold:PRESet....1697
BUS<m>:CMSB:MANChester:THReshold:COUPling....1697
BUS<m>:CMSB:MANChester:CPHase....1698
BUS<m>:CMSB:NRZ:CLCK....1698
BUS<m>:CMSB:NRZ:DATA....1699
BUS<m>:CMSB:NRZ:IDLPolarity....1699
BUS<m>:CMSB:NRZ:CPOLarity....1699
BUS<m>:CMSB:NRZ:CPHase....1700

BUS<m>:CMSB:NRZ:ENBLe....1700
BUS<m>:CMSB:NRZ:ENAPolarity....1701
BUS<m>:CMSB:NRZ:POLarity....1701
BUS<m>:CMSB:NRZ:THReshold:CLCK....1701
BUS<m>:CMSB:NRZ:HYSTeresis:CLCK....1701
BUS<m>:CMSB:NRZ:THReshold:DATA....1702
BUS<m>:CMSB:NRZ:HYSTeresis:DATA....1702
BUS<m>:CMSB:NRZ:THReshold:ENBLe....1702
BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe....1702
BUS<m>:CMSB:NRZ:THReshold:PRESet....1703
BUS<m>:CMSB:NRZ:THReshold:COUPling....1703
BUS<m>:CMSB:BITRate:ENABle....1704
BUS<m>:CMSB:BITRate:VALue....1704
BUS<m>:CMSB:GAPTime:ENABle....1704
BUS<m>:CMSB:GAPTime:VALue....1704
BUS<m>:CMSB:ADDFrame....1705
BUS<m>:CMSB:FCOunt?....1705
BUS<m>:CMSB:CLR....1705
BUS<m>:CMSB:FRAMe<n>:TYPE....1705
BUS<m>:CMSB:FRAMe<n>:APPend....1706
BUS<m>:CMSB:FRAMe<n>:CCOunt?....1706
BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME....1706
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount....1706
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition....1707
BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat....1707
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder....1708
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB....1708
BUS<m>:CMSB:EXRBits....1708
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN....1708
BUS<m>:CMSB:LOAD....1709
BUS<m>:CMSB:SAVE....1709

BUS<m>:CMSB:CODing <CodingStandard>

Selects the custom serial bus coding standard.

Suffix:

<m> 1..4

Parameters:

<CodingStandard> MANC | MANT | NRZ | NRZU

MANC
Manchester (normal polarity)

MANT
Manchester II (inverted polarity).
Note that some additional subtle differences between MANC and MANT require separate protocols.

NRZ
NRZ (non-return-to-zero), clocked

NRZU
NRZ (non-return-to-zero), unclocked

*RST: MANC

BUS<m>:CMSB:MANChester:DATA <SourceMANData>

Selects the source channel for the data signal. For triggering on a serial bus, analog channels "C1"–"C4" are required. Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels .

Suffix:
<m> 1..4

Parameters:
<SourceMANData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

BUS<m>:CMSB:MANChester:POLarity <PolarityData>

Selects the polarity of the custom serial bus data signal in Manchester coding standards.

Suffix:
<m> 1..4

Parameters:
<PolarityData> NORMal | INVert

NORMal
Manchester or Manchester II polarity remains unchanged

INVert
Manchester polarity is inverted and becomes Manchester II polarity, Manchester II polarity is inverted and becomes Manchester polarity

*RST: NORMal

BUS<m>:CMSB:MANChester:THReshold:HIGH <ThresholdUpper>

Sets the upper threshold for data signal digitization in Manchester coding standards.

Suffix:
<m> 1..4

Parameters:
<ThresholdUpper> Range: -25 to 25
Increment: 0.1
*RST: 5
Default unit: V

BUS<m>:CMSB:MANChester:THReshold:LOW <ThresholdLower>

Sets the lower threshold for data signal digitization in Manchester coding standards.

Suffix:

<m>	1..4

Parameters:

<ThresholdLower>
Range: -25 to 25
Increment: 0.1
*RST: -5
Default unit: V

BUS<m>:CMSB:MANChester:THReshold:PRESet <PresetValue>

Sets the Manchester thresholds to predefined or individually definable voltage levels.

Suffix:

<m>	1..4

Parameters:

<PresetValue> V05 | V2 | V5 | V7 | MANual

V05
Sets the upper threshold to +0.5 V and the lower threshold to -0.5 V

V2
Sets the upper threshold to +2.0 V and the lower threshold to -2.0 V

V5
Sets the upper threshold to +5.0 V and the lower threshold to -5.0 V

V7
Sets the upper threshold to +7.0 V and the lower threshold to -7.0 V

MANual
Allows to set individual threshold voltage levels

*RST: V5

BUS<m>:CMSB:MANChester:THReshold:COUPling <ThresholdCoupling>

Couples the upper and lower threshold values for the Manchester and Manchester II coding standards. The values are coupled to voltages with the same magnitude but opposite sign (positive for the upper threshold and negative for the lower threshold). However, if the upper threshold is set to a negative voltage or the lower threshold is set to a positive voltage, coupling is disabled, and the other voltage (the one that was not actively set) is automatically adjusted, to avoid an upper threshold below the lower one, or a lower threshold above the upper one.

Suffix:

<m> 1..4

Parameters:

<ThresholdCoupling> ON | OFF

ON
Activates coupling of the upper and lower threshold values.

OFF
Disables coupling of the upper and lower threshold values.

*RST: ON

BUS<m>:CMSB:MANChester:CPHase <ClockPhase>

Selects the phase of the custom serial bus clock signal for the "Manchester" coding standards. For details, see "Clock Phase (Manchester)" on page 739.

Suffix:

<m> 1..4

Parameters:

<ClockPhase> FEDGe | SEDGe

FEDGe
Sets the sampling edge to be on the first edge.

SEDGe
Sets the sampling edge to be on the second edge.

AUTO
Lets the decoder automatically select the method ("First Edge" or "Second Edge") for detecting the sampling edge.

*RST: SEDGe

BUS<m>:CMSB:NRZ:CLCK <SourceNRZClock>

Selects the source channel for the clock signal in the NRZ Clocked coding standard.

For triggering on the serial bus when the NRZ clocked coding standard is selected, analog or digital channel sources are required.

Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels.

Suffix:

<m> 1..4

Parameters:

<SourceNRZClock> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C2W1

BUS<m>:CMSB:NRZ:DATA <SourceNRZData>

Selects the source channel for the data signal in NRZ coding standards.

For triggering on the serial bus when the NRZ clocked coding standard is selected, analog or digital channel sources are required.

For triggering on the serial bus when the NRZ unclocked coding standard is selected, analog channel sources are required.

Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels.

Suffix:

<m> 1..4

Parameters:

<SourceNRZData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C1W1

BUS<m>:CMSB:NRZ:IDLPolarity <PolarityIdle>

Selects the idle polarity of the custom serial bus data signal (only available for the coding standard "NRZ Unclocked").

Suffix:

<m> 1..4

Parameters:

<PolarityIdle> IDLLow | IDLHigh

IDLLow
Sets the base value of the data bus to be "0". After an idle period, the data signal starts with a low-to-high transition

IDLHigh
Sets the base value of the data bus to be "1". After an idle period, the data signal starts with a high-to-low transition

*RST: IDLLow

BUS<m>:CMSB:NRZ:CPOLarity <PolarityClock>

Selects the polarity of the clock signal for the coding standard NRZ Clocked.

Suffix:

<m> 1..4

Parameters:

<PolarityClock> IDLLow | IDLHigh

IDLLow
Sets the base value of the clock to be "0", the clock signal starts with a low-to-high transition

IDLHigh
Sets the base value of the clock to be "1", the clock signal starts with a high-to-low transition.

*RST: IDLLow

BUS<m>:CMSB:NRZ:CPHase <ClockPhase>

Selects the phase of the custom serial bus clock signal for the coding standard "NRZ Clocked", depending on `BUS<m>:CMSB:NRZ:IDLPolarity` on page 1699.

Suffix:

<m> 1..4

Parameters:

<ClockPhase> FEDGe | SEDGe

FEDGe
Sets the clocking transaction to be on the first edge:
If Clock Polarity = "IDLLow", data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge
If Clock Polarity = "IDLHigh", data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge

SEDGe
Sets the clocking transaction to be on the second edge:
If Clock Polarity = "IDLLow", data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge
If Clock Polarity = "IDLHigh", data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge

*RST: FEDGe

BUS<m>:CMSB:NRZ:ENBLe <SourceNRZEnable>

Selects the input source for the custom serial bus enable signal.

If an input is chosen, signals will be only decoded while this channel is in the enabled state. This allows you to mark a time when the signal on the selected source is active and when not.

For triggering on a serial bus, analog channels "C1"–"C4" are required. Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels.

Suffix:

<m> 1..4

Parameters:

<SourceNRZEnable> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

BUS<m>:CMSB:NRZ:ENAPolarity <PolarityEnable>

Sets whether the transmitted enable signal is active when the voltage is below the threshold (`ENALow`) or higher than it (`ENAHIgh`).

Suffix:

<m> 1..4

Parameters:

<PolarityEnable> ENALow | ENAHigh

*RST: ENAHigh

BUS<m>:CMSB:NRZ:POLarity <PolarityData>

Selects the polarity of the custom serial bus data signal in NRZ coding standards.

Suffix:

<m> 1..4

Parameters:

<PolarityData> ACTLow | ACTHigh

ACTLow
Active low: the value "1" is represented by a voltage below the threshold

ACTHigh
Active high: the value "1" is represented by a voltage above the threshold

*RST: ACTHigh

BUS<m>:CMSB:NRZ:THReshold:CLCK <ThresholdClock>

Sets the threshold for the clock signal digitization in the NRZ Clocked coding standard.

Suffix:

<m> 1..4

Parameters:

<ThresholdClock> Range: -25 to 25
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:CMSB:NRZ:HYSTeresis:CLCK <HystClock>

Sets the hysteresis size on the clock channel.

Suffix:
<m> 1..4

Parameters:
<HystClock> Range: -25 to 25
Increment: 0.1
*RST: 0.1
Default unit: V

BUS<m>:CMSB:NRZ:THReshold:DATA <ThresholdData>

Sets the threshold for the data signal digitization in NRZ coding standards.

Suffix:
<m> 1..4

Parameters:
<ThresholdData> Range: -25 to 25
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:CMSB:NRZ:HYSTeresis:DATA <HystData>

Sets the hysteresis size on the data line channel.

Suffix:
<m> 1..4

Parameters:
<HystData> Range: -25 to 25
Increment: 0.1
*RST: 0.1
Default unit: V

BUS<m>:CMSB:NRZ:THReshold:ENBLe <ThresholdEnable>

Sets the threshold for the enable signal digitization in NRZ coding standards.

Suffix:
<m> 1..4

Parameters:
<ThresholdEnable> Range: -25 to 25
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe <HystEnable>

Sets the hysteresis size on the enable line channel.

Suffix:

<m>	1..4

Parameters:

<HystEnable> Range: -25 to 25
Increment: 0.1
*RST: 0.1
Default unit: V

BUS<m>:CMSB:NRZ:THReshold:PRESet <PresetValue>

Sets the NRZ thresholds to predefined or individually definable voltage levels.

Suffix:

<m>	1..4

Parameters:

<PresetValue> V05 | V2 | V5 | V7 | MANual

V05
Sets the clock and data threshold to +0.5 V (in case of NRZ Unclocked: data threshold, only)

V2
Sets the clock and data threshold to +2.0 V (in case of NRZ Unclocked: data threshold, only)

V5
Sets the clock and data threshold to +5.0 V (in case of NRZ Unclocked: data threshold, only)

V7
Sets the clock and data threshold to +7.0 V (in case of NRZ Unclocked: data threshold, only)

MANual
Allows to set individual threshold voltage levels

*RST: V5

BUS<m>:CMSB:NRZ:THReshold:COUPling <ThresholdCoupling>

Couples the clock and data threshold values for the NRZ Clocked coding standard. The values are coupled to the same number.

Suffix:

<m>	1..4

Parameters:

<ThresholdCoupling> ON | OFF

ON
Activates coupling of the NRZ clock and data threshold values.

OFF
Disables coupling of the NRZ clock and data threshold values.

*RST: ON

BUS<m>:CMSB:BITRate:ENABle <BitrateEnable>

Enables the bit rate settings (must always be enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard).

Suffix:

<m> 1..4

Parameters:

<BitrateEnable> ON | OFF

ON
Bit rate settings enabled

OFF
Bit rate settings disabled

*RST: OFF

BUS<m>:CMSB:BITRate:VALue <Bitrate>

Sets the transmission speed setting for the custom serial bus data signal.

Suffix:

<m> 1..4

Parameters:

<Bitrate> Range: 300 to 2E9
Increment: 100E3
*RST: 10E6
Default unit: bps

BUS<m>:CMSB:GAPTime:ENABle <GapTimeEnable>

Enables the gap time settings (must always be enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard).

Suffix:

<m> 1..4

Parameters:

<GapTimeEnable> ON | OFF

ON
Gap time settings enabled

OFF
Gap time settings disabled

*RST: OFF

BUS<m>:CMSB:GAPTime:VALue <MinGapTime>

Sets a minimum gap time for synchronization.

Suffix:

<m> 1..4

Parameters:

<MinGapTime>	Range: 1E-9 to 1 Increment: 1E-9 *RST: 10E-6 Default unit: s

---

BUS<m>:CMSB:ADDFrame

Creates an empty frame format description and adds it to the end of the frame description list.

Suffix:
<m> 1..4

Usage: Event

BUS<m>:CMSB:FCOunt?

Returns the number of frames.

Suffix:
<m> 1..4

Return values:
<Count>

Usage: Query only

BUS<m>:CMSB:CLR

Erases all cells and frames that have been created for a specific custom protocol.

Suffix:
<m> 1..4

Usage: Event

BUS<m>:CMSB:FRAMe<n>:TYPE <FrameType>

Enables the user to set a string to describe the frame type, typically according to the applicable protocol standard specifications. (For example, MDIO (Option R&S RTE-K55) defines the frames READ, WRITE, ADDRESS, etc.)

Suffix:
<m> 1..4

<n> *

Parameters:
<FrameType>

BUS<m>:CMSB:FRAMe<n>:APPend

Creates an empty cell description and adds it to the end of the active frame description.

Suffix:

<m>	1..4
<n>	*

Usage: Event

BUS<m>:CMSB:FRAMe<n>:CCOunt?

Returns the number of cells in the specified frame.

Suffix:

<m>	1..4
<n>	*

Return values:
<Count>

Usage: Query only

BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME <CellName>

Enables the user to set a cell name within a frame. Names do not have to be unique, they are just for user support.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:
<CellName>

BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount <BitCount>

Sets the bit count of a cell, hence its length. Based upon the lengths of the previous cells, this also defines the position of the cell start and end within a frame.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<BitCount>	Range:	1 to 64
	Increment:	1
	*RST:	1

BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition <Condition>

Sets various operators for a cell, to identify, e.g., mandatory values such as a CRC checksum or an ID, that help to identify a frame.

The implemented conditions and functionalities are the "equal" and "array" operators. For details, see "Condition" on page 747.

The numeric format of the condition needs to be set according to `BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat` on page 1707.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<Condition>

BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat <CellFormat>

Selects the numeric data format for the command `BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition` on page 1707, as well as for the result and honeycomb display.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<CellFormat> DEC | HEX | OCT | BIN

DEC
Decimal

HEX
Hexadecimal

OCT
Octal

BIN
Binary

*RST: BIN

BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder <BitOrder>

Selects in which order the bits of a cell are evaluated, as well as presented in the results table and honeycomb display.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<BitOrder> LSBF | MSBF

LSBF
Least significant bit first, evaluation starts at the LSB

MSBF
Most significant bit first, evaluation starts at the MSB

*RST: MSBF

BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB <Color>

Selects a cell's color representation in the honeycomb display.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<Color> ARGB value of the color to be used for the table entry.
ARGB=<Opacity(alpha) value><red value><green value><blue value>, in hexadecimal or decimal format.

Range: 0 to 4294967295
Increment: 1
*RST: 0

BUS<m>:CMSB:EXRBits

Exports the decoded frames as raw bits.

Suffix:

<m> 1..4

Usage: Event

BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN <Column>

Selects which cell shall be displayed in which result column of the decode table.

The decode table supports three result columns, which have to be unique for each frame type. For different frame types, though, different result columns can be defined to display unrelated information.

Suffix:

<m> 1..4

<n> *

<o> *

Parameters:

<Column> NONE | COL1 | COL2 | COL3

NONE
The result is not displayed in the decode table.

COL1
The result is displayed in column 1 of the decode table.

COL2
The result is displayed in column 2 of the decode table.

COL3
The result is displayed in column 3 of the decode table.

*RST: NONE

BUS<m>:CMSB:LOAD <FileName>

Opens an existing frame description file in xml format. The default path is `\\Public\Documents\Rohde-Schwarz\RTx\SaveXML`

Suffix:

<m> 1..4

Setting parameters:

<FileName>

Usage: Setting only

BUS<m>:CMSB:SAVE <FileName>

Saves a created frame description into an xml file ("Save As..."). The default path is `\\Public\Documents\Rohde-Schwarz\RTx\SaveXML`

Suffix:

<m> 1..4

Setting parameters:

<FileName>

Usage: Setting only

17.17.15.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:CMSB:TYPE....1710
TRIGger<m>:CMSB:PATTern....1711
TRIGger<m>:CMSB:ICONdition....1711
TRIGger<m>:CMSB:IMIN....1711
TRIGger<m>:CMSB:IMAX....1712
TRIGger<m>:CMSB:NRZ:WRDLength....1712
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle....1712
TRIGger<m>:CMSB:ADVanced:FRENable....1712
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle....1712
TRIGger<m>:CMSB:ADVanced:FIENable....1712
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT....1713
TRIGger<m>:CMSB:ADVanced:BIT....1713
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX....1713
TRIGger<m>:CMSB:ADVanced:DMAX....1713
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN....1714
TRIGger<m>:CMSB:ADVanced:DMIN....1714
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator....1714
TRIGger<m>:CMSB:ADVanced:DOPerator....1714
TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle....1714
TRIGger<m>:CMSB:ADVanced:ERENable....1714
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX....1715
TRIGger<m>:CMSB:ADVanced:IMAX....1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN....1715
TRIGger<m>:CMSB:ADVanced:IMIN....1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator....1715
TRIGger<m>:CMSB:ADVanced:IOPerator....1715

TRIGger<m>:CMSB:TYPE <Type>

Selects the trigger type for custom serial bus analysis.

Suffix:

<m>	1..3

Parameters:

<Type> STARt | PATTern | ADVanced

STARt
Triggers on the frame start, which is the end of the gap time as specified in `BUS<m>:CMSB:GAPTime:VALue` on page 1704

ADVanced
Advanced trigger with `TRIGger:CMSB:ADVanced` commands.
Only available if frames are defined.

PATTern
Triggers on a data pattern to be specified in `TRIGger<m>:CMSB:PATTern` on page 1711

*RST: STARt

TRIGger<m>:CMSB:PATTern <DataPattern>

Sets the pattern match conditions for a payload data check. The trigger is set to the first occurrence of a matching data bit pattern (which can be freely specified), starting after the minimum gap time (`BUS<m>:CMSB:GAPTime:VALue` on page 1704), and after the detected start of the data frame.

Suffix:
<m> 1..3

Parameters:
<DataPattern>

TRIGger<m>:CMSB:ICONdition <DataIdxOperator>

Sets the operator to set a specific bit index (data position).

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | GETHan | INRange

EQUal
Equal

GETHan
Greater than or equal

INRange
In range

*RST: INRange

TRIGger<m>:CMSB:IMIN <DataPosition>

Sets the bit index (data position). If `TRIGger<m>:CMSB:ICONdition` on page 1711 is set to "INRange", "IMIN" sets the start value of the bit index range.

Suffix:
<m> 1..3

Parameters:
<DataPosition> Range: 0 to 65535
Increment: 1
*RST: 0

TRIGger<m>:CMSB:IMAX <DataPositionTo>

Sets the end value of the bit index range (data position range), if `TRIGger<m>:CMSB:ICONdition` on page 1711 is set to "INRange".

Suffix:

<m>	1..3

Parameters:

<DataPositionTo>	Range: 0 to 65535 Increment: 1 *RST: 65535

TRIGger<m>:CMSB:NRZ:WRDLength <NRZWordLength>

Sets the number of bits in an NRZ Unclocked word.

Suffix:

<m>	1..3

Parameters:

<NRZWordLength>	Range: 0 to 31 Increment: 1 *RST: 8

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle <Enable>
TRIGger<m>:CMSB:ADVanced:FRENable <Frame>, <Enabler>

Enables or disables the specific frame to be triggered on.

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:

<Frame>	
<Enabler>	ON | OFF

Example:	`:TRIGger:CMSB:ADVanced:FRENable "Master" 1`
	Enables the Master frame.
	`:TRIGger:CMSB:ADVanced:FRAMe1:ENABle 1`
	Enables the 1st frame that you have defined.

Usage:	Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle <CondEnabler>
TRIGger<m>:CMSB:ADVanced:FIENable <Frame>, <Field>, <Enabler>

Enables or disables the specific field within the defined frame to be triggered on.

Suffix:

<m>	1..3 Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Enabler> ON | OFF

Example: `:TRIGger:CMSB:ADVanced:FIEN "Master", "CRC", 1`
Enables the CRC field of the Master frame.
`:TRIGger:CMSB:ADVanced:FRAMe1:FLD3:ENABle 1`
Enables the 3rd field of the 1st frame that you have defined.

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT <BitState>
TRIGger<m>:CMSB:ADVanced:BIT <Frame>, <Field>, <Bit>

Sets the bit state of a field to be triggered on that only consists of one bit.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Bit> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Do not care" (DC) = X

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX <DataMax>
TRIGger<m>:CMSB:ADVanced:DMAX <Frame>, <Field>, <Data>

Sets the end value of a data pattern range if `TRIGger<m>:CMSB:ADVanced:DOPerator` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN <DataMin>
TRIGger<m>:CMSB:ADVanced:DMIN <Frame>, <Field>, <Data>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator <DataOperator>
TRIGger<m>:CMSB:ADVanced:DOPerator <Frame>, <Field>, <Operator>

Sets the operator for the data pattern.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Operator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

ANY = OFF
The data of the required pattern is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
This condition requires one endpoint value to be set using `TRIGger<m>:CMSB:ADVanced:DMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:CMSB:ADVanced:DMAX` and `TRIGger<m>:CMSB:ADVanced:DMIN`.

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle <Enable>
TRIGger<m>:CMSB:ADVanced:ERENable <ErrorName>, <Enabler>

Enables triggering on errors. You can search for all error types in parallel.

Suffix:
<m> 1..3

Setting parameters:
<ErrorName>

<Enabler> ON | OFF

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX <IndexMax>
TRIGger<m>:CMSB:ADVanced:IMAX <Frame>, <Field>, <Data>

Sets the end value of an index range if `TRIGger<m>:CMSB:ADVanced:IMAX` is set to `INRange` or `RANGe`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN <IndexMin>
TRIGger<m>:CMSB:ADVanced:IMIN <Frame>, <Field>, <Data>

Specifies the index, or sets the start value of an index range.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>
TRIGger<m>:CMSB:ADVanced:IOPerator <Frame>, <Field>, <Operator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:
<m> Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Operator> ANY | OFF | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

ANY = OFF
The index of the required pattern is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
This condition requires one endpoint value to be set using `TRIGger<m>:CMSB:ADVanced:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:CMSB:ADVanced:IMAX` and `TRIGger<m>:CMSB:ADVanced:IMIN`.

Usage: Setting only

17.17.15.3 Filter

BUS<m>:CMSB:FILTer:BIT....1716
BUS<m>:CMSB:FILTer:DMAX....1717
BUS<m>:CMSB:FILTer:DMIN....1717
BUS<m>:CMSB:FILTer:DOPerator....1717
BUS<m>:CMSB:FILTer:ENABle....1718
BUS<m>:CMSB:FILTer:ERENable....1718
BUS<m>:CMSB:FILTer:ERRor<n>:ENABle....1718
BUS<m>:CMSB:FILTer:FIENable....1718
BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle....1719
BUS<m>:CMSB:FILTer:FRENable....1719
BUS<m>:CMSB:FILTer:IMAX....1719
BUS<m>:CMSB:FILTer:IMIN....1720
BUS<m>:CMSB:FILTer:IOPerator....1720

BUS<m>:CMSB:FILTer:BIT <Frame>, <Field>, <Bit>

Sets the bit state of a field to be filtered on that only consists of one bit.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Bit> ONE | ZERO | DC

Usage: Setting only

BUS<m>:CMSB:FILTer:DMAX <Frame>, <Field>, <Data>

Sets the end value of a data pattern range if `BUS<m>:CMSB:FILTer:DOPerator` is set to `INRange` or `OORange`.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

BUS<m>:CMSB:FILTer:DMIN <Frame>, <Field>, <Data>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

BUS<m>:CMSB:FILTer:DOPerator <Frame>, <Field>, <Operator>

Sets the operator for the data pattern.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Operator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

ANY = OFF
The data of the required pattern is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
This condition requires one endpoint value to be set using `BUS<m>:CMSB:FILTer:DMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `BUS<m>:CMSB:FILTer:DMAX` and `BUS<m>:CMSB:FILTer:DMIN`.

Usage: Setting only

BUS<m>:CMSB:FILTer:ENABle <Enable>

Enables filtering for the custom serial bus.

Suffix:
<m> 1..4

Parameters:
<Enable> ON | OFF
*RST: OFF

BUS<m>:CMSB:FILTer:ERENable <ErrorName>, <Enabler>

Enables filtering on error.

Suffix:
<m> 1..4

Setting parameters:
<ErrorName>

<Enabler> ON | OFF

Usage: Setting only

BUS<m>:CMSB:FILTer:ERRor<n>:ENABle <Enable>

Enables filtering on error. You can filter all error types in parallel.

Suffix:
<m> 1..4

<n> *

Parameters:
<Enable> ON | OFF
*RST: ON

BUS<m>:CMSB:FILTer:FIENable <Frame>, <Field>, <Enabler>

Enables or disables the specific field within the defined frame to be filtered.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Enabler> ON | OFF

Usage: Setting only

BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle <Enable>

Enables or disables the specific frame to be filtered.

Suffix:
<m> 1..4

<n> *

Parameters:
<Enable> ON | OFF
*RST: OFF

BUS<m>:CMSB:FILTer:FRENable <Frame>, <Enabler>

Enables or disables the specific frame to be filtered.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Enabler> ON | OFF

Usage: Setting only

BUS<m>:CMSB:FILTer:IMAX <Frame>, <Field>, <Data>

Sets the end value of an index range if `BUS<m>:CMSB:FILTer:IOPerator` is set to `INRange` or `RANGe`.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

BUS<m>:CMSB:FILTer:IMIN <Frame>, <Field>, <Data>

Specifies the index, or sets the start value of an index range.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

BUS<m>:CMSB:FILTer:IOPerator <Frame>, <Field>, <Operator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Operator> ANY | OFF | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

ANY = OFF
The index of the required pattern is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
This condition requires one endpoint value to be set using `BUS<m>:CMSB:FILTer:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `BUS<m>:CMSB:FILTer:IMAX` and `BUS<m>:CMSB:FILTer:IMIN`.

Usage: Setting only

17.17.15.4 Decode Results

In all `BUS<m>:CSMB:RESult<n>` commands, the suffix <m> selects the serial bus, the suffix <n> selects the result number in the decode table, and the suffix <o> selects the cell number.

As an example, with reference to Figure 12-91, Table 12-17 and Table 12-18, a set of query commands for bus #1 and result #1 is shown in the following, together with examples for outcomes of these queries:

- `:BUS1:CMSB:RCOunt?` !5

- `:BUS1:CMSB:RESult1:STATe?` !OK
- `:BUS1:CMSB:RESult1:STARt?` ! -0.0024964177
- `:BUS1:CMSB:RESult1:STOP?` ! -0.0024030384
- `:BUS1:CMSB:RESult1:TYPE?` !ff
- `:BUS1:CMSB:RESult1:CONE?` !0b11111111
- `:BUS1:CMSB:RESult1:CTWO?` !0xAA
- `:BUS1:CMSB:RESult1:CTHR?` !0xF590

- `:BUS1:CMSB:RESult1:CCOunt?` !5

- `:BUS1:CMSB:RESult1:CELL1:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL1:NAME?` !Start Delim
- `:BUS1:CMSB:RESult1:CELL1:VALue?` !101010101HL10HL0

- `:BUS1:CMSB:RESult1:CELL2:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL2:NAME?` !OP-FF
- `:BUS1:CMSB:RESult1:CELL2:VALue?` !0b11111111

- `:BUS1:CMSB:RESult1:CELL3:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL3:NAME?` !data
- `:BUS1:CMSB:RESult1:CELL3:VALue?` !0xAA

- `:BUS1:CMSB:RESult1:CELL4:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL4:NAME?` !CRC
- `:BUS1:CMSB:RESult1:CELL4:VALue?` !0xF590

- `:BUS1:CMSB:RESult1:CELL5:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL5:NAME?` !End Delim
- `:BUS1:CMSB:RESult1:CELL5:VALue?` !1HLHL101

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1383
- `BUS<m>:RESDetail` on page 1384

BUS<m>:CMSB:RCOunt?....................1722
BUS<m>:CMSB:RESult<n>:STATe?....................1722
BUS<m>:CMSB:RESult<n>:STARt?....................1723
BUS<m>:CMSB:RESult<n>:STOP?....................1723
BUS<m>:CMSB:RESult<n>:TYPE?....................1723
BUS<m>:CMSB:RESult<n>:CONE?....................1724
BUS<m>:CMSB:RESult<n>:CTWO?....................1724
BUS<m>:CMSB:RESult<n>:CTHRee?....................1724
BUS<m>:CMSB:RESult<n>:CCOunt?....................1724

BUS<m>:CMSB:RESult<n>:CELL<o>:NAME?...1725
BUS<m>:CMSB:RESult<n>:CELL<o>:STARt?...1725
BUS<m>:CMSB:RESult<n>:CELL<o>:STOP?.. 1725
BUS<m>:CMSB:RESult<n>:CELL<o>:STATe?...1726
BUS<m>:CMSB:RESult<n>:CELL<o>:VALue?...1726

BUS<m>:CMSB:RCOunt?

Returns the count number of decoded result frames in a custom serial bus waveform. Basically, this is the maximum result index <n> when querying results by using `BUS<m>:CMSB:RESult<n>:XXX`.

Suffix:

<m> 1..4

Return values:

<Count>

Usage: Query only

BUS<m>:CMSB:RESult<n>:STATe?

Returns the overall state of the frame: either OK or the relevant error condition.
R&S RTE-K50 marks each frame with a status that indicates whether the decode succeeded or not.

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
The frame was decoded normally and conforms to the frame description.

LENGth
The length error indicates that the frame ended prematurely, or an array in the frame had too few elements. The amount of bits that the software expected (based upon the user's frame description) was not found before the frame was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The frame ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

PARity
Parity bit error, indicating a transmission error (only available if a parity is configured)
*RST: OK

Usage: Query only

BUS<m>:CMSB:RESult<n>:STARt?

Returns the start time of the frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:STOP?

Returns the stop time of the frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:TYPE?

Returns the name of the selected frame (n) from the user defined frame format description, labeled according to `BUS<m>:CMSB:FRAMe<n>:TYPE` on page 1705.

Suffix:
<m> 1..4

<n> *

Return values:
<Type>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CONE?

Returns the 1st cell content as specified in the Result Column of the "Frame Format" specification table.

Suffix:

<m>	1..4
<n>	*

Return values:
<Custom1>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CTWO?

Returns the 2nd cell content as specified in the Result Column of the "Frame Format" specification table.

Suffix:

<m>	1..4
<n>	*

Return values:
<Custom2>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CTHRee?

Returns the 3rd cell content as specified in the Result Column of the "Frame Format" specification table.

Suffix:

<m>	1..4
<n>	*

Return values:
<Custom3>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CCOunt?

Returns the number of decoded cells in the specified result frame.

Suffix:

<m>	1..4
<n>	*

Return values:

<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:NAME?

Returns the name of the specified cell. Cell names are not necessarily unique.

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Name>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:STARt?

Returns the start time of the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

<o> *
Selects the cell.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:STOP?

Returns the end time of the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

<o> *
Selects the cell.

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:STATe?

Returns the state of the cell.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
No error detected

LENGth
The length error indicates that the cell ended prematurely. The amount of bits that the software expected (based upon the user's frame description) was not found before the cell was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The cell ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

PARity
Parity bit error, indicating a transmission error (only available if a parity is configured)

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:VALue?

Data content of the specified cell.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<Value>

Usage: Query only

17.17.15.5 Search Settings

SEARch:TRIGger:CMSB:ERENable....1727
SEARch:TRIGger:CMSB:ERRor<m>:ENABle....1727
SEARch:TRIGger:CMSB:FIENable....1727
SEARch:TRIGger:CMSB:FRENable....1728
SEARch:TRIGger:CMSB:BIT....1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT....1728
SEARch:TRIGger:CMSB:DMAX....1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX....1728
SEARch:TRIGger:CMSB:DMIN....1729
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN....1729
SEARch:TRIGger:CMSB:DOPerator....1729
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator....1729
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle....1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle....1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX....1730
SEARch:TRIGger:CMSB:IMAX....1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN....1731
SEARch:TRIGger:CMSB:IMIN....1731
SEARch:TRIGger:CMSB:IOPerator....1731
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator....1731

SEARch:TRIGger:CMSB:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:CMSB:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:CMSB:ERRor<m>:ENABle? <SearchName>

Defines the error type to be searched for. You can search for all error types in parallel.

Suffix:
<m> *

Parameters:
<Enable> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:FIENable <SearchName>,<Frame>, <Field>, <Enabler>

Enables or disables the specific field within the defined frame to be searched for.

Setting parameters:
<SearchName>

<Frame>

<Field>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:CMSB:FRENable <SearchName>,<Frame>, <Enabler>

Enables or disables the specific frame to be searched for.

Setting parameters:
<SearchName>

<Frame>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:CMSB:BIT <SearchName>,<Frame>, <Field>, <Bit>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT <SearchName>,<BitState>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT? <SearchName>

Sets the bit state of a field to be searched that only consists of one bit.

Suffix:
<m> *

<n> *

Parameters:
<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Do not care" (DC) = X

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> *

<n> *

Parameters:

<DataMax> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CMSB:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:

<m> *

<n> *

Parameters:

<DataMin> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:CMSB:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator <SearchName>,<DataOperator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:

<m> *

<n> *

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of end-point values to be set using SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN and SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle? <SearchName>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle <SearchName>,<CondEnabler>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific data pattern in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:
<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX <SearchName>,<IndexMax>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX? <SearchName>
SEARch:TRIGger:CMSB:IMAX <SearchName>,<Frame>, <Field>, <Data>

Sets the end value of the bit index range (data position range), if SEARch:TRIGger:CMSB:IOPerator is set to INRange.

Setting parameters:
<SearchName>

<Frame>

<Field>

<Data>

Usage: Setting only

SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN <SearchName>,<IndexMin>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN? <SearchName>
SEARch:TRIGger:CMSB:IMIN <SearchName>,<Frame>, <Field>, <Data>

Sets the bit index (data position).

Setting parameters:
<SearchName>

<Frame>

<Field>

<Data>

Usage: Setting only

SEARch:TRIGger:CMSB:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator
<SearchName>,<IndexOperator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:
<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

17.17.15.6 Search Results

SEARch:RESult:CMSB<m>:FCOunt? ... 1732
SEARch:RESult:CMSB<m>:FRAMe<n>:CCOunt? ... 1732
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:NAME? ... 1732
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STARt? ... 1733
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STATe? ... 1733
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STOP? ... 1734
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:VALue? ... 1734
SEARch:RESult:CMSB<m>:FRAMe<n>:CONE? ... 1735

SEARch:RESult:CMSB<m>:FRAMe<n>:CTHRee?...........1735
SEARch:RESult:CMSB<m>:FRAMe<n>:CTWO?...........1735
SEARch:RESult:CMSB<m>:FRAMe<n>:STARt?...........1736
SEARch:RESult:CMSB<m>:FRAMe<n>:STATe?...........1736
SEARch:RESult:CMSB<m>:FRAMe<n>:STOP?...........1737
SEARch:RESult:CMSB<m>:FRAMe<n>:TYPE?...........1737

SEARch:RESult:CMSB<m>:FCOunt? <SearchName>

Returns the count number of decoded result frames in a custom serial bus waveform. Basically, this is the maximum result index <n> when querying results by using `SEARch:RESult:CMSB<m>:FRAMe<n>:XXX`.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CCOunt? <SearchName>

Returns the count number of cell in the frames, within the search result, in a custom serial bus waveform.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:NAME? <SearchName>

Returns the name of the specified cell within the search result. Cell names are not necessarily unique.

Suffix:

<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<Name>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STARt? <SearchName>

Returns the start time of the cell within the search result.

Suffix:
<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STATe? <SearchName>

Returns the state of the cell within the search result.

Suffix:
<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
No error detected

LENGth
The length error indicates that the cell ended prematurely. The amount of bits that the software expected (based upon the user's frame description) was not found before the cell was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The cell ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

PARity
Parity bit error, indicating a transmission error (only available if a parity is configured)

*RST: OK

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STOP? <SearchName>

Returns the stop time of the cell within the search result.

Suffix:
<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:VALue? <SearchName>

Data content of the specified cell within the search result.

Suffix:
<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<Value>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CONE? <SearchName>

Returns the 1st cell content within the search result, as specified in the Result Column of the "Frame Format" specification table.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Custom1>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CTHRee? <SearchName>

Returns the 3rd cell content, within the search result as specified in the Result Column of the "Frame Format" specification table.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Custom3>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CTWO? <SearchName>

Returns the 2nd cell content within the search result, as specified in the Result Column of the "Frame Format" specification table.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Custom2>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:STARt? <SearchName>

Returns the start time of the frame within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:STATe? <SearchName>

Returns the overall state of the frame within the search result: either OK or the relevant error condition. R&S RTE-K50 marks each frame with a status that indicates whether the decode succeeded or not.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
The frame was decoded normally and conforms to the frame description.

LENGth
The length error indicates that the frame ended prematurely, or an array in the frame had too few elements. The amount of bits that the software expected (based upon the user's frame description) was not found before the frame was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The frame ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

*RST: OK

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:STOP? <SearchName>

Returns the stop time of the frame within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:TYPE? <SearchName>

Returns the name of the selected frame (n) from the user defined frame format description, labeled according to `BUS<m>:CMSB:FRAMe<n>:TYPE` on page 1705.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Type>

Usage: Query only

17.17.16 MDIO (Option R&S RTE-K55)

- Configuration....1738
- Trigger....1740
- Decode Results....1743
- Search Settings....1746
- Search Results....1749

17.17.16.1 Configuration

In all `BUS<m>:MDIO` commands, the suffix <m> selects the serial bus.

BUS<m>:MDIO:CLOCk:SOURce....1738
BUS<m>:MDIO:DATA:SOURce....1738
BUS<m>:MDIO:CLOCk:THReshold:HIGH....1739
BUS<m>:MDIO:CLOCk:THReshold:LOW....1739
BUS<m>:MDIO:DATA:THReshold:HIGH....1739
BUS<m>:MDIO:DATA:THReshold:LOW....1740
BUS<m>:MDIO:PRESet....1740
BUS<m>:MDIO:COUPling....1740

BUS<m>:MDIO:CLOCk:SOURce <SourceClock>

Selects the source for the clock line (management data clock, MDC). Permitted selections are the analog channels "C1"–"C4" and the digital channels "D0"–"D15".

Suffix:

<m> 1..4

Parameters:

<SourceClock> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines. For triggering on a serial bus, analog or digital input channels are required.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C2W1

BUS<m>:MDIO:DATA:SOURce <SourceData>

Selects the source for the data signal. Permitted selections are the analog channels "C1"–"C4" and the digital channels "D0"–"D15", but not the same as for "Clock".

Suffix:

<m> 1..4

Parameters:

<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines. For triggering on a serial bus, analog or digital input channels are required.
See Chapter 17.4.2, "Waveform Parameter", on page 1037

*RST: C1W1

BUS<m>:MDIO:CLOCk:THReshold:HIGH <ThresClkHigh>

Defines the upper threshold level for the clock signal.

Suffix:

<m> 1..4

Parameters:

<ThresClkHigh> Range: -5 to 5
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:MDIO:CLOCk:THReshold:LOW <ThresClkLow>

Defines the lower threshold level for the clock signal.

Suffix:

<m> 1..4

Parameters:

<ThresClkLow> Range: -5 to 5
Increment: 0.1
*RST: 0.8
Default unit: V

BUS<m>:MDIO:DATA:THReshold:HIGH <ThresDatHigh>

Defines the upper threshold level for the data signal.

Suffix:

<m> 1..4

Parameters:

<ThresDatHigh> Range: -5 to 5
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:MDIO:DATA:THReshold:LOW <ThresDatLow>

Defines the lower threshold level for the data signal.

Suffix:

<m>	1..4

Parameters:

<ThresDatLow>	Range: -5 to 5 Increment: 0.1 *RST: 0.8 Default unit: V

BUS<m>:MDIO:PRESet <Preset>

Selects the default threshold settings according to the Ethernet standard: 2.0 V and 0.8 V.

Suffix:

<m>	1..4

Parameters:

<Preset>	DEFault | MANual *RST: DEFault

BUS<m>:MDIO:COUPling <ThresCpl>

Overwrites the data thresholds with the clock thresholds.

Suffix:

<m>	1..4

Parameters:

<ThresCpl>	ON | OFF *RST: ON

17.17.16.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:MDIO:TYPE....1741
TRIGger<m>:MDIO:ST....1741
TRIGger<m>:MDIO:FRAMetype....1742

TRIGger<m>:MDIO:PHYS.. 1742
TRIGger<m>:MDIO:REGI.. 1742
TRIGger<m>:MDIO:DATA..1742

TRIGger<m>:MDIO:TYPE <Type>

Selects the trigger type for MDIO analysis.

Parameters:

<Type> STARt | STOP | DATA

STARt
Sets the trigger to the start of frame (SOF) field. The start of frame condition and the trigger instant is the end of the preamble. Trigger pattern: preamble (32 bits "1")

STOP
Sets the trigger to the end of frame (EOF) field. The trigger instant is after the last data bit. Trigger pattern: preamble (32 bits "1") + 32 bits "X"

DATA
Sets the trigger to the data field. For more information on the data condition, see MDIO in the Ethernet standard. The trigger instant is at the end of the frame after the last data bit, as indicated in the GUI. Trigger pattern: preamble (32 bits "1") + 2 bits "ST" (Start of Frame Code) + 2 bits "OP" (Frame Type Code, or "OpCode") + 5 bits "PHYAD/PRTAD" (Physical Layer Entity Address / Port Address) + 5 bits "REGAD/DEVAD" (Register Address / Device Address) + 2 bits "TA" (turnaround time, X bits) + 16 bits "DATA/ADDRESS"

*RST: STARt

TRIGger<m>:MDIO:ST <StartCode>

Selects the start of frame code of the frame pattern; available only in trigger type "Data".

Note that Clause 22 is coded by "01", while Clause 45 is coded by "00", thus the lower Clause number is represented by the higher parameter value.

Parameters:

<StartCode> ST00 | ST01 | ST0X

ST00
Clause 45

ST01
Clause 22

ST0X
Any permissible start pattern

*RST: ST0X

TRIGger<m>:MDIO:FRAMetype <FrameType>

Selects the Type of Frame code (or OP code, OpCode, operation code); available only in trigger type "Data".

Note that the same OpCode may have different meanings in Clause 22 and Clause 45.

Parameters:

<FrameType> OP00 | OP01 | OP10 | OP11 | OPXX

OP00
Address frame (in Clause 45, only)

OP01
Write frame (in Clause 22 or Clause 45)

OP10
Read frame (in Clause 22) or Post-Read increment address frame (in Clause 45)

OP11
Read frame (in Clause 45)

OPXX
Any frame type

*RST: OPXX

Note that the user interface shows interpretations of the numerical OpCode values corresponding to Clause 45. Clause 22 is not represented by this interpretation.

TRIGger<m>:MDIO:PHYS <PhyAddr>

Sets the physical address (in Clause 22) or port address (in Clause 45) of the frame pattern (5 bits); available only in trigger type "Data".

Parameters:

<PhyAddr>

TRIGger<m>:MDIO:REGI <RegAddr>

Sets the register address (in Clause 22) or device address (in Clause 45) of the frame pattern (5 bits); available only in trigger type "Data".

Parameters:

<RegAddr>

TRIGger<m>:MDIO:DATA <Data>

Defines the 16-bit payload data pattern (both in Clause 22 or Clause 45) or the address pattern (in Clause 45, only) to trigger for; available only in trigger type "Data".

Parameters:

<Data>

17.17.16.3 Decode Results

In all `BUS<m>:MDIO:WORD<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the word number in the decode table.

BUS<m>:MDIO:WCOunt?....1743
BUS<m>:MDIO:WORD<n>:DATA?....1743
BUS<m>:MDIO:WORD<n>:PHYS?....1743
BUS<m>:MDIO:WORD<n>:REGI?....1744
BUS<m>:MDIO:WORD<n>:ST?....1744
BUS<m>:MDIO:WORD<n>:STARt?....1744
BUS<m>:MDIO:WORD<n>:STATe?....1745
BUS<m>:MDIO:WORD<n>:STOP?....1745
BUS<m>:MDIO:WORD<n>:SYMBol?....1746
BUS<m>:MDIO:WORD<n>:TYPE?....1746

BUS<m>:MDIO:WCOunt?

Returns the word count for the selected serial bus, i.e. the number of words in the present acquisition.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MDIO:WORD<n>:DATA?

Returns the 16-bit payload data field content (in Clause 22 or Clause 45), or the 16-bit address field content (in Clause 45, only) in the present acquisition of the selected word and the selected serial bus. The most significant bit (MSB) is transmitted first.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Usage: Query only

BUS<m>:MDIO:WORD<n>:PHYS?

Returns the 5-bit address field content (PHYAD/PRTAD) in the present acquisition of the selected word and the selected serial bus.

Suffix:

<m>	1..4
<n>	*

Return values:

<PhyAd>
Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MDIO:WORD<n>:REGI?

Returns the 5-bit register or device address field content (REGAD/DEVAD) in the present acquisition of the selected word and the selected serial bus.

Suffix:

<m>	1..4
<n>	*

Return values:

<RegAd>
Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MDIO:WORD<n>:ST?

Returns the Start Code (= start of frame code) in the present acquisition of the selected word and the selected serial bus.

Suffix:

<m>	1..4
<n>	*

Return values:

<StartCode>
Range: 0 to 3
Increment: 1
*RST: 0
The parameter value "0" represents Clause 45, and "1" stands for Clause 22.
The values "2" and "3" do not correspond with any legal parameters according to the standard, but they can be searched for.

Usage: Query only

BUS<m>:MDIO:WORD<n>:STARt?

Returns the start time of the frame in the selected word of the selected serial bus.

Suffix:

<m>	1..4
<n>	*

Return values:

<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MDIO:WORD<n>:STATe?

Returns the state of the frame in the present acquisition of the selected serial bus.

Suffix:

<m>	1..4
<n>	*

Return values:

<State>
OK | UNSYN | UNSYncronized | OPCO | TA_ERROR | INComplete | SHORT | SHORt

OK
No error detected

UNSYN = UNSYncronized
UNSYncronized happens when bits are found, but they are not correlated with any synchronization sequence. We don't know what these bits are, but they are there. These bits receive a flag, but they are not decoded.

OPCO
OPcode Error

TA_ERROR
Tirnaround time error

INComplete
Incomplete Frame

SHORT = SHORt
Length Error

*RST: OK

Usage: Query only

BUS<m>:MDIO:WORD<n>:STOP?

Returns the stop time of the frame from the selected word within the search result.

Suffix:

<m>	1..4
<n>	*

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:MDIO:WORD<n>:SYMBol?

Returns a textual translation (called Register Name) of the PHY or port address label in the present acquisition of the selected word and the selected serial bus.

The translation is defined in the label list.

Suffix:

<m>	1..4
<n>	*

Return values:

<Translation>

Usage: Query only

BUS<m>:MDIO:WORD<n>:TYPE?

Returns the OpCode (= operation code or frame type) in the present acquisition of the selected word and the selected serial bus.

Suffix:

<m>	1..4
<n>	*

Return values:

<FrameType>	Range: 0 to 3 Increment: 1 *RST: 0 The parameter value "0" represents Clause 45, and "1" stands for Clause 22. The values "2" and "3" do not correspond with any legal parameters according to the standard, but they can be searched for.

Usage: Query only

17.17.16.4 Search Settings

SEARch:TRIGger:MDIO:DATA.......... 1747
SEARch:TRIGger:MDIO:FRAMetype.......... 1747
SEARch:TRIGger:MDIO:PHYS..........1748

SEARch:TRIGger:MDIO:REGI..1748
SEARch:TRIGger:MDIO:ST..1748
SEARch:TRIGger:MDIO:TYPE..1748

SEARch:TRIGger:MDIO:DATA <SearchName>,<Data>
SEARch:TRIGger:MDIO:DATA? <SearchName>

Allows to define the 16-bit payload data pattern (both in Clause 22 or Clause 45) or the address pattern (in Clause 45, only) to search for; available only in search criteria type "Data".

Parameters:
<Data>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:FRAMetype <SearchName>,<FrameType>
SEARch:TRIGger:MDIO:FRAMetype? <SearchName>

Allows to select the Type of Frame code (or OP code, OpCode, operation code); available only in search criteria type "Data".

Note that the same OpCode may have different meanings in Clause 22 and Clause 45.

Parameters:
<FrameType> OP00 | OP01 | OP10 | OP11 | OPXX

OP00
Address frame (in Clause 45, only)

OP01
Write frame (in Clause 22 or Clause 45)

OP10
Read frame (in Clause 22) or Post-Read increment address frame (in Clause 45)

OP11
Read frame (in Clause 45)

OPXX
Any frame type

*RST: OPXX

Note that the user interface shows interpretations of the numerical OpCode values corresponding to Clause 45. Clause 22 is not represented by this interpretation.
Also, note that OPXX will never be a result of decoding, but it is still an option for triggering.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:PHYS <SearchName>,<PhyAddr>
SEARch:TRIGger:MDIO:PHYS? <SearchName>

Allows to set the physical address (in Clause 22) or port address (in Clause 45) of the frame pattern (5 bits); available only in search criteria type "Data".

Parameters:
<PhyAddr>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:REGI <SearchName>,<RegAddr>
SEARch:TRIGger:MDIO:REGI? <SearchName>

Allows to set the register address (in Clause 22) or device address (in Clause 45) of the frame pattern (5 bits); available only in search criteria type "Data".

Parameters:
<RegAddr>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:ST <SearchName>,<StartCode>
SEARch:TRIGger:MDIO:ST? <SearchName>

Allows to select the start of frame code of the frame pattern; available only in search criteria type "Data".

Parameters:
<StartCode> ST00 | ST01 | ST0X

ST00
Clause 45

ST01
Clause 22

ST0X
Any permissible start pattern

*RST: ST0X

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:TYPE <SearchName>,<Type>
SEARch:TRIGger:MDIO:TYPE? <SearchName>

Selects the event type to search for.

Parameters:
<Type> STARt | STOP | DATA

STARt
Searches for the start of frame (SOF).

STARt
Searches for the end of frame (EOF).

DATA
Allows to specify a payload data pattern (both in Clause 22 or Clause 45) or an address pattern (in Clause 45, only) to search for.

*RST: STARt

Parameters for setting and query:
<SearchName>

17.17.16.5 Search Results

In all `SEARch:RESult:MDIO:WORD<m>` commands, the suffix <m> selects the packet number in the list of search results.

SEARch:RESult:MDIO:WORD<m>:DATA?.......... 1749
SEARch:RESult:MDIO:WORD<m>:PHYS?..........1749
SEARch:RESult:MDIO:WORD<m>:REGI?.......... 1750
SEARch:RESult:MDIO:WORD<m>:ST?.......... 1750
SEARch:RESult:MDIO:WORD<m>:STARt?.......... 1750
SEARch:RESult:MDIO:WORD<m>:STATe?.......... 1751
SEARch:RESult:MDIO:WORD<m>:STOP?..........1751
SEARch:RESult:MDIO:WORD<m>:SYMBol?..........1752
SEARch:RESult:MDIO:WORD<m>:TYPE?..........1752
SEARch:RESult:MDIO:WCOunt?..........1752

SEARch:RESult:MDIO:WORD<m>:DATA? <SearchName>

Returns the 16-bit payload data field content (in Clause 22 or Clause 45), or the 16-bit address field content (in Clause 45, only) from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:PHYS? <SearchName>

Returns the 5-bit address field content (PHYAD/PRTAD) from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<PhyAd> Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:REGI? <SearchName>

Returns the 5-bit register or device address field content (REGAD/DEVAD) from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<RegAd> Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:ST? <SearchName>

Returns the start of frame code from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<StartCode> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:STARt? <SearchName>

Returns the start time of the frame from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:STATe? <SearchName>

Returns the state of the frame from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State>

OK | UNSYN | UNSYncronized | OPCO | TA_ERROR | INComplete | SHORT | SHORt

OK
No error detected

UNSYN = UNSYncronized
UNSYncronized happens when bits are found, but they are not correlated with any synchronization sequence. We don't know what these bits are, but they are there. These bits receive a flag, but they are not decoded.

OPCO
OPcode error

TA_ERROR
turnaround time error

INComplete
Incomplete Frame

SHORT = SHORt
Length Error

*RST: OK

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:STOP? <SearchName>

Returns the stop time of the frame from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:SYMBol? <SearchName>

Returns a textual translation (called Register Name) of the PHY or port address label from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:TYPE? <SearchName>

Returns the frame type (= operation code or OpCode) for the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameType> Range: 0 to 3
Increment: 1
*RST: 0
Note that the user interface shows interpretations of the numerical OpCode values corresponding to Clause 45: "0" (= [bin]00) represents Address, "1" (= [bin]01" represents Write, "2" (= [bin]10) represents Post Read, "3" (= [bin]11) represents Read. Clause 22 is not represented by this interpretation.

Usage: Query only

SEARch:RESult:MDIO:WCOunt? <SearchName>

Returns the word count within the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

17.17.17 USB (Option R&S RTE-K60)

● Configuration....1753
● Trigger....1756
● Decode Results....1768
● Search Settings....1773
● Search Results....1789

17.17.17.1 Configuration

In all `BUS<m>:USB` commands, the suffix <m> selects the serial bus.

BUS<m>:USB:TECHnology....1753
BUS<m>:USB:DPLus:SOURce....1754
BUS<m>:USB:DMINus:SOURce....1754
BUS<m>:USB:DIFFerential:SOURce....1754
BUS<m>:USB:DATA:SOURce....1754
BUS<m>:USB:STRobe:SOURce....1755
BUS<m>:USB:DPLus:THReshold....1755
BUS<m>:USB:DMINus:THReshold....1755
BUS<m>:USB:DIFFerential:THReshold....1756
BUS<m>:USB:DATA:THReshold....1756
BUS<m>:USB:STRobe:THReshold....1756

BUS<m>:USB:TECHnology <ProtocolType>

Defines the USB protocol technology and transmission speed.

Suffix:
<m> 1..4

Parameters:
<ProtocolType> LOW | FULL | HIGH | HSIC

LOW
USB low speed protocol (1.5 Mbit/s)

FULL
USB full speed protocol (12 Mbit/s)

HIGH
USB high speed protocol (480 Mbit/s)

HSIC
USB high speed inter-chip (HSIC) protocol (480 Mbit/s)

*RST: LOW

BUS<m>:USB:DPLus:SOURce <SourceDplus>

Selects the source for the D+ data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:
<SourceDplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USB:DMINus:SOURce <SourceDminus>

Selects the source for the D- data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:
<SourceDminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C2W1

Usage: Asynchronous command

BUS<m>:USB:DIFFerential:SOURce <SrcDiff>

Selects the source for the differential signal in the USB high speed protocol.

Suffix:
<m> 1..4

Parameters:
<SrcDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USB:DATA:SOURce <SourceData>

Selects the source for the data signal in the USB HSIC protocol.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

Usage: Asynchronous command

BUS<m>:USB:STRobe:SOURce <SourceStrobe>

Selects the source for the strobe signal in the USB HSIC protocol.

Suffix:
<m> 1..4

Parameters:
<SourceStrobe> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C2W1

Usage: Asynchronous command

BUS<m>:USB:DPLus:THReshold <ThresholdDplus>

Defines the threshold level for the D+ data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:
<ThresholdDplus> Range: -5 to 5
Increment: 0.01
*RST: 1.55
Default unit: V

BUS<m>:USB:DMINus:THReshold <ThresholdDminus>

Defines the threshold level for the D- data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:
<ThresholdDminus> Range: -5 to 5
Increment: 0.01
*RST: 1.55
Default unit: V

BUS<m>:USB:DIFFerential:THReshold <ThresDiff>

Defines the threshold level for the differential signal in the USB high speed protocol.

Suffix:

<m>	1..4

Parameters:

<ThresDiff>	Range: -2 to 2 Increment: 0.01 *RST: 0 Default unit: V

BUS<m>:USB:DATA:THReshold <ThresholdData>

Defines the threshold level for the data signal in the USB HSIC protocol.

Suffix:

<m>	1..4

Parameters:

<ThresholdData>	Range: -2 to 2 Increment: 0.01 *RST: 0.65 Default unit: V

BUS<m>:USB:STRobe:THReshold <ThresholdStrobe>

Defines the threshold level for the strobe signal in the USB HSIC protocol.

Suffix:

<m>	1..4

Parameters:

<ThresholdStrobe>	Range: -2 to 2 Increment: 0.01 *RST: 0.65 Default unit: V

17.17.17.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:USB:TYPE....1757
TRIGger<m>:USB:ACONdition....1759
TRIGger<m>:USB:AMIN....1759
TRIGger<m>:USB:AMAX....1759
TRIGger<m>:USB:DATA....1759
TRIGger<m>:USB:DCONdition....1760
TRIGger<m>:USB:DPOPerator....1760
TRIGger<m>:USB:DPOSition....1760
TRIGger<m>:USB:ECONdition....1760
TRIGger<m>:USB:EMIN....1761
TRIGger<m>:USB:EMAX....1761
TRIGger<m>:USB:ERRC....1761
TRIGger<m>:USB:FCONdition....1762
TRIGger<m>:USB:FMIN....1762
TRIGger<m>:USB:FMAX....1762
TRIGger<m>:USB:HAND....1762
TRIGger<m>:USB:PATT....1763
TRIGger<m>:USB:PCONdition....1763
TRIGger<m>:USB:PMIN....1763
TRIGger<m>:USB:PMAX....1764
TRIGger<m>:USB:SCONdition....1764
TRIGger<m>:USB:SMIN....1764
TRIGger<m>:USB:SMAX....1764
TRIGger<m>:USB:SPEC....1764
TRIGger<m>:USB:STCO....1765
TRIGger<m>:USB:TCONdition....1765
TRIGger<m>:USB:TMIN....1766
TRIGger<m>:USB:TMAX....1766
TRIGger<m>:USB:TOKen....1766
TRIGger<m>:USB:WADD....1766
TRIGger<m>:USB:WEND....1766
TRIGger<m>:USB:WETCheck....1767
TRIGger<m>:USB:WFRN....1767
TRIGger<m>:USB:WPAY....1767
TRIGger<m>:USB:WPID....1767
TRIGger<m>:USB:WPOR....1767
TRIGger<m>:USB:WSEU....1767
TRIGger<m>:USB:WSTC....1768

TRIGger<m>:USB:TYPE <Type>

Selects the trigger type for USB analysis. The available trigger types depend on the activated USB protocol type.

Parameters:

<Type> SOP | EOP | RST | SUSPend | RESume | TOKen | DATA | HANDshake | SPECial | ERRCond

SOP
Sets the trigger to the SOP (start of packet) field. The start of packet condition is the end of the SYNC field. The trigger instant is the end of the SOP field.

EOP
Sets the trigger to the EOP (end of packet) field. Not available for USB High Speed and USB HSIC protocol types. The trigger instant is the beginning of the EOP field.

RST
Sets the trigger to the Reset field. Not available for USB High Speed and USB HSIC protocol types. For more information on the reset condition, see the USB standard. The trigger instant is the end of the 10 ms period after the SE0 field.

SUSPend
Sets the trigger to the Suspend field. Not available for USB High Speed and USB HSIC protocol types. For more information on the suspend condition, see the USB standard. The trigger instant will be declared after the defined 3 ms timeout.

RESume
Sets the trigger to the Resume field. Not available for USB High Speed and USB HSIC protocol types. For more information on the resume condition, see the USB standard. The trigger instant will be declared after the defined 20 ms timeout.

TOKen
Sets the trigger to one out of four different token trigger types: OUT, IN, SOF, or SETUP.

DATA
Sets the trigger to one out of four different data trigger types: DATA0, DATA1, DATA2, or MDATA.

HANDshake
Sets the trigger to one out of four different handshake trigger types: ACK, NAK, STALL, or NYET.

SPECial
Sets the trigger to one out of four different Special PID trigger types: PREamble, ERR, SPLIT, or PING.

ERRCond
Sets the trigger to one out of seven different error condition trigger types: PID error, CRC5 error, CRC16 error, Bitstuffing error, Unexpected PID error, SE1 error, or Glitching error.

*RST: SOP

TRIGger<m>:USB:ACONdition <OperatorAddress>

Sets the operator to set a specific address or an address range. The address values are set with `TRIGger<m>:USB:AMIN` and `TRIGger<m>:USB:AMAX`.

Parameters:

<OperatorAddress> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one address value to be set using `TRIGger<m>:USB:AMIN`.

INRange | OORange
In range, out of range. These conditions require an address range to be set using `TRIGger<m>:USB:AMIN` and `TRIGger<m>:USB:AMAX`.

*RST: EQUal

TRIGger<m>:USB:AMIN <Address>

Specifies the address, or sets the the start value of an address range. The string parameter does not accept the bit value X (don't care).

Parameters:

<Address>

TRIGger<m>:USB:AMAX <AddressTo>

Sets the the end value of an address range if `TRIGger<m>:USB:ACONdition` is set to `INRange` or `OORange`. The string parameter does not accept the bit value X (don't care).

Parameters:

<AddressTo>

TRIGger<m>:USB:DATA <USBDataType>

Sets the trigger to one out of four different payload data types: DATA0, DATA1, DATA2, or MDATA.

Parameters:

<USBDataType> D0 | D1 | D2 | MD

D0
Sets the trigger to the DATA0 field (even PID).

D1
Sets the trigger to the DATA1 field (odd PID).

D2
Sets the trigger to the DATA2 field (data packet PID for high-speed, high bandwidth isochronous transaction in a microframe).

MD
Sets the trigger to the MDATA field (high-speed data packet PID for split and high bandwidth isochronous transactions).

*RST: D0

TRIGger<m>:USB:DCONdition <OperatorData>

Sets the operator (equal or unequal) to set a specific payload data pattern.

Parameters:

<OperatorData> EQUal | NEQual

*RST: EQUal

TRIGger<m>:USB:DPOPerator <DataPosOperator>

Sets the operator (any or equal) for the payload data index position.

Parameters:

<DataPosOperator> ANY | OFF | EQUal

ANY = OFF
The position of the bit pattern within the payload data is not relevant.

EQUal
Sets the operator for specifying a special start position for the bit pattern within the payload data.

*RST: ANY

TRIGger<m>:USB:DPOSition <DataPosition>

Specifies the position within a payload data packet, in which a special data pattern is to be searched.

Parameters:

<DataPosition> Range: 0 to 1024
Increment: 1
*RST: 0

TRIGger<m>:USB:ECONdition <EndpointOptor>

Sets the operator to set a specific endpoint or an endpoint range. The endpoint values are set with `TRIGger<m>:USB:EMIN` and `TRIGger<m>:USB:EMAX`.

Parameters:

<EndpointOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using TRIGger<m>:USB:EMIN.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using TRIGger<m>:USB:EMIN and TRIGger<m>:USB:EMAX.

*RST: EQUal

TRIGger<m>:USB:EMIN <Endp>

Specifies the endpoint, or sets the the start value of an endpoint range.

Parameters:
<Endp>

TRIGger<m>:USB:EMAX <EndpTo>

Sets the the end value of an endpoint range if TRIGger<m>:USB:ECONdition is set to INRange or OORange.

Parameters:
<EndpTo>

TRIGger<m>:USB:ERRC <ErrorType>

Sets the trigger to an error condition type.

Parameters:
<ErrorType> ANY | PIDerror | CRC5error | CRC16error | BTST | UNEXpid | SE1error | GLITcherr

ANY
Triggers on any of the errors listed below.

PIDerror
Triggers on any packet identifier error.

CRC5error
Triggers on any CRC5 error event.

CRC16error
Triggers on any CRC16 error event.

BTST
Triggers on any bitstuffing error event (erroneous or missing bit stuffing sequence, see USB standard).

UNEXpid
Triggers on any unexpected PID error (illegal PID, that is not allowed in USB low speed and USB full speed protocols, especially PID's announcing packets such as SPLIT, DATA2, MDATA, or other noncompliant packets).

SE1error
Triggers on the illegal bus state Single Ended 1 (SE1 = both lines high).

GLITcherr
Triggers on any glitching error (illegal bit period, see USB standard for the definition of glitching).

*RST: ANY

TRIGger<m>:USB:FCONdition <FrameNoOperator>

Sets the operator to set a specific frame number or a frame number range. The frame number values are set with `TRIGger<m>:USB:FMIN` and `TRIGger<m>:USB:FMAX`.

Parameters:

<FrameNoOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one frame number value to be set using `TRIGger<m>:USB:FMIN`.

INRange | OORange
In range, out of range. These conditions require a range of frame number values to be set using `TRIGger<m>:USB:FMIN` and `TRIGger<m>:USB:FMAX`.

*RST: EQUal

TRIGger<m>:USB:FMIN <FrameNumber>

Specifies the frame number, or sets the the start value of a frame number range.

Parameters:

<FrameNumber>

TRIGger<m>:USB:FMAX <FrameNumberTo>

Sets the the end value of a frame number range if `TRIGger<m>:USB:FCONdition` is set to `INRange` or `OORange`.

Parameters:

<FrameNumberTo>

TRIGger<m>:USB:HAND <HandshakeType>

Sets the trigger to one out of four different handshake types: ACK, NAK, STALl, or NYET.

Parameters:

<HandshakeType> ACK | NAK | STALl | NYET

ACK
Sets the trigger to the ACK field (acknowledgment of error-free data packet).

NAK
Sets the trigger to the NAK field (non-acknowledgment, no successful data transmission).

STALl
Sets the trigger to the STALL field (endpoint is halted or a control pipe request is not supported).

NYET
Sets the trigger to the NYET field (no response yet from receiver).

*RST: ACK

TRIGger<m>:USB:PATT <PayloadMuster>

Specifies the payload data pattern that is to be searched.

Parameters:
<PayloadMuster>

TRIGger<m>:USB:PCONdition <OperatorPort>

Sets the operator to set a specific port number or a port number range. The port number values are set with TRIGger<m>:USB:PMIN and TRIGger<m>:USB:PMAX.

Parameters:
<OperatorPort> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one port number to be set using TRIGger<m>:USB:PMIN.

INRange | OORange
In range, out of range. These conditions require a port number range to be set using TRIGger<m>:USB:PMIN and TRIGger<m>:USB:PMAX.

*RST: EQUal

TRIGger<m>:USB:PMIN <Port>

Specifies the port number, or sets the the start value of a port number range.

Parameters:
<Port>

TRIGger<m>:USB:PMAX <PortTo>

Sets the the end value of a port number range if `TRIGger<m>:USB:PCONdition` is set to `INRange` or `OORange`.

Parameters:
<PortTo>

TRIGger<m>:USB:SCONdition <OperatorSEU>

Sets the operator to set a specific SEU or an SEU range. The SEU values are set with `TRIGger<m>:USB:SMIN` and `TRIGger<m>:USB:SMAX`.

For SEU, see "SEU check" on page 801.

Parameters:

<OperatorSEU> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one SEU value to be set using `TRIGger<m>:USB:SMIN`.

INRange | OORange
In range, out of range. These conditions require a range of SEU values to be set using `TRIGger<m>:USB:SMIN` and `TRIGger<m>:USB:SMAX`.

*RST: EQUal

TRIGger<m>:USB:SMIN <SEU>

Specifies the SEU, or sets the the start value of an SEU range.

Parameters:
<SEU>

TRIGger<m>:USB:SMAX <SEUTo>

Sets the the end value of an SEU range if `TRIGger<m>:USB:SCONdition` is set to `INRange` or `OORange`.

Parameters:
<SEUTo>

TRIGger<m>:USB:SPEC <USBSpecialType>

Sets the trigger to one out of four different Special PID types: PREamble, ERR, SPLit, or PING.

Parameters:

<USBSpecialType> PREamble | ERR | SPLit | PING

PREamble
Sets the trigger to the PREamble PID

ERR
Sets the trigger to the ERRor PID

SPLit
Sets the trigger to the SPLIT PID (in USB high speed transactions)

PING
Sets the trigger to the PING PID (in USB high speed transactions, flow control probe for a bulk/control endpoint)

*RST: PREamble

TRIGger<m>:USB:STCO <SC>

Sets the trigger to a specific start-split or complete-split transaction endpoint.

Parameters:

<SC> ONE | ZERO | DC

ONE
SC = 1 represents a complete-split (CSPLIT) transaction.

ZERO
SC = 0 represents a start-split (SSPLIT) transaction.

DC
SC = X represents "don't care" (DC)

*RST: DC

TRIGger<m>:USB:TCONdition <OperatorET>

Sets the operator to set a specific endpoint type (ET) or an ET range. The ET values are set with `TRIGger<m>:USB:TMIN` and `TRIGger<m>:USB:TMAX`.

Parameters:

<OperatorET> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The position of the endpoint type is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one ET value to be set using `TRIGger<m>:USB:TMIN`.

INRange | OORange
In range, out of range. These conditions require a range of ET values to be set using `TRIGger<m>:USB:SMIN` and `TRIGger<m>:USB:SMAX`.

*RST: EQUal

TRIGger<m>:USB:TMIN <ET>

Specifies the endpoint type, or sets the the start value of an endpoint type range.

Parameters:
<ET>

TRIGger<m>:USB:TMAX <ETTo>

Sets the the end value of an endpoint type range if `TRIGger<m>:USB:TCONdition` is set to `INRange` or `OORange`.

Parameters:
<ETTo>

TRIGger<m>:USB:TOKen <USBTokenType>

Sets the trigger to one out of four different token types: OUT, IN, SOF, or SETup.

Parameters:

<USBTokenType> OUT | IN | SOF | SETup

OUT
Sets the trigger to the OUT token (OUT packet from host to device).

IN
Sets the trigger to the IN token (IN packet from device to host).

SOF
Sets the trigger to the SOF token (start of frame marker and frame number).

SETup
Sets the trigger to the SETUP token (address and endpoint number in OUT transaction for setup to a control pipe).

*RST: OUT

TRIGger<m>:USB:WADD <AddressCheck>

Defines, whether the address check shall be activated or not.

Parameters:

<AddressCheck> ON | OFF

*RST: OFF

TRIGger<m>:USB:WEND <EndpointCheck>

Defines, whether the endpoint check shall be activated or not.

Parameters:

<EndpointCheck> ON | OFF

*RST: OFF

TRIGger<m>:USB:WETCheck <WithETCheck>

Defines, whether the Endpoint Type (ET) check shall be activated or not.

Parameters:

<WithETCheck>	ON | OFF	
	*RST:	OFF

TRIGger<m>:USB:WFRN <FrameNoChk>

Defines, whether the frame number check shall be activated or not.

Parameters:

<FrameNoChk>	ON | OFF	
	*RST:	OFF

TRIGger<m>:USB:WPAY <PayloadCheck>

Defines, whether the payload data check shall be activated or not.

Parameters:

<PayloadCheck>	ON | OFF	
	*RST:	OFF

TRIGger<m>:USB:WPID <WithPIDCheck>

Defines, whether the packet ID error check shall be activated or not.

Parameters:

<WithPIDCheck>	ON | OFF	
	*RST:	OFF

TRIGger<m>:USB:WPOR <WithPortCheck>

Defines, whether the port check shall be activated or not.

Parameters:

<WithPortCheck>	ON | OFF	
	*RST:	OFF

TRIGger<m>:USB:WSEU <WithSEUCheck>

Defines, whether the SEU check shall be activated or not.

For SEU, see "SEU check" on page 801.

Parameters:

<WithSEUCheck>	ON | OFF	
	*RST:	OFF

TRIGger<m>:USB:WSTC <WithSCCheck>

Defines, whether the Start / Complete (SC) check shall be activated or not.

Parameters:

<WithSCCheck> ON | OFF

*RST: OFF

17.17.17.3 Decode Results

In all `BUS<m>:USB:PACKet<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the packet number in the decode table.

As an example, with reference to Figure 12-109 (packet #19) in Chapter 12.16.4, "USB 2.0 Decode Results", on page 803, the status of the Token IN packet can be queried in the following way:

▶ `BUS:USB:PACKet19:STAT?`

The result of this remote command query should be "OK".

BUS<m>:USB:PACKet<n>:PID?....1768
BUS<m>:USB:PACKet<n>:ADDRess?....1769
BUS<m>:USB:PACKet<n>:CRC?....1770
BUS<m>:USB:PACKet<n>:DATA?....1770
BUS<m>:USB:PACKet<n>:ENDPoint?....1770
BUS<m>:USB:PACKet<n>:ET?....1771
BUS<m>:USB:PACKet<n>:FRAMe?....1771
BUS<m>:USB:PACKet<n>:PORT?....1771
BUS<m>:USB:PACKet<n>:SC?....1772
BUS<m>:USB:PACKet<n>:SEU?....1772
BUS<m>:USB:PACKet<n>:STARt?....1772
BUS<m>:USB:PACKet<n>:STATus?....1773
BUS<m>:USB:PACKet<n>:STOP?....1773
BUS<m>:USB:PCOunt?....1773

BUS<m>:USB:PACKet<n>:PID?

Returns the packet PID for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<PID> RES | OUT | ACK | DATA0 | DATa0 | PING | SOF | NYET | DATA2 | DATa2 | SPLIT | SPLit | IN | NAK | DATA1 | DATa1 | PRE | SETUP | SETup | STALL | STALl | MDATA | MDATa | UNK

RES

RES = Reserved

OUT
OUT Token PID

IN
IN Token PID

SOF
Start Of Frame PID

SETUP = SETup
SETUP PID

DATA0 = DATa0
DATA0 PID, even PID

DATA1 = DATa1
DATA1 PID, odd PID

DATA2 = DATa2
DATA2 PID (only valid in USB high speed and USB HSIC protocols)

MDATA = MDATa
MDATA PID (only valid in USB high speed and USB HSIC protocols)

ACK
ACKnowledgment PID

NAK
Non-AcKnowledgment PID

STALL = STALl
STALL PID

NYET
Not ready YET (only valid in USB high speed and USB HSIC protocols)

PRE
PREamble PID (only valid in USB high speed and USB HSIC protocols)

SPLIT = SPLit
SPLIT PID (only valid in USB high speed and USB HSIC protocols)

PING
PING PID (only valid in USB high speed and USB HSIC protocols)

UNK
UNK = Unknown PID

*RST: RES

Usage: Query only

BUS<m>:USB:PACKet<n>:ADDRess?

Returns the packet address for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<Addr> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:CRC?

Returns the packet CRC (Cyclic Redundancy Code) for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<CRC> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:DATA?

Returns the payload data from the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Parameters:

<Data> Comma-separated sequence of integer values (N, D1, D2,…, DN). N is the number of bytes in the packet. and D1…DN are the values of the bytes.

Example:

```
BUS:USB:PACKet4:DATA?
<-- 6,18,52,86,120,154,188
```

Usage: Query only

BUS<m>:USB:PACKet<n>:ENDPoint?

Returns the endpoint for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<Endp> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:ET?

Returns the endpoint type (ET) for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<ET> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:FRAMe?

Returns the frame number for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameNo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:PORT?

Returns the port number for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<Port> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:SC?

Returns the Start- / Complete-split transaction (SSPLIT / CSPLIT) flag bits for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<SC>

Usage: Query only

BUS<m>:USB:PACKet<n>:SEU?

Returns the SEU values for the selected serial bus and packet number.

For SEU, see "SEU check" on page 801.

Suffix:

<m> 1..4

<n> *

Return values:

<SEU>
Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:STARt?

Returns the start position of the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USB:PACKet<n>:STATus?

Returns the status of the packet with the selected packet number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	*

Return values:

<State>	OK | PID | CRC | BTST | GLITCH | GLITch | BYTE *RST: OK

Usage: Query only

BUS<m>:USB:PACKet<n>:STOP?

Returns the stop time of the packet with the selected packet number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	*

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:USB:PCOunt?

Returns the packet count for the selected serial bus, i.e. the number of packets in the present acquisition.

Suffix:

<m>	1..4

Return values:

<Count>	Range: 0 to 100000 Increment: 1 *RST: 0

Usage: Query only

17.17.17.4 Search Settings

SEARch:TRIGger:USB:ACONdition....1775
SEARch:TRIGger:USB:AMIN....1775
SEARch:TRIGger:USB:AMAX....1775

SEARch:TRIGger:USB:BITSterror 1775
SEARch:TRIGger:USB:CRC16error 1776
SEARch:TRIGger:USB:CRC5error 1776
SEARch:TRIGger:USB:DATA 1776
SEARch:TRIGger:USB:DCONdition 1777
SEARch:TRIGger:USB:DPOPerator 1777
SEARch:TRIGger:USB:DPOSition 1777
SEARch:TRIGger:USB:ECONdition 1778
SEARch:TRIGger:USB:EMIN 1778
SEARch:TRIGger:USB:EMAX 1778
SEARch:TRIGger:USB:FCONdition 1779
SEARch:TRIGger:USB:FMIN 1779
SEARch:TRIGger:USB:FMAX 1779
SEARch:TRIGger:USB:GLITcherror 1779
SEARch:TRIGger:USB:HAND 1780
SEARch:TRIGger:USB:PATT 1780
SEARch:TRIGger:USB:PCONdition 1780
SEARch:TRIGger:USB:PMIN 1781
SEARch:TRIGger:USB:PMAX 1781
SEARch:TRIGger:USB:PIDerror 1781
SEARch:TRIGger:USB:SCONdition 1781
SEARch:TRIGger:USB:SMIN 1782
SEARch:TRIGger:USB:SMAX 1782
SEARch:TRIGger:USB:SDATa 1782
SEARch:TRIGger:USB:SERRor 1783
SEARch:TRIGger:USB:SHANdshake 1783
SEARch:TRIGger:USB:SSOP 1783
SEARch:TRIGger:USB:SSPE 1783
SEARch:TRIGger:USB:SPEC 1784
SEARch:TRIGger:USB:STCO 1784
SEARch:TRIGger:USB:STOKen 1785
SEARch:TRIGger:USB:TCONdition 1785
SEARch:TRIGger:USB:TMIN 1785
SEARch:TRIGger:USB:TMAX 1786
SEARch:TRIGger:USB:TOKen 1786
SEARch:TRIGger:USB:WADD 1786
SEARch:TRIGger:USB:WEND 1787
SEARch:TRIGger:USB:WETCheck 1787
SEARch:TRIGger:USB:WFRN 1787
SEARch:TRIGger:USB:WPAY 1787
SEARch:TRIGger:USB:WPID 1788
SEARch:TRIGger:USB:WPOR 1788
SEARch:TRIGger:USB:WSEU 1788
SEARch:TRIGger:USB:WSTC 1788

SEARch:TRIGger:USB:ACONdition <SearchName>,<OperatorAddress>
SEARch:TRIGger:USB:ACONdition? <SearchName>

Sets the operator to set a specific address or an address range.

Parameters:
<OperatorAddress> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one address value to be set using `SEARch:TRIGger:USB:AMIN`.

INRange | OORange
In range, out of range. These conditions require an address range to be set using `SEARch:TRIGger:USB:AMIN` and `SEARch:TRIGger:USB:AMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:AMIN <SearchName>,<Address>
SEARch:TRIGger:USB:AMIN? <SearchName>

Specifies an address, or sets the start value of an address range. The string parameter does not accept the bit value X (don't care).

Parameters:
<Address>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:AMAX <SearchName>,<AddressTo>
SEARch:TRIGger:USB:AMAX? <SearchName>

Sets the the end value of an address range if `TRIGger<m>:USB:ACONdition` is set to `INRange` or `OORange`. The string parameter does not accept the bit value X (don't care).

Parameters:
<AddressTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:BITSterror <SearchName>,<BitstuffError>
SEARch:TRIGger:USB:BITSterror? <SearchName>

Defines, whether a search for any bitstuffing error shall be activated or not.

Parameters:

<BitstuffError> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:CRC16error <SearchName>,<CRC16Error>
SEARch:TRIGger:USB:CRC16error? <SearchName>

Defines, whether a search for any CRC16 error shall be activated or not.

Parameters:

<CRC16Error> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:CRC5error <SearchName>,<CRC5Error>
SEARch:TRIGger:USB:CRC5error? <SearchName>

Defines, whether a search for any CRC5 error shall be activated or not.

Parameters:

<CRC5Error> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:DATA <SearchName>,<USBDataType>
SEARch:TRIGger:USB:DATA? <SearchName>

Defines, which data packet type is searched for: "DATA0", "DATA1", "DATA2", or "MDATA", as well as "Any" data packet.

Parameters:

<USBDataType> ANY | D0 | D1 | D2 | MD

ANY
Searches for any of the data packet types listed below

D0
Searches for a DATA0 packet (even PID)

D1
Searches for a DATA1 packet (odd PID)

D2
Searches for a DATA2 packet (high-speed data packet for high bandwidth isochronous transaction in a microframe)

MD
Searches for an MDATA packet (high-speed data packet for split and high bandwidth isochronous transactions)

*RST: D0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DCONdition <SearchName>,<OperatorData>
SEARch:TRIGger:USB:DCONdition? <SearchName>

Sets the operator (equal or unequal) to set a specific payload data pattern.

Parameters:
<OperatorData> EQUal | NEQual

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:USB:DPOPerator? <SearchName>

Sets the operator (any or equal) for the payload data index position.

Parameters:
<DataPosOperator> ANY | OFF | EQUal

ANY = OFF
The position of payload data is not relevant for the search condition.

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:USB:DPOSition? <SearchName>

Specifies the position within a payload data packet, in which a special data pattern is to be searched.

Parameters:
<DataPosition> Range: 0 to 1024
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:ECONdition <SearchName>,<EndpointOptor>
SEARch:TRIGger:USB:ECONdition? <SearchName>

Sets the operator to set a specific endpoint or an endpoint range. The endpoint values are set with `SEARch:TRIGger:USB:EMIN` and `SEARch:TRIGger:USB:EMAX`.

Parameters:

<EndpointOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:USB:EMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:USB:EMIN` and `SEARch:TRIGger:USB:EMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:EMIN <SearchName>,<Endp>
SEARch:TRIGger:USB:EMIN? <SearchName>

Specifies an endpoint, or sets the start value of an endpoint range.

Parameters:
<Endp>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:EMAX <SearchName>,<EndpTo>
SEARch:TRIGger:USB:EMAX? <SearchName>

Sets the the end value of an endpoint range if `TRIGger<m>:USB:ECONdition` is set to `INRange` or `OORange`.

Parameters:
<EndpTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:FCONdition <SearchName>,<FrameNoOperator>
SEARch:TRIGger:USB:FCONdition? <SearchName>

Sets the operator to set a specific frame number or a frame number range. The frame number values are set with SEARch:TRIGger:USB:FMIN and SEARch:TRIGger:USB:FMAX.

Parameters:

<FrameNoOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one frame number value to be set using SEARch:TRIGger:USB:FMIN.

INRange | OORange
In range, out of range. These conditions require a range of frame number values to be set using SEARch:TRIGger:USB:FMIN and SEARch:TRIGger:USB:FMAX.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:FMIN <SearchName>,<FrameNumber>
SEARch:TRIGger:USB:FMIN? <SearchName>

Specifies a frame number, or sets the start value of a frame number range.

Parameters:
<FrameNumber>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:FMAX <SearchName>,<FrameNumberTo>
SEARch:TRIGger:USB:FMAX? <SearchName>

Sets the the end value of a frame number range if TRIGger<m>:USB:FCONdition is set to INRange or OORange.

Parameters:
<FrameNumberTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:GLITcherror <SearchName>,<GlitchError>
SEARch:TRIGger:USB:GLITcherror? <SearchName>

Defines, whether a search for any glitch error shall be activated or not.

Parameters:

<GlitchError> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:HAND <SearchName>,<HandshakeType>
SEARch:TRIGger:USB:HAND? <SearchName>

Defines, which handshake type is searched for.

Parameters:

<HandshakeType> ANY | ACK | NAK | STALl | NYET

*RST: ACK

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:PATT <SearchName>,<PayloadMuster>
SEARch:TRIGger:USB:PATT? <SearchName>

Defines the payload data pattern to search for.

Parameters:

<PayloadMuster>

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:PCONdition <SearchName>,<OperatorPort>
SEARch:TRIGger:USB:PCONdition? <SearchName>

Sets the operator to set a specific port number or a port number range. The port number values are set with `SEARch:TRIGger:USB:PMIN` and `SEARch:TRIGger:USB:PMAX`.

Parameters:

<OperatorPort> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one port number to be set using `SEARch:TRIGger:USB:PMIN`.

INRange | OORange
In range, out of range. These conditions require a port number range to be set using `SEARch:TRIGger:USB:PMIN` and `SEARch:TRIGger:USB:PMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PMIN <SearchName>,<Port>
SEARch:TRIGger:USB:PMIN? <SearchName>

Specifies a port number, or sets the start value of a port number range.

Parameters:
<Port>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PMAX <SearchName>,<PortTo>
SEARch:TRIGger:USB:PMAX? <SearchName>

Sets the the end value of a port number range if `TRIGger<m>:USB:PCONdition` is set to `INRange` or `OORange`.

Parameters:
<PortTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PIDerror <SearchName>,<PIDError>
SEARch:TRIGger:USB:PIDerror? <SearchName>

Defines, whether a search for any PID error shall be activated or not.

Parameters:

<PIDError>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SCONdition <SearchName>,<OperatorSEU>
SEARch:TRIGger:USB:SCONdition? <SearchName>

Sets the operator to set a specific SEU or an SEU range. The SEU values are set with `SEARch:TRIGger:USB:SMIN` and `SEARch:TRIGger:USB:SMAX`.

For SEU, see "SEU check" on page 801.

Parameters:

<OperatorSEU>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one SEU value to be set using `SEARch:TRIGger:USB:SMIN`.

INRange | OORange
In range, out of range. These conditions require a range of SEU values to be set using `SEARch:TRIGger:USB:SMIN` and `SEARch:TRIGger:USB:SMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SMIN <SearchName>,<SEU>
SEARch:TRIGger:USB:SMIN? <SearchName>

Specifies an SEU, or sets the start value of an SEU range.

Parameters:
<SEU>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SMAX <SearchName>,<SEUTo>
SEARch:TRIGger:USB:SMAX? <SearchName>

Sets the the end value of an SEU range if `TRIGger<m>:USB:SCONdition` is set to `INRange` or `OORange`.

Parameters:
<SEUTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SDATa <SearchName>,<DataPacket>
SEARch:TRIGger:USB:SDATa? <SearchName>

Enables the search for any data packet.

Parameters:
<DataPacket> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:USB:SERRor? <SearchName>

Enables the search for various errors.

Parameters:

<ErrorCondition>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SHANdshake <SearchName>,<HandshakePacket>
SEARch:TRIGger:USB:SHANdshake? <SearchName>

Defines, whether the search for any handshake packet shall be activated or not.

Parameters:

<HandshakePacket>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SSOP <SearchName>,<StartofPacket>
SEARch:TRIGger:USB:SSOP? <SearchName>

Defines, whether a start of packet (SOP) search shall be activated or not.

Parameters:

<StartofPacket>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SSPE <SearchName>,<SpecialPacket>
SEARch:TRIGger:USB:SSPE? <SearchName>

Defines, whether the search for any special PID packet shall be activated or not.

Parameters:

<SpecialPacket>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SPEC <SearchName>,<USBSpecialType>
SEARch:TRIGger:USB:SPEC? <SearchName>

Defines, which special PID packet type is searched for: "PREamble", "ERR", "SPLit", "PING", or "Any" special PID packet.

Parameters:

<USBSpecialType> ANY | PREamble | ERR | SPLit | PING

ANY
Searches for any of the special PID packet types listed below

PREamble
Searches for any host-issued preamble token (enables downstream bus traffic to low speed USB devices)

ERR
Searches for any SPLIT transaction error handshake token (reuses PRE value)

SPLit
Searches for any high speed SPLIT transaction token

PING
Searches for any high speed flow control probe for a bulk/control endpoint

*RST: PREamble

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:STCO <SearchName>,<SC>
SEARch:TRIGger:USB:STCO? <SearchName>

Defines, which Start- / Complete- (SC) split transaction type is searched for: SSPLIT or CSPLIT.

Parameters:

<SC> ONE | ZERO | DC

ONE
SC = 1 represents a complete-split (CSPLIT) transaction

ZERO
SC = 0 represents a start-split (SSPLIT) transaction

DC
SC = X represents "don't care" (DC)

*RST: DC

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:STOKen <SearchName>,<TokenPacket>
SEARch:TRIGger:USB:STOKen? <SearchName>

Defines, whether a search for any token packet shall be activated or not.

Parameters:

<TokenPacket> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:TCONdition <SearchName>,<OperatorET>
SEARch:TRIGger:USB:TCONdition? <SearchName>

Sets the operator to set a specific endpoint type (ET) or an ET range. The ET values are set with `SEARch:TRIGger:USB:TMIN` and `SEARch:TRIGger:USB:TMAX`.

Parameters:

<OperatorET> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The endpoint type is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one ET value to be set using `SEARch:TRIGger:USB:TMIN`.

INRange | OORange
In range, out of range. These conditions require a range of ET values to be set using `SEARch:TRIGger:USB:TMIN` and `SEARch:TRIGger:USB:TMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:TMIN <SearchName>,<ET>
SEARch:TRIGger:USB:TMIN? <SearchName>

Specifies an endpoint type (ET), or sets the start value of an ET range.

Parameters:

<ET>

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:USB:TMAX <SearchName>,<ETTo>
SEARch:TRIGger:USB:TMAX? <SearchName>

Sets the the end value of an endpoint type (ET) range if `TRIGger<m>:USB:TCONdition` is set to `INRange` or `OORange`.

Parameters:
<ETTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:TOKen <SearchName>,<USBTokenType>
SEARch:TRIGger:USB:TOKen? <SearchName>

Defines, which token packet type is searched for: "OUT", "IN", "SOF", "SETUP", or "Any" token packet.

Parameters:
<USBTokenType> ANY | OUT | IN | SOF | SETup

ANY
Searches for any of the token packet types listed below.

OUT
Searches for an OUT token.

IN
Searches for an IN token.

SOF
Searches for a start of frame (SOF) token.

SETup
Searches for a SETup token.

*RST: OUT

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WADD <SearchName>,<AddressCheck>
SEARch:TRIGger:USB:WADD? <SearchName>

Defines, whether a search for any address field shall be activated or not.

Parameters:
<AddressCheck> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WEND <SearchName>,<EndpointCheck>
SEARch:TRIGger:USB:WEND? <SearchName>

Defines, whether a search for any endpoint shall be activated or not.

Parameters:
<EndpointCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WETCheck <SearchName>,<WithETCheck>
SEARch:TRIGger:USB:WETCheck? <SearchName>

Defines, whether a search for any Endpoint Type (ET) shall be activated or not.

Parameters:
<WithETCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WFRN <SearchName>,<FrameNoCheck>
SEARch:TRIGger:USB:WFRN? <SearchName>

Defines, whether a search for any frame number shall be activated or not.

Parameters:
<FrameNoCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WPAY <SearchName>,<PayloadCheck>
SEARch:TRIGger:USB:WPAY? <SearchName>

Defines, whether a search for any payload data shall be activated or not.

Parameters:
<PayloadCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WPID <SearchName>,<WithPIDCheck>
SEARch:TRIGger:USB:WPID? <SearchName>

Defines, whether a search for any packet ID error shall be activated or not.

Parameters:
<WithPIDCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WPOR <SearchName>,<WithPortCheck>
SEARch:TRIGger:USB:WPOR? <SearchName>

Defines, whether a search for any port shall be activated or not.

Parameters:
<WithPortCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WSEU <SearchName>,<WithSEUCheck>
SEARch:TRIGger:USB:WSEU? <SearchName>

Defines, whether a search for any SEU shall be activated or not.

For SEU, see "SEU check" on page 801.

Parameters:
<WithSEUCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WSTC <SearchName>,<WithSCCheck>
SEARch:TRIGger:USB:WSTC? <SearchName>

Defines, whether a search for any Start / Complete (SC) shall be activated or not.

Parameters:
<WithSCCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

17.17.17.5 Search Results

In all `SEARch:RESult:USB:PACKet<m>` commands, the suffix <m> selects the packet number in the list of search results.

SEARch:RESult:USB:PACKet<m>:ADDRess?.....1789
SEARch:RESult:USB:PACKet<m>:CRC?.....1789
SEARch:RESult:USB:PACKet<m>:DATA?.....1790
SEARch:RESult:USB:PACKet<m>:ENDPoint?.....1790
SEARch:RESult:USB:PACKet<m>:ET?.....1790
SEARch:RESult:USB:PACKet<m>:FRAMe?.....1790
SEARch:RESult:USB:PACKet<m>:PID?.....1791
SEARch:RESult:USB:PACKet<m>:PORT?.....1791
SEARch:RESult:USB:PACKet<m>:SC?.....1792
SEARch:RESult:USB:PACKet<m>:SEU?.....1792
SEARch:RESult:USB:PACKet<m>:STARt?.....1792
SEARch:RESult:USB:PACKet<m>:STATus?.....1793
SEARch:RESult:USB:PACKet<m>:STOP?.....1793
SEARch:RESult:USB:PCOunt?.....1793

SEARch:RESult:USB:PACKet<m>:ADDRess? <SearchName>

Returns the packet address for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Addr> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:CRC? <SearchName>

Returns the packet CRC (Cyclic Redundancy Code) for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CRC> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:DATA? <SearchName>

Returns the payload data from the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:USB:PACKet<m>:ENDPoint? <SearchName>

Returns the endpoint for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Endp> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:ET? <SearchName>

Returns the endpoint type (ET) for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ET> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:FRAMe? <SearchName>

Returns the frame number for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameNo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:PID? <SearchName>

Returns the packet PID for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<PID> RES | OUT | ACK | DATA0 | DATa0 | PING | SOF | NYET | DATA2 | DATa2 | SPLIT | SPLit | IN | NAK | DATA1 | DATa1 | PRE | SETUP | SETup | STALL | STALI | MDATA | MDATa | UNK

For a description of the return values, see `BUS<m>:USB:PACKet<n>:PID?` on page 1768.

*RST: RES

Usage: Query only

SEARch:RESult:USB:PACKet<m>:PORT? <SearchName>

Returns the port number for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Port> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:SC? <SearchName>

Returns the Start- / Complete-split transaction (SSPLIT / CSPLIT) flag bits for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SC>

Usage: Query only

SEARch:RESult:USB:PACKet<m>:SEU? <SearchName>

Returns the SEU values for the selected packet number within the search result.

For SEU, see "SEU check" on page 801.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SEU> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:STARt? <SearchName>

Returns the start time of the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USB:PACKet<m>:STATus? <SearchName>

Returns the status of the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | PID | CRC | BTST | GLITCH | GLITch | BYTE
*RST: OK

Usage: Query only

SEARch:RESult:USB:PACKet<m>:STOP? <SearchName>

Returns the stop time of the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USB:PCOunt? <SearchName>

Returns the search result's packet count, i.e. the number of packets found in the search result.

Query parameters:
<SearchName>

Return values:
<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

17.17.18 USBPD (Option R&S RTE-K63)

- Configuration.....1794
- Trigger.....1796
- Decode Results.....1802
- Search Settings.....1807
- Search Results.....1813

17.17.18.1 Configuration

In all `BUS<m>:USBPd` commands, the suffix <m> selects the serial bus.

BUS<m>:USBPd:DETail.....1794
BUS<m>:USBPd:HYSTeresis.....1794
BUS<m>:USBPd:SOURce.....1794
BUS<m>:USBPd:THReshold.....1795
BUS<m>:USBPd:THRBottom.....1795
BUS<m>:USBPd:THRMid.....1795
BUS<m>:USBPd:THRTop.....1796

BUS<m>:USBPd:DETail <Detail>

If enabled, the data words are broken down into subframes. If not enabled the data words are displayed as 32-bit data words.

Suffix:

<m>	1..4

Parameters:

<Detail>	ON | OFF	
	*RST:	OFF

BUS<m>:USBPd:HYSTeresis <Hysteresis>

Sets a value for the hysteresis of the data.

Suffix:

<m>	1..4

Parameters:

<Hysteresis>	Range:	-10 to 10
	Increment:	0.01
	*RST:	0.05
	Default unit:	V

BUS<m>:USBPd:SOURce <SourceData>

Selects the source for the data signal in the USBPD protocol.

For triggering on a serial bus, analog input channels are required.

Suffix:

<m> 1..4

Parameters:

<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USBPd:THReshold <Threshold>

Sets the threshold value of the data.

Suffix:

<m> 1..4

Parameters:

<Threshold> Range: -10 to 10
Increment: 0.01
*RST: 0.6
Default unit: V

BUS<m>:USBPd:THRBottom <ThresholdBot>

Sets the bottom threshold (for the low current) for the current advertisment mode.

Suffix:

<m> 1..4

Parameters:

<ThresholdBot> Range: -10 to 10
Increment: 0.01
*RST: 0.2
Default unit: V

BUS<m>:USBPd:THRMid <ThresholdMid>

Sets the middle threshold (at default USB Type-C currnet) for the current advertisment mode.

Suffix:

<m> 1..4

Parameters:

<ThresholdMid> Range: -10 to 10
Increment: 0.01
*RST: 0.66
Default unit: V

BUS<m>:USBPd:THRTop <ThresholdTop>

Sets the top threshold (at USB Type-C currnet of 1.5 A) for the current advertisment mode.

Suffix:

<m> 1..4

Parameters:

<ThresholdTop> Range: -10 to 10
Increment: 0.01
*RST: 1.23
Default unit: V

17.17.18.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

There are two commands for each parameter, that you can use for defining the USBPD settings.

For example, to set the *Frame type =Control Frame* > *Field =MsgID* >*Data* value you can use one of the following commands:

- `TRIGger:USBPD:FRAMe3:FLD3:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number. For an overview, see Overview of frame and field numbers for USBPD.
- `TRIGger:USBPD:DMIN "Control", "MsgID", 01100`
  Defines the parameter by using the frame and field name.

Table 17-12 gives an overview of the available frame and field numbers. Those numbers follow the GUI numbers.

Table 17-12: Overview of frame and field numbers for USBPD

Frame number	Frame name	Field number	Field name
1	Test Frame	-	-
2	Reset	-	-
3	Control frame	1	Extended
		2	NumDataObjs
		3	MsgID

Frame number	Frame name	Field number	Field name
		4	PwrRole/Plug
		5	Rev
		6	DataRole
		7	MsgType
4	Data Frame	1	Extended
		2	NumDataObjs
		3	MsgID
		4	PwrRole/Plug
		5	Rev
		6	DataRole
		7	MsgType
		8	DATA
5	Extended	1	Extended
		2	NumDataObjs
		3	MsgID
		4	PwrRole/Plug
		5	Rev
		6	DataRole
		7	MsgType
		8	Chunked
		9	Chunk Num
		10	Req Chunk
		11	Data Size
		12	DATA

TRIGger<m>:USBPd:ERENable....1798
TRIGger<m>:USBPd:ERRor<n>:ENABle....1798
TRIGger<m>:USBPd:FRENable....1798
TRIGger<m>:USBPd:FRAMe<n>:ENABle....1798
TRIGger<m>:USBPd:FIENable....1799
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle....1799
TRIGger<m>:USBPd:BIT....1799
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT....1799
TRIGger<m>:USBPd:DMAX....1800
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX....1800
TRIGger<m>:USBPd:DMIN....1800
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN....1800

TRIGger<m>:USBPd:DOPerator 1800
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator 1800
TRIGger<m>:USBPd:IMAX 1801
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX 1801
TRIGger<m>:USBPd:IMIN 1801
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN 1801
TRIGger<m>:USBPd:IOPerator 1802
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator 1802

TRIGger<m>:USBPd:ERENable <ErrorName>, <Enabler>
TRIGger<m>:USBPd:ERRor<n>:ENABle <Enable>

Enables or disables the checking condition for a specific error in the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the error number.

Parameters:

<Enable> ON | OFF
*RST: ON

Example:
`TRIGger:USBPd:ERRor1:ENABle ON`
Enables CRC Error.
`TRIGger:USBPd:ERENable "Length Error",ON`
Enables Length Error.

TRIGger<m>:USBPd:FRENable <Frame>, <Enabler>
TRIGger<m>:USBPd:FRAMe<n>:ENABle <Enable>

Enables or disables the checking condition for the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

Parameters:

<Enable> ON | OFF
*RST: OFF

Example:
`TRIGger:USBPd:FRAMe1:ENABle ON`
Enables searching for for the Test frame.
`TRIGger:USBPd:FRENable "Control Frame" ON`
Enables searching for the Control frame.

TRIGger<m>:USBPd:FIENable <Frame>, <Field>, <Enabler>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle <CondEnabler>

Enables or disables the checking condition for a specific data pattern in the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame.

<o> *
Specifies the field number within the frame.

Parameters:

<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Example:
`TRIGger:USBPd:FRAMe3:FLD1:ENABle ON`
Enables the checking condition for the Extended field of the Control frame.
`TRIGger:USBPd:FRENable "Data Frame", "Data Role" ON`
Enables the checking condition for the Data Role field of the Data frame.

TRIGger<m>:USBPd:BIT <Frame>, <Field>, <Bit>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT <BitState>

Sets the bit state of a field that only consists of one bit.

Suffix:

<m> 1..3

<n> *

<o> *
Specifies the field number within the frame.

Parameters:

<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" (DC) = X

*RST: DC

TRIGger<m>:USBPd:DMAX <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX <DataMax>

Sets the end value of a data pattern range if TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX is set to INRange or OORange.

Suffix:

<m>	1..3
<n>	* Specifies the field number within the frame.
<o>	*

Parameters:

<DataMax>	Specifies the name of the generic trigger setting frame, see BUS<m>:USBPd:RESult:FRAMe<n>:TYPE? on page 1807.

TRIGger<m>:USBPd:DMIN <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN <DataMin>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:

<m>	1..3
<n>	* Specifies the field number within the frame.
<o>	*

Parameters:

<DataMin>

TRIGger<m>:USBPd:DOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator <DataOperator>

Sets the operator for the data pattern in the selected field of the selected frame.

Suffix:

<m>	1..3
<n>	* Specifies the field number within the frame.
<o>	*

Parameters:

<DataOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN` and `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX`.

*RST: EQUal

TRIGger<m>:USBPd:IMAX <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX <IndexMax>

Sets the end value of an index range if `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator` on page 1802 is set to `INRange` or `RANGe`.

Suffix:

<m>	1..3
<n>	* Specifies the field number within the frame.
<o>	*

Parameters:

<IndexMax>	Specifies the end value for the index range within the field. The index range, increment and *RST values depend on the field type.

TRIGger<m>:USBPd:IMIN <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN <IndexMin>

Specifies the index, or sets the start value of an index range.

Suffix:

<m>	1..3
<n>	* Specifies the field number within the frame.
<o>	*

Parameters:

<IndexMin>	Specifies the index value or index start value within the field. The index range, increment and *RST values depend on the field type.

TRIGger<m>:USBPd:IOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:

<m>	1..3
<n>	*
<o>	*

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN` and `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX`.

*RST: INRange

17.17.18.3 Decode Results

In all `BUS<m>:USBPd:RESult:FRAMe<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the frame number in the decode table.

BUS<m>:USBPd:RESult:FCOunt?......1802
BUS<m>:USBPd:RESult:FRAMe<n>:DATA?......1803
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL?......1803
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus?......1803
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt?......1804
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP?......1804
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE?......1805
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL?......1805
BUS<m>:USBPd:RESult:FRAMe<n>:INFO?......1805
BUS<m>:USBPd:RESult:FRAMe<n>:STARt?......1806
BUS<m>:USBPd:RESult:FRAMe<n>:STATe?......1806
BUS<m>:USBPd:RESult:FRAMe<n>:STOP?......1806
BUS<m>:USBPd:RESult:FRAMe<n>:TYPE?......1807

BUS<m>:USBPd:RESult:FCOunt?

Returns the number of decoded frames in the selected serial bus.

Suffix:

<m> 1..4

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:DATA?

Returns the data from the frame with the selected frame number (for the selected serial bus). This is the hex value of the first four bytes of data fields, prefixed by the total number of data fields (in square brackets), e.g. `[5]FF008041`.

Suffix:

<m> 1..4

<n> *

Return values:

<Data>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FrameByteValue>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus?

Returns the status of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FieldState> OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt?

Returns the start time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP?

Returns the stop time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE?

Returns the type of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:
<FieldType>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:
<FrameByteValue>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:INFO?

Returns information on the frame with the selected frame number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	*

Return values:
<Info>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:STARt?

Returns the start time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	*

Return values:

<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:STATe?

Returns the status of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	*

Return values:

<State>
OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:STOP?

Returns the stop time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m>	1..4
<n>	*

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:TYPE?

Returns the frame type for the selected serial bus and frame number.

Suffix:

<m>	1..4
<n>	*

Return values:

<FrameType> STARt | RESet | CTRL | DATA | BIST | RQST | SINK | SRC | VEND | BATT | ALRT | XMSG | XSRC | XSTA | XGBC | XGBS | XBAC | XGMI | XMFI | XSRQ | XSRS | XFRQ | XFRS | TEST | LOWP | UNKNown

For a description of the frame types, see "Frame packet types" on page 816.

*RST: UNKNown

Usage: Query only

17.17.18.4 Search Settings

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

There are two commands for each parameter, that you can use for defining the USBPD settings.

For example, to set the *Frame type =Control Frame > Field =MsgID >Data* value you can use one of the following commands:

- `SEARch:TRIGger:USBPD:FRAMe3:FLD3:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number. For an overview, see Table 17-12.
- `SEARch:TRIGger:USBPD:DMIN "Control", "MsgID", 01100`
  Defines the parameter by using the frame and field name.

SEARch:TRIGger:USBPd:ERENable....1808
SEARch:TRIGger:USBPd:ERRor<m>:ENABle....1808
SEARch:TRIGger:USBPd:FRENable....1808
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle....1808
SEARch:TRIGger:USBPd:FIENable....1809
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle....1809
SEARch:TRIGger:USBPd:BIT....1809
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT....1809
SEARch:TRIGger:USBPd:DMAX....1810
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX....1810
SEARch:TRIGger:USBPd:DMIN....1810
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN....1810
SEARch:TRIGger:USBPd:DOPerator....1811
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator....1811
SEARch:TRIGger:USBPd:IMAX....1811
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX....1811
SEARch:TRIGger:USBPd:IMIN....1812
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN....1812
SEARch:TRIGger:USBPd:IOPerator....1812
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator....1812

SEARch:TRIGger:USBPd:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:USBPd:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:USBPd:ERRor<m>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific error in the selected field of the selected frame.

Suffix:
<m> *
Specifies the error number.

Parameters:
<Enable> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Example: `SEARch:TRIGger:USBPd:ERRor1:ENABle ON`
Enables searching for CRC Error.
`SEARch:TRIGger:USBPd:ERENable "Length Error",ON`
Enables searching for Length Error.

SEARch:TRIGger:USBPd:FRENable <SearchName>,<Frame>, <Enabler>
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle? <SearchName>
Suffix:
<m> *
Specifies the frame number.

Parameters:

<Enable> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName>

Example: `SEARch:TRIGger:USBPd:FRAMe1:ENABle ON`

Enables the checking condition for the Test frame.

`SEARch:TRIGger:USBPd:FRENable "Control Frame" ON`

Enables the checking condition for the Control frame.

SEARch:TRIGger:USBPd:FIENable <SearchName>,<Frame>, <Field>, <Enabler>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle <SearchName>,<CondEnabler>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific data pattern in the selected field of the selected frame.

Suffix:

<m> *

Specifies the frame.

<n> *

Specifies the field number within the frame.

Parameters:

<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:

<SearchName> String with the name of the search.

Example: `SEARch:TRIGger:USBPd:FRAMe3:FLD1:ENABle ON`

Enables searching for the Extended field of the Control frame.

`SEARch:TRIGger:USBPd:FRENable "Data Frame", "Data Role" ON`

Enables searching for the Data Role field of the Data frame.

SEARch:TRIGger:USBPd:BIT <SearchName>,<Frame>, <Field>, <Bit>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT <SearchName>,<BitState>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT? <SearchName>

Sets the bit state of a field to be searched that only consists of one bit.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" = X

*RST: DC

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<DataMax> Specifies the name of the generic trigger setting frame, see `SEARch:RESult:USBPd:FRAMe<m>:TYPE?` on page 1817.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<DataMin> Specifies the name of the generic trigger setting frame, see `SEARch:RESult:USBPd:FRAMe<m>:TYPE?` on page 1817.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator <SearchName>,<DataOperator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN` and `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:IMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX <SearchName>,<IndexMax>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX? <SearchName>

Sets the end value of an index range if `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<IndexMax> Specifies the end value for the index range within the field.
The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:IMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN <SearchName>,<IndexMin>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN? <SearchName>

Specifies the index at which the data is to be searched, or sets the start value of an index range in which the data is to be searched.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<IndexMin> Specifies the index value or the start value of an index range within the field.
The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator
<SearchName>,<IndexOperator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN.

INRange | RANGe
This condition requires a range of endpoint values to be set using SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN and SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX.

*RST: INRange

Parameters for setting and query:
<SearchName> String with the name of the search.

17.17.18.5 Search Results

In all SEARch:RESult:USBPd:FRAMe<m> commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:USBPd:FCOunt?... 1813
SEARch:RESult:USBPd:FRAMe<m>:DATA?... 1813
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL?.. 1814
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus?... 1814
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt?... 1815
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP?.. 1815
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE?.. 1815
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL?.. 1816
SEARch:RESult:USBPd:FRAMe<m>:INFO?... 1816
SEARch:RESult:USBPd:FRAMe<m>:STARt?... 1816
SEARch:RESult:USBPd:FRAMe<m>:STATe?... 1817
SEARch:RESult:USBPd:FRAMe<m>:STOP?... 1817
SEARch:RESult:USBPd:FRAMe<m>:TYPE?... 1817

SEARch:RESult:USBPd:FCOunt? <SearchName>

Returns the number of decoded frames in the search result.

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:DATA? <SearchName>

Searches for data from the frame with the selected frame number (for the selected serial bus). This is the hex value of the first four bytes of data fields, prefixed by the total number of data fields (in square brackets), e.g. [5]FF008041.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL? <SearchName>

Returns the formatted value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ValueFormatted>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus? <SearchName>

Returns the field status of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<FieldState> OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt? <SearchName>

Returns the start time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP? <SearchName>

Returns the stop time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE? <SearchName>

Returns the field name of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<FieldType>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL? <SearchName>

Returns the value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ByteValue>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:INFO? <SearchName>

Returns specific frame information details of the selected frame in the search result. This information is also shown in the "Info" column of the decode results table.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Info>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:STARt? <SearchName>

Returns the start time of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start>	Range: -100E+24 to 100E+24 Increment: 100E-12 Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:STATe? <SearchName>

Returns the status of the frame with the selected frame number within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<State> OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:STOP? <SearchName>

Returns the stop time of the frame with the selected frame number within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:TYPE? <SearchName>

Returns the frame type for the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> STARt | RESet | CTRL | DATA | BIST | RQST | SINK | SRC | VEND | BATT | ALRT | XMSG | XSRC | XSTA | XGBC | XGBS | XBAC | XGMI | XMFI | XSRQ | XSRS | XFRQ | XFRS | TEST | LOWP | UNKNown

For a description of the frame types, see "Frame packet types" on page 816.

*RST: UNKNown

Usage: Query only

17.17.19 Space Wire (Option R&S RTE-K65)

- Configuration....1818
- Trigger....1821
- Decode Results....1824
- Search Settings....1829
- Search Results....1833

17.17.19.1 Configuration

BUS<m>:SWIRe:BPOSition....1818
BUS<m>:SWIRe:DATA:HYSTeresis....1819
BUS<m>:SWIRe:DATA:SOURce....1819
BUS<m>:SWIRe:DATA:THReshold....1819
BUS<m>:SWIRe:MGAP....1819
BUS<m>:SWIRe:STRBe:HYSTeresis....1820
BUS<m>:SWIRe:STRBe:SOURce....1820
BUS<m>:SWIRe:STRBe:THReshold....1820
BUS<m>:SWIRe:SYSLect....1820
BUS<m>:SWIRe:COUPling....1821
BUS<m>:SWIRe:PRESet....1821

BUS<m>:SWIRe:BPOSition <SyncBitPos>

Sets the bit position, the align position for the manual synchronisation mode. This can be useful when parity errors exist in the signal, and parity check is the main indicator for the decoder to do packet alignment.

Suffix:
<m> 1..4

Parameters:
<SyncBitPos> Range: 0 to 10
Increment: 1
*RST: 0

BUS<m>:SWIRe:DATA:HYSTeresis <HysteresisData>

Sets a value for the hysteresis of the data signal.

Suffix:
<m> 1..4

Parameters:
<HysteresisData> Range: 0 to 0.8
Increment: 0.01
*RST: 0.2
Default unit: V

BUS<m>:SWIRe:DATA:SOURce <SourceData>

Selects the source of the data signal.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
*RST: C1W1

BUS<m>:SWIRe:DATA:THReshold <ThresholdData>

Sets the threshold value for the digitization of the data signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdData> Range: -10 to 10
Increment: 0.1
*RST: 1.25
Default unit: V

BUS<m>:SWIRe:MGAP <MinGapTime>

Sets the minimum duration of a gap. Any inactivity greater than this time will be interpreted as a gap and lead to a resynchronization to the signal.

Suffix:
<m> 1..4

Parameters:

<MinGapTime> Range: 1E-9 to 10E-6
Increment: 100E-9
*RST: 200E-9
Default unit: s

Firmware/Software: FW 3.30. Replaces `BUS<m>:SWIRe:MINGap`.

BUS<m>:SWIRe:STRBe:HYSTeresis <HystStrobe>

Sets a value for the hysteresis of the strobe signal.

Suffix:

<m> 1..4

Parameters:

<HystStrobe> Range: 0 to 0.8
Increment: 0.01
*RST: 0.2
Default unit: V

BUS<m>:SWIRe:STRBe:SOURce <SourceStrobe>

Selects the source of the strobe signal.

Suffix:

<m> 1..4

Parameters:

<SourceStrobe> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
*RST: C2W1

BUS<m>:SWIRe:STRBe:THReshold <ThresholdStrobe>

Sets the threshold value for the digitization of the strobe signal.

Suffix:

<m> 1..4

Parameters:

<ThresholdStrobe> Range: -10 to 10
Increment: 0.1
*RST: 1.25
Default unit: V

BUS<m>:SWIRe:SYSLect <SyncSelector>

Selects the mode for the synchronisation settings, i.e i.e. packet align.

Suffix:
<m> 1..4

Parameters:
<SyncSelector> AUTO | MANual
*RST: AUTO

BUS<m>:SWIRe:COUPling <ThresCpl>

Enables coupling, i.e. the same threshold and hysteresis value is used for the strob and the data signal.

Suffix:
<m> 1..4

Parameters:
<ThresCpl> ON | OFF
*RST: ON

BUS<m>:SWIRe:PRESet <ThresholdPreset>

Prests the threshold and hysteresis value of the strobe and data signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdPreset> V25 | V165 | V125 | V09 | V12 | V0 | MANual

V25: 2.5 V (CMOS 5.0 V)
V165: 1.65 V (CMOS 3.5 V)
V125: 1.25 V (CMOS 1.5 V)
V09: 2.5 V (CMOS 1.8V)
V12: 1.2 V (LVDS single ended)
V0: 0 V (LVDS differential)
MANual: user-defined value

*RST: V12

17.17.19.2 Trigger

TRIGger<m>:SWIRe:CTYPe....1822
TRIGger<m>:SWIRe:DATA:CONDition....1822
TRIGger<m>:SWIRe:DATA:MAX....1822
TRIGger<m>:SWIRe:DATA:MIN....1823
TRIGger<m>:SWIRe:ERRor:ESC....1823
TRIGger<m>:SWIRe:ERRor:PARity....1823
TRIGger<m>:SWIRe:TIME:CONDition....1823
TRIGger<m>:SWIRe:TIME:MAX....1824
TRIGger<m>:SWIRe:TIME:MIN....1824
TRIGger<m>:SWIRe:TYPE....1824

TRIGger<m>:SWIRe:CTYPe <ControlType>

Triggers on a specific control type character.

Suffix:
<m> 1..3

Parameters:
<ControlType> ANY | FCT | EOP | EEP

FCT
Flow Control Token

EOP
Normal End of Packet

EEP
Error End of Packet

*RST: ANY

TRIGger<m>:SWIRe:DATA:CONDition <DataOperator>

Set the condition for the data value. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:SWIRe:DATA:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:SWIRe:DATA:MIN` and `TRIGger<m>:SWIRe:DATA:MAX`.

*RST: EQUal

TRIGger<m>:SWIRe:DATA:MAX <DataPatternMax>

Sets the end value of a data for the data pattern range if `TRIGger<m>:SWIRe:DATA:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataPatternMax>

TRIGger<m>:SWIRe:DATA:MIN <DataPatternMin>

Specifies a data for the data pattern, or sets the the start value of a pattern range.

Suffix:

<m>	1..3

Parameters:

<DataPatternMin>

TRIGger<m>:SWIRe:ERRor:ESC <ErrorESC>

Enables searching for an escape error.

Suffix:

<m>	1..3

Parameters:

<ErrorESC>	ON | OFF *RST: ON

TRIGger<m>:SWIRe:ERRor:PARity <ErrorParity>

Enables triggering on a parity error (even parity).

Suffix:

<m>	1..3

Parameters:

<ErrorParity>	ON | OFF *RST: ON

TRIGger<m>:SWIRe:TIME:CONDition <TimeOperator>

Set the condition for the data value for the time code. You can define an exact value or a value range.

Suffix:

<m>	1..3

Parameters:

<TimeOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange **EQUal | NEQual | LTHan | LETHan | GTHan | GETHan** Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:SWIRe:TIME:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:SWIRe:TIME:MIN` and `TRIGger<m>:SWIRe:TIME:MAX`.

*RST: EQUal

TRIGger<m>:SWIRe:TIME:MAX <TimePatternMax>

Sets the the end value of a data value for the time code range if `TRIGger<m>:SWIRe:TIME:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<TimePatternMax>

TRIGger<m>:SWIRe:TIME:MIN <TimePatternMin>

Specifies a pattern for the data value for the time code, or sets the the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<TimePatternMin>

TRIGger<m>:SWIRe:TYPE <Type>

Sets the trigger type for the SpaceWire analysis.

Suffix:
<m> 1..3

Parameters:
<Type> CTRL | DATA | NULL | TIME | ERRor

*RST: CTRL

17.17.19.3 Decode Results

BUS<m>:SWIRe:RESults:FCOunt 1825
BUS<m>:SWIRe:RESults:FRAMe<n>:DATA? 1825
BUS<m>:SWIRe:RESults:FRAMe<n>:STARt? 1825
BUS<m>:SWIRe:RESults:FRAMe<n>:STATe? 1826
BUS<m>:SWIRe:RESults:FRAMe<n>:STOP? 1826
BUS<m>:SWIRe:RESults:FRAMe<n>:TYPE? 1826
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:FVAL? 1827
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STARt? 1827
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STATus? 1828

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STOP?..1828
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:TYPE?.. 1829
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:VAL?...1829

BUS<m>:SWIRe:RESults:FCOunt <Key>

Returns the number of decoded frames for the selected serial bus.

Suffix:
<m> 1..4

Setting parameters:
<Key>

Return values:
<Count>

BUS<m>:SWIRe:RESults:FRAMe<n>:DATA?

Returns the data value.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:
<Data> 8-bit data value

Example:
```
BUS:SWIRe:RESults:FRAMe2:DATA?
<-- #H12
```

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:
<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:STATe?

Returns the overall state of the frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the serial bus.

Return values:

<State> OK | PAR | ESC | AMB | INComplete

OK
The frame is valid.

PARity
Parity error

ESC
Escape error

AMB
Ambiguous

INComplete
The frame is incomplete

*RST: OK

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.

Return values:

<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:TYPE?

Returns the type of frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Type> DATA | FCT | EOP | EEP | ESC | NULL | TCOD

DATA
Data frame

FCT
Flow control token

EOP
End of packet

EEP
Error end of packet

ESC
Escape

NULL
Null symbol

TCOD
Time code

*RST: DATA

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FrameByteValue>

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STARt?

Returns the start time of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:

<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STATus?

Returns the status of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:

<State>
OK | PAR | ESC | AMB | INComplete

`PAR`: parity error
`ESC`: escape
`AMB`: ambiguous

*RST: OK

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STOP?

Returns the stop time of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:TYPE?

Returns the type of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:

<FrameByteValue>

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m>	1..4
<n>	* Selects the frame number.
<o>	* Selects the field number.

Return values:

<FrameByteValue>	Range: 0 to 65535 Increment: 1 *RST: 0

Usage: Query only

17.17.19.4 Search Settings

SEARch:TRIGger:SWIRe:CTYPe 1830
SEARch:TRIGger:SWIRe:DATA:CONDition 1830
SEARch:TRIGger:SWIRe:DATA:MAX 1831
SEARch:TRIGger:SWIRe:DATA:MIN 1831

SEARch:TRIGger:SWIRe:ERRor:ESC ... 1831
SEARch:TRIGger:SWIRe:ERRor:PARity ... 1831
SEARch:TRIGger:SWIRe:TIME:CONDition ... 1832
SEARch:TRIGger:SWIRe:TIME:MAX ... 1832
SEARch:TRIGger:SWIRe:TIME:MIN ... 1832
SEARch:TRIGger:SWIRe:TYPE ... 1833

SEARch:TRIGger:SWIRe:CTYPe <SearchName>,<ControlType>
SEARch:TRIGger:SWIRe:CTYPe? <SearchName>

Searches for a specific control type character.

Parameters:

<ControlType> ANY | FCT | EOP | EEP

FCT
Flow Control Token

EOP
Normal End of Packet

EEP
Error End of Packet

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:DATA:CONDition <SearchName>,<DataOperator>
SEARch:TRIGger:SWIRe:DATA:CONDition? <SearchName>

Set the condition for the data pattern data value. You can define an exact value or a value range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:SWIRe:DATA:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:SWIRe:DATA:MIN` and `SEARch:TRIGger:SWIRe:DATA:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:DATA:MAX <SearchName>,<DataPatternMax>
SEARch:TRIGger:SWIRe:DATA:MAX? <SearchName>

Sets the the end value of a data type range if `SEARch:TRIGger:SWIRe:DATA:CONDition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternMax> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SWIRe:DATA:MIN <SearchName>,<DataPatternMin>
SEARch:TRIGger:SWIRe:DATA:MIN? <SearchName>

Specifies a data bit pattern, or sets the the start value of a pattern range.

Parameters:

<DataPatternMin>

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SWIRe:ERRor:ESC <SearchName>,<ErrorESC>
SEARch:TRIGger:SWIRe:ERRor:ESC? <SearchName>

Enables triggering on an escape error.

Parameters:

<ErrorESC> ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SWIRe:ERRor:PARity <SearchName>,<ErrorParity>
SEARch:TRIGger:SWIRe:ERRor:PARity? <SearchName>

Enables searching for a parity error (even parity).

Parameters:

<ErrorParity> ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:SWIRe:TIME:CONDition <SearchName>,<TimeOperator>
SEARch:TRIGger:SWIRe:TIME:CONDition? <SearchName>

Set the condition for the data value of the time code. You can define an exact value or a value range.

Parameters:

<TimeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:SWIRe:TIME:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:SWIRe:TIME:MIN` and `SEARch:TRIGger:SWIRe:TIME:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TIME:MAX <SearchName>,<TimePatternMax>
SEARch:TRIGger:SWIRe:TIME:MAX? <SearchName>

Sets the the end value of a data type range for the time code if `SEARch:TRIGger:SWIRe:TIME:CONDition` is set to `INRange` or `OORange`.

Parameters:
<TimePatternMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TIME:MIN <SearchName>,<TimePatternMin>
SEARch:TRIGger:SWIRe:TIME:MIN? <SearchName>

Specifies a data bit pattern for the time code, or sets the the start value of a pattern range.

Parameters:
<TimePatternMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TYPE <SearchName>,<Type>
SEARch:TRIGger:SWIRe:TYPE? <SearchName>

Sets the search type for the SpaceWire analysis.

Parameters:
<Type> CTRL | DATA | NULL | TIME | ERRor
*RST: CTRL

Parameters for setting and query:
<SearchName>

17.17.19.5 Search Results

In all `SEARch:RESult:DPHY:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:SWIRe:FCOunt?....................1833
SEARch:RESult:SWIRe:FRAMe<m>:DATA?....................1833
SEARch:RESult:SWIRe:FRAMe<m>:STARt?....................1834
SEARch:RESult:SWIRe:FRAMe<m>:STATe?....................1834
SEARch:RESult:SWIRe:FRAMe<m>:STOP?....................1834
SEARch:RESult:SWIRe:FRAMe<m>:TYPE?....................1835

SEARch:RESult:SWIRe:FCOunt? <SearchName>

Returns the number of frames within the search result for the selected serial bus.

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:DATA? <SearchName>

Returns the data value for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:STARt? <SearchName>

Returns the start time of the specified frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:STATe? <SearchName>

Returns the overall state of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | PAR | ESC | AMB | INComplete

OK
The frame is valid

PARity
Parity error

ESC
Escape Error

AMB
Ambiguous

INComplete
The frame is incomplete

*RST: OK

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:STOP? <SearchName>

Returns the end time for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:TYPE? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> DATA | FCT | EOP | EEP | ESC | NULL | TCOD

DATA
Data frame

FCT
Flow control token

EOP
End of packet

EEP
Error end of packet

ESC
Escape

NULL
Null symbol

TCOD
Time code

*RST: DATA

Usage: Query only

17.17.20 CXPI (Option R&S RTE-K76)

- Configuration....1836
- Trigger....1839
- Decode Results....1848
- Search Settings....1851
- Search Result Settings....1860

17.17.20.1 Configuration

BUS<m>:CXPI:BITRate:ENABle....1836
BUS<m>:CXPI:BITRate:VALue....1836
BUS<m>:CXPI:RESult:BITRate?....1836
BUS<m>:CXPI:DORD....1837
BUS<m>:CXPI:HYSTeresis....1837
BUS<m>:CXPI:POLarity....1837
BUS<m>:CXPI:THReshold....1837
BUS<m>:CXPI:SDATa....1838
BUS<m>:CXPI:IBS:MAX....1838
BUS<m>:CXPI:IBS:MIN....1838
BUS<m>:CXPI:IFS:MAX....1839
BUS<m>:CXPI:IFS:MIN....1839

BUS<m>:CXPI:BITRate:ENABle <BitrateEnable>

Enables the manual setting of a bit rate. You can set the bitrate value with `BUS<m>:CXPI:BITRate:VALue`.

Suffix:

<m>	1..4

Parameters:

<BitrateEnable>	ON | OFF	
	*RST:	OFF

BUS<m>:CXPI:BITRate:VALue <Bitrate>

Sets the number of transmitted bits per second. To set this value, you have to enable the bitrate first with `BUS<m>:CXPI:BITRate:ENABle`.

Suffix:

<m>	1..4

Parameters:

<Bitrate>	Range:	1000 to 100000
	Increment:	100
	*RST:	19200
	Default unit:	bps

BUS<m>:CXPI:RESult:BITRate?

Queries the measured average bit rate, which is calculated if `BUS<m>:CXPI:BITRate:ENABle` is set to `OFF`.

Suffix:

<m>	1..4

Return values:

<AverageBitrate>	Range: 1000 to 100000 Increment: 100 *RST: 19200 Default unit: bps

Usage: Query only

BUS<m>:CXPI:DORD <DisplayOrder>

Selects the order in which the signal is displayed in the honeycomb.

If `TRANsmitted` is selected, the signal is displayed in the order it occurs. If `LOGical` is selected, the signal is displayed according to the definition of the standard (MSB bit order).

Suffix:
<m> 1..4

Parameters:
<DisplayOrder> TRANsmitted | LOGical

*RST: TRANsmitted

BUS<m>:CXPI:HYSTeresis <ThresHyst>

Sets a value for the hysteresis.

Suffix:
<m> 1..4

Parameters:

<ThresHyst>	Range: -12 to 12 Increment: 0.01 *RST: 1 Default unit: V

BUS<m>:CXPI:POLarity <Polarity>

Sets the polarity of the transmitted waveform to normal (high = 1) or inverted (low = 1).

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVert

*RST: NORMal

BUS<m>:CXPI:THReshold <Threshold>

Sets the threshold value for the signal.

Suffix:

<m> 1..4

Parameters:

<Threshold> Range: -15 to 12
Increment: 0.01
*RST: 6
Default unit: V

BUS<m>:CXPI:SDATa <SourceData>

Selects the input channel for the signal.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTE-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Suffix:

<m> 1..4

Parameters:

<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

For triggering on a serial bus, analog or digital channel sources are required.

*RST: C1W1

BUS<m>:CXPI:IBS:MAX <ExpectedIBSMax>

Sets the maximum for the inter-byte-space range.

Suffix:

<m> 1..4

Parameters:

<ExpectedIBSMax> Range: 0 to 15
Increment: 1
*RST: 10

BUS<m>:CXPI:IBS:MIN <ExpectedIBSMin>

Sets the minimum for the inter-byte-space range.

Suffix:

<m> 1..4

Parameters:

<ExpectedIBSMin>	Range:	0 to 15
	Increment:	1
	*RST:	0

BUS<m>:CXPI:IFS:MAX <ExpectedIFSMax>

Sets the maximum for the inter-frame-space range.

Suffix:
<m> 1..4

Parameters:

<ExpectedIFSMax>	Range:	16 to 100000
	Increment:	1
	*RST:	1000

BUS<m>:CXPI:IFS:MIN <ExpectedIFSMin>

Sets the minimum for the inter-frame-space range.

Suffix:
<m> 1..4

Parameters:

<ExpectedIFSMin>	Range:	16 to 1000
	Increment:	1
	*RST:	21

17.17.20.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:CXPI:CT....1840
TRIGger<m>:CXPI:DATA:DCONdition....1840
TRIGger<m>:CXPI:DATA:DMAX....1841
TRIGger<m>:CXPI:DATA:DMIN....1841
TRIGger<m>:CXPI:DATA:ICONdition....1841
TRIGger<m>:CXPI:DATA:IMAX....1841
TRIGger<m>:CXPI:DATA:IMIN....1842
TRIGger<m>:CXPI:DEXTension:CONDition....1842
TRIGger<m>:CXPI:DEXTension:MAX....1842
TRIGger<m>:CXPI:DEXTension:MIN....1843
TRIGger<m>:CXPI:DLC:CONDition....1843

TRIGger<m>:CXPI:DLC:MAX....1843
TRIGger<m>:CXPI:DLC:MIN....1843
TRIGger<m>:CXPI:ERRor:CRC....1844
TRIGger<m>:CXPI:ERRor:DLC....1844
TRIGger<m>:CXPI:ERRor:IBS....1844
TRIGger<m>:CXPI:ERRor:IFS....1844
TRIGger<m>:CXPI:ERRor:LENGth....1845
TRIGger<m>:CXPI:ERRor:PARity....1845
TRIGger<m>:CXPI:ERRor:UART....1845
TRIGger<m>:CXPI:FID:CONDition....1845
TRIGger<m>:CXPI:FID:MAX....1846
TRIGger<m>:CXPI:FID:MIN....1846
TRIGger<m>:CXPI:NM....1846
TRIGger<m>:CXPI:TYPE....1846

TRIGger<m>:CXPI:CT <CT>

Sets the value of the counter (CT), 2 bits indicating the continuity of the frame.

Suffix:
<m> 1..3

Parameters:
<CT>

TRIGger<m>:CXPI:DATA:DCONdition <DataOperator>

Sets the condition for the data pattern. You can define no value, an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
Any data pattern is considered

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:CXPI:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:DATA:DMIN` and `TRIGger<m>:CXPI:DATA:DMAX`.

*RST: OFF

TRIGger<m>:CXPI:DATA:DMAX <DataMax>

Sets the end value of a data type range if TRIGger<m>:CXPI:DATA:DCONdition is set to INRange or OORange.

Suffix:
<m> 1..3

Parameters:
<DataMax>

TRIGger<m>:CXPI:DATA:DMIN <DataMin>

Specifies a data bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataMin>

TRIGger<m>:CXPI:DATA:ICONdition <DataIdxOperator>

Set the condition for the data index value. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding TRIGger<m>:CXPI:DATA:IMIN command.

INRange = RANGe
In range: Set the minimum and maximum value using TRIGger<m>:CXPI:DATA:IMIN and TRIGger<m>:CXPI:DATA:IMAX.

*RST: INRange

TRIGger<m>:CXPI:DATA:IMAX <DataIndexMax>

Sets the end value of a data index range if TRIGger<m>:CXPI:DATA:ICONdition is set to INRange or RANGe.

Suffix:
<m> 1..3

Parameters:

<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

TRIGger<m>:CXPI:DATA:IMIN <DataIndexMin>

Specifies a data index bit pattern, or sets the start value of a pattern range.

Suffix:

<m> 1..3

Parameters:

<DataIndexMin> Range: 1 to 0
Increment: 1
*RST: 1

TRIGger<m>:CXPI:DEXTension:CONDition <DLCEXTOperator>

Set the condition for the DLC extend value. You can define an exact value or a value range.

Suffix:

<m> 1..3

Parameters:

<DLCEXTOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:CXPI:DEXTension:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:DEXTension:MIN` and `TRIGger<m>:CXPI:DEXTension:MAX`.

*RST: EQUal

TRIGger<m>:CXPI:DEXTension:MAX <DLCEXTMax>

Sets the end value of a DLC extend range if `TRIGger<m>:CXPI:DEXTension:CONDition` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3

Parameters:

<DLCEXTMax>

TRIGger<m>:CXPI:DEXTension:MIN <DLCEXTMin>

Specifies a data extension bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DLCEXTMin>

TRIGger<m>:CXPI:DLC:CONDition <DLCOperator>

Set the condition for the DLC. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DLCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one DLC pattern to be set with the corresponding `TRIGger<m>:CXPI:DLC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:DLC:MIN` and `TRIGger<m>:CXPI:DLC:MAX`.

*RST: EQUal

TRIGger<m>:CXPI:DLC:MAX <DLCMax>

Sets the end value of a data type range if `TRIGger<m>:CXPI:DLC:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DLCMax>

TRIGger<m>:CXPI:DLC:MIN <DLCMin>

Specifies a DLC pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DLCMin>

TRIGger<m>:CXPI:ERRor:CRC <ErrorCRC>

Triggers on a cyclic redundancy check error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

The transmitting node calculates the cyclic redundancy check (CRC) value of a frame and stores it into the CRC field of the frame. CRC error occurs when this CRC field value differs from the value calculated by the receiving node.

Suffix:
<m> 1..3

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:DLC <ErrorDLC>

Triggers on a data length code (DLC) error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

DLC error occurs when the value of the DLC field is different from the data byte field value.

Suffix:
<m> 1..3

Parameters:
<ErrorDLC> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:IBS <ErrorIBS>

Triggers on an inter-byte-space (IBS) error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

Suffix:
<m> 1..3

Parameters:
<ErrorIBS> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:IFS <ErrorIFS>

Triggers on an inter-frame-space (IFS) error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

Suffix:
<m> 1..3

Parameters:

<ErrorIFS>	ON | OFF *RST: ON

TRIGger<m>:CXPI:ERRor:LENGth <ErrorLength>

Triggers on a length error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

Suffix:

<m>	1..3

Parameters:

<ErrorLength>	ON | OFF *RST: ON

TRIGger<m>:CXPI:ERRor:PARity <ErrorParity>

Triggers on a parity error indicating a transmission error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

Suffix:

<m>	1..3

Parameters:

<ErrorParity>	ON | OFF *RST: ON

TRIGger<m>:CXPI:ERRor:UART <ErrorUart>

Triggers on an UART error, if TRIGger<m>:CXPI:TYPE is set to ERRor.

Suffix:

<m>	1..3

Parameters:

<ErrorUart>	ON | OFF *RST: ON

TRIGger<m>:CXPI:FID:CONDition <FrameIDOperator>

Set the condition for the frame ID. You can define an exact value or a value range.

Suffix:

<m>	1..3

Parameters:

<FrameIDOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:CXPI:FID:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:FID:MIN` and `TRIGger<m>:CXPI:FID:MAX`.

*RST: EQUal

TRIGger<m>:CXPI:FID:MAX <FrameIDMax>

Sets the end value of a frame ID range if `TRIGger<m>:CXPI:FID:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<FrameIDMax>

TRIGger<m>:CXPI:FID:MIN <FrameIDMin>

Specifies a frame ID pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<FrameIDMin>

TRIGger<m>:CXPI:NM <NM>

Sets the value of the network management (NM) field.

Suffix:
<m> 1..3

Parameters:
<NM>

TRIGger<m>:CXPI:TYPE <Type>

Selects the trigger type for the CXPI analysis.

Suffix:
<m> 1..3

Parameters:
<Type> STARt | NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPD | ERRor

STARt
Triggers on the STARt of the frame.

NORMal
Triggers on a NORMal frame. Optionally you can define a frame ID condition, an NM, a CT, a DLC condition, a data pattern condition and an index condition.

NPOLI
Triggers on a Normal POLL frame. Optionally you can define a frame ID condition, an NM, a CT, a DLC condition, a data pattern condition and an index condition.

SLEEP
Triggers on a SLEEP frame. Optionally you can define an NM, a CT, a data pattern condition and an index condition.

LONG
Triggers on a Long frame. Optionally you can define a frame ID condition, an NM, a CT, a DLC extend condition, a data pattern condition and an index condition.

LPOLI
Triggers on a Long POLL frame.

PID
Triggers on a Protected ID field. Optionally, you can define the frame ID pattern.

PTYPe
Triggers on a Protected TYPE field.

PTPID
Triggers on a Protected Type field followed by a PID field. Optionally, you can define the frame ID pattern.

ERRor
Trigger on selected error types. Define the error types with:
`TRIGger<m>:CXPI:ERRor:CRC`
`TRIGger<m>:CXPI:ERRor:DLC`
`TRIGger<m>:CXPI:ERRor:IBS`
`TRIGger<m>:CXPI:ERRor:IFS`
`TRIGger<m>:CXPI:ERRor:LENGth`
`TRIGger<m>:CXPI:ERRor:PARity`
`TRIGger<m>:CXPI:ERRor:UART`

*RST: STARt

17.17.20.3 Decode Results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1383
- `BUS<m>:RESDetail` on page 1384

BUS<m>:CXPI:RESult:FCOunt?....1848
BUS<m>:CXPI:RESult:FRAMe<n>:DATA?....1848
BUS<m>:CXPI:RESult:FRAMe<n>:DLCV?....1848
BUS<m>:CXPI:RESult:FRAMe<n>:STARt?....1849
BUS<m>:CXPI:RESult:FRAMe<n>:STATe?....1849
BUS<m>:CXPI:RESult:FRAMe<n>:STOP?....1849
BUS<m>:CXPI:RESult:FRAMe<n>:TYPE?....1850
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:STATus?....1850
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:TYPE?....1850
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:VALue?....1851

BUS<m>:CXPI:RESult:FCOunt?

Returns the number of frames in the acquisition.

Suffix:

<m> 1..4

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:DATA?

Returns the data bytes of the specified frame.

Suffix:

<m> 1..4

<n> *

Return values:

<Data>

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:DLCV?

Returns the data length code/ extension data length code field value.

Suffix:

<m> 1..4

<n> *

Return values:

<DLC>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:

<m> 1..4

<n> *

Return values:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:STATe?

Returns the state of the specified frame.

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART

*RST: OK

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:STOP?

Returns the stop time of the specified frame.

Suffix:

<m> 1..4

<n> *

Return values:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:TYPE?

Returns the type of the specified frame.

Suffix:

<m> 1..4

<n> *

Return values:

<Type> NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPID | UNKNown | IFS

NPOLI
Normal POLI

LPOLI
Long POLI

PID
Protected ID

PTPID
Protected Type + Protected ID

IFS
Inter-Frame-Space

*RST: NORMal

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:STATus?

Returns the status of the specified field.

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Status> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART

*RST: OK

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:TYPE?

Returns the type (name) of the specified field.

Suffix:

<m> 1..4

<n> *

<o> *

Return values:
<WordType>

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:VALue?

Returns the numeric value of the specified field.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

17.17.20.4 Search Settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to CXPI trigger commands.

SEARch:TRIGger:CXPI:CT....1852
SEARch:TRIGger:CXPI:DATA:DCONdition....1852
SEARch:TRIGger:CXPI:DATA:DMAX....1852
SEARch:TRIGger:CXPI:DATA:DMIN....1853
SEARch:TRIGger:CXPI:DATA:ICONdition....1853
SEARch:TRIGger:CXPI:DATA:IMAX....1853
SEARch:TRIGger:CXPI:DATA:IMIN....1854
SEARch:TRIGger:CXPI:DEXTension:CONDition....1854
SEARch:TRIGger:CXPI:DEXTension:MAX....1854
SEARch:TRIGger:CXPI:DEXTension:MIN....1855
SEARch:TRIGger:CXPI:DLC:CONDition....1855
SEARch:TRIGger:CXPI:DLC:MAX....1855
SEARch:TRIGger:CXPI:DLC:MIN....1855
SEARch:TRIGger:CXPI:ERRor:CRC....1856
SEARch:TRIGger:CXPI:ERRor:DLC....1856
SEARch:TRIGger:CXPI:ERRor:IBS....1856
SEARch:TRIGger:CXPI:ERRor:IFS....1856
SEARch:TRIGger:CXPI:ERRor:LENGth....1857
SEARch:TRIGger:CXPI:ERRor:PARity....1857
SEARch:TRIGger:CXPI:ERRor:UART....1857

SEARch:TRIGger:CXPI:FID:CONDition....................1857
SEARch:TRIGger:CXPI:FID:MAX....................1858
SEARch:TRIGger:CXPI:FID:MIN....................1858
SEARch:TRIGger:CXPI:NM....................1858
SEARch:TRIGger:CXPI:TYPE....................1859

SEARch:TRIGger:CXPI:CT <SearchName>,<CT>
SEARch:TRIGger:CXPI:CT? <SearchName>

Sets the value of the counter (CT), 2 bits indicating the continuity of the frame.

Parameters:
<CT>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:CXPI:DATA:DCONdition? <SearchName>

Sets the condition for the data pattern. You can define no value, an exact value or a value range.

Parameters:

<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
Any data pattern is considered

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:DATA:DMIN` and `SEARch:TRIGger:CXPI:DATA:DCONdition`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:CXPI:DATA:DMAX? <SearchName>

Sets the end value of a data type range if `SEARch:TRIGger:CXPI:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:CXPI:DATA:DMIN? <SearchName>

Specifies a data bit pattern, or sets the start value of a pattern range.

Parameters:
<DataMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:CXPI:DATA:ICONdition? <SearchName>

Set the condition for the data index value. You can define an exact value or a value range.

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DATA:IMIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `SEARch:TRIGger:CXPI:DATA:IMIN` and `SEARch:TRIGger:CXPI:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:CXPI:DATA:IMAX? <SearchName>

Sets the end value of a data index range if `SEARch:TRIGger:CXPI:DATA:ICONdition` is set to `INRange` or `RANGe`.

Parameters:
<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:CXPI:DATA:IMIN? <SearchName>

Specifies a data index bit pattern, or sets the start value of a pattern range.

Parameters:

<DataIndexMin> Range: 1 to 0
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DEXTension:CONDition <SearchName>,<DLCEXTOperator>
SEARch:TRIGger:CXPI:DEXTension:CONDition? <SearchName>

Set the condition for the DLC extend value. You can define an exact value or a value range.

Parameters:

<DLCEXTOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DEXTension:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:DEXTension:MIN` and `SEARch:TRIGger:CXPI:DEXTension:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DEXTension:MAX <SearchName>,<DLCEXTMax>
SEARch:TRIGger:CXPI:DEXTension:MAX? <SearchName>

Sets the end value of a DLC extend range if `SEARch:TRIGger:CXPI:DEXTension:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DLCEXTMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DEXTension:MIN <SearchName>,<DLCEXTMin>
SEARch:TRIGger:CXPI:DEXTension:MIN? <SearchName>

Specifies a data extension bit pattern, or sets the start value of a pattern range.

Parameters:
<DLCEXTMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DLC:CONDition <SearchName>,<DLCOperator>
SEARch:TRIGger:CXPI:DLC:CONDition? <SearchName>

Set the condition for the DLC. You can define an exact value or a value range.

Parameters:

<DLCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one DLC pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DLC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:DLC:MIN` and `SEARch:TRIGger:CXPI:DLC:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DLC:MAX <SearchName>,<DLCMax>
SEARch:TRIGger:CXPI:DLC:MAX? <SearchName>

Sets the end value of a data type range if `SEARch:TRIGger:CXPI:DLC:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DLCMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DLC:MIN <SearchName>,<DLCMin>
SEARch:TRIGger:CXPI:DLC:MIN? <SearchName>

Specifies a DLC pattern, or sets the start value of a pattern range.

Parameters:
<DLCMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:CRC <SearchName>,<ErrorCRC>
SEARch:TRIGger:CXPI:ERRor:CRC? <SearchName>

Searches for a cyclic redundancy check error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorCRC>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:DLC <SearchName>,<ErrorDLC>
SEARch:TRIGger:CXPI:ERRor:DLC? <SearchName>

Searches for a data length code error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorDLC>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:IBS <SearchName>,<ErrorIBS>
SEARch:TRIGger:CXPI:ERRor:IBS? <SearchName>

Searches for an inter-byte-space error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorIBS>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:IFS <SearchName>,<ErrorIFS>
SEARch:TRIGger:CXPI:ERRor:IFS? <SearchName>

Searches for an inter-frame-space error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorIFS>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:LENGth <SearchName>,<ErrorLength>
SEARch:TRIGger:CXPI:ERRor:LENGth? <SearchName>

Searches for a length error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorLength>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:PARity <SearchName>,<ErrorParity>
SEARch:TRIGger:CXPI:ERRor:PARity? <SearchName>

Searches for a parity error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorParity>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:UART <SearchName>,<ErrorUart>
SEARch:TRIGger:CXPI:ERRor:UART? <SearchName>

Searches for an UART error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:

<ErrorUart>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:FID:CONDition <SearchName>,<FrameIDOperator>
SEARch:TRIGger:CXPI:FID:CONDition? <SearchName>

Set the condition for the frame ID. You can define an exact value or a value range.

Parameters:

<FrameIDOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:CXPI:FID:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:FID:MIN` and `SEARch:TRIGger:CXPI:FID:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:FID:MAX <SearchName>,<FrameIDMax>
SEARch:TRIGger:CXPI:FID:MAX? <SearchName>

Sets the end value of a frame ID range if `SEARch:TRIGger:CXPI:FID:CONDition` is set to `INRange` or `OORange`.

Parameters:
<FrameIDMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:FID:MIN <SearchName>,<FrameIDMin>
SEARch:TRIGger:CXPI:FID:MIN? <SearchName>

Specifies a frame ID pattern, or sets the start value of a pattern range.

Parameters:
<FrameIDMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:NM <SearchName>,<NM>
SEARch:TRIGger:CXPI:NM? <SearchName>

Sets the value of the network management (NM) field.

Parameters:
<NM>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:TYPE <SearchName>,<Type>
SEARch:TRIGger:CXPI:TYPE? <SearchName>

Searches for the selected CXPI frame type.

Parameters:

<Type> STARt | NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPD | ERRor

STARt
Triggers on the STARt of the frame.

NORMal
Triggers on a NORMal frame. Optionally you can define a frame ID condition, a NM, a CT, a DLC condition, a data pattern condition and an index condition.

NPOLI
Triggers on a Normal POLL frame. Optionally you can define a frame ID condition, a NM, a CT, a DLC condition, a data pattern condition and an index condition.

SLEEP
Triggers on a SLEEP frame. Optionally you can define a NM, a CT, a data pattern condition and an index condition.

LONG
Triggers on a Long frame. Optionally you can define a frame ID condition, a NM, a CT, a DLC extend condition, a data pattern condition and an index condition.

LPOLI
Triggers on a Long POLL frame.

PID
Triggers on a Protected ID field. Optionally, you can define the frame ID pattern.

PTYPe
Triggers on a Protected TYPE field.

PTPID
Triggers on a PType field followed by a PID field.Optionally, you can define the frame ID pattern.

ERRor
Trigger on selected error types. Define the error types with:
`SEARch:TRIGger:CXPI:ERRor:CRC` on page 1856
`SEARch:TRIGger:CXPI:ERRor:CRC` on page 1856
`SEARch:TRIGger:CXPI:ERRor:DLC` on page 1856
`SEARch:TRIGger:CXPI:ERRor:IBS` on page 1856
`SEARch:TRIGger:CXPI:ERRor:IFS` on page 1856
`SEARch:TRIGger:CXPI:ERRor:LENGth` on page 1857
`SEARch:TRIGger:CXPI:ERRor:PARity` on page 1857
`SEARch:TRIGger:CXPI:ERRor:UART` on page 1857

*RST: STARt

Parameters for setting and query:
<SearchName>

17.17.20.5 Search Result Settings

The search on decoded CXPI data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 17.17.20.3, "Decode Results", on page 1848.

SEARch:RESult:CXPI:FCOunt? 1860
SEARch:RESult:CXPI:FRAMe<m>:DATA? 1860
SEARch:RESult:CXPI:FRAMe<m>:DLCV? 1861
SEARch:RESult:CXPI:FRAMe<m>:STARt? 1861
SEARch:RESult:CXPI:FRAMe<m>:STATe? 1861
SEARch:RESult:CXPI:FRAMe<m>:STOP? 1861
SEARch:RESult:CXPI:FRAMe<m>:TYPE? 1862
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus? 1862
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE? 1862
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue? 1863

SEARch:RESult:CXPI:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:DLCV? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DLC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:STATe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART
*RST: OK

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPID | UNKNown | IFS
*RST: NORMal

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Status> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART
*RST: OK

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordType>

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue? <SearchName>

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

17.18 Mixed Signal Option (MSO, R&S RTE-B1)

This chaptes decribes the remote commands of MSO option R&S RTE-B1.

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.
- Chapter 17.3, "Programming Examples", on page 1018

- Digital Channels....1863
- Parallel Bus Configuration....1866
- Digital Resolution....1873
- Trigger Settings for Digital Signals and Parallel Buses....1874
- MSO Data....1884

17.18.1 Digital Channels

All `DIGital:` commands affect only the settings of the first MSO bus (Bus1). The settings of all other parallel buses (Bus 2, 3,4) remain unchanged.

DIGital<m>:DISPlay....1864
DIGital<m>:TECHnology....1864
DIGital<m>:THReshold....1864
DIGital<m>:THCoupling....1865
DIGital<m>:HYSTeresis....1865
DIGital<m>:LABel....1865
DIGital<m>:DESKew....1866

DIGital<m>:DISPlay <Display>

Enables or disables the indicated digital channel, displays it, and enables the parallel Bus1 if the bus was disabled. That is, `BUS<m>:PARallel:DISPlay:SHDI` and `BUS<m>:PARallel:STATe` are set to ON automatically.

If another active bus already uses the selected digital channel, the instrument disables the other bus to avoid conflicts.

For Bus1, the `DIG:DISP` command has the same effect as `BUS<m>:PARallel:BIT<n>[:STATe]`. To enable digital channels for buses 2, 3 and 4, use the `BUS:PAR:BIT[:STAT]` command.

Suffix:

<m>	0..15 Number of the digital channel

Parameters:

<Display>	ON | OFF

DIGital<m>:TECHnology <Technology>

Selects the threshold voltage for various types of integrated circuits and applies it to all digital channels.

The setting affects only the settings of the first MSO bus (Bus1). You can set the technology value for all buses with `BUS<m>:PARallel:TECHnology`.

Suffix:

<m>	0..15 The suffix is irrelevant.

Parameters:

<Technology>	V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual See `BUS<m>:PARallel:TECHnology`

DIGital<m>:THReshold <Value>

Sets the logical threshold for the channel group to which the indicated digital channel belongs. The setting affects only the settings of the first MSO bus (Bus1).

The setting affects only the settings of the first MSO bus (Bus1). You can set the threshold for all buses with `BUS<m>:PARallel:TECHnology` or `BUS<m>:PARallel:THReshold<n>`.

See also: `DIGital<m>:THCoupling`

Suffix:

<m>	0..15 Number of the digital channel. Channel groups: 0..3; 4..7; 8..11; 12..15

Parameters:

<Value> Range: -8.0 to 8.0
Increment: 200.0e-12
*RST: 0
Default unit: V

DIGital<m>:THCoupling <State>

Sets the threshold and the hysteresis for all digital channels of parallel bus 1 to the same value.

The command `BUS<m>:PARallel:THCoupling` is used to set all buses.

Suffix:

<m> 0..15
The suffix is irrelevant.

Parameters:

<State> ON | OFF

DIGital<m>:HYSTeresis <Hysteresis>

Defines the size of the hysteresis to avoid the change of signal states due to noise for the channel group to which the indicated digital channel belongs.

The setting affects only the settings of the first MSO bus (Bus1). You can set the hysteresis for all buses with `BUS<m>:PARallel:HYSTeresis<n>`.

See also: `DIGital<m>:THCoupling`

Suffix:

<m> 0..15
Number of the digital channel
Channel groups: 0..3; 4..7; 8..11; 12..15

Parameters:

<Hysteresis> MAXIMUM | MAXimum | ROBUST | ROBust | NORMAL | NORMal

MAXIMUM = MAXimum
Maximum value that is possible and useful for the signal and its settings

ROBUST = ROBust
Different hysteresis values for falling and rising edges to avoid an undefined state of the trigger system.

NORMAL = NORMal
The instrument sets a value suitable for the signal and its settings.

DIGital<m>:LABel <Label>

Sets a name for the indicated digital channel. The name is displayed in the diagram.

The setting affects only the settings of the first MSO bus (Bus1). You can set the label for all buses with `BUS<m>:PARallel:BIT<n>:LABel`.

Suffix:

<m>	0..15 Number of the digital channel

Parameters:

<Label>	String containing the channel name

DIGital<m>:DESKew <Deskew>

Sets an individual delay for each digital channel to time-align it with other digital channels. The deskew value compensates delays that are known from the circuit specifics or caused by the different length of cables.

The setting affects only the settings of the first MSO bus (Bus1). You can set the deskew for all buses with `BUS<m>:PARallel:BIT<n>:DESKew`.

Suffix:

<m>	0..15 Number of the digital channel

Parameters:

<Deskew>	Range: -200.0E-09 to 200.0E-09 Increment: 200.0E-12 *RST: 0 Default unit: s

17.18.2 Parallel Bus Configuration

The following commands configure the four parallel buses of R&S RTE-B1.

BUS<m>:PARallel:STATe....1867
BUS<m>:PARallel:BIT<n>[:STATe]....1867
BUS<m>:PARallel:THReshold<n>....1867
BUS<m>:PARallel:TECHnology....1868
BUS<m>:PARallel:THCoupling....1868
BUS<m>:PARallel:HYSTeresis<n>....1869
BUS<m>:PARallel:BIT<n>:DESKew....1869
BUS<m>:PARallel:DESoffset....1870
BUS<m>:PARallel:BIT<n>:LABel....1870
BUS<m>:PARallel:DATA:FORMat....1870
BUS<m>:PARallel:DISPlay:SHDI....1871
BUS<m>:PARallel:DISPlay:SHBU....1871
BUS<m>:PARallel:DISPlay:BTYP....1871
BUS<m>:PARallel:CLON....1872
BUS<m>:PARallel:CLOCk....1872
BUS<m>:PARallel:CLSLope....1872
BUS<m>:PARallel:CLEar....1872

BUS<m>:PARallel:DECTable<n>:SHOW..1873
BUS<m>:PARallel:DECTable<n>:COUNt?..1873
BUS<m>:PARallel:DECTable<n>:DATA?...1873

BUS<m>:PARallel:STATe <Enable>

Enables or disables the indicated parallel bus. The threshold settings of the bus take effect for all *active* parallel buses.

Dependencies: At least one digital channel must be enabled for the selected bus, otherwise the command does not work. The bus is enabled automatically if the first digital channel is enabled with `BUS<m>:PARallel:BIT<n>[:STATe]` or `DIGital<m>:DISPlay`.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<Enable> ON | OFF
*RST: OFF

BUS<m>:PARallel:BIT<n>[:STATe] <Assigned>

Assigns the selected digital channel to the indicated bus, displays it, and enables the bus if the bus was disabled. That is, `BUS<m>:PARallel:DISPlay:SHDI` and `BUS<m>:PARallel:STATe` are set to ON automatically.

If another active bus already uses the selected digital channel, the instrument disables the other bus to avoid conflicts.

For parallel bus 1, the `BUS:PAR:BIT[:STATe]` command has the same effect as `DIGital<m>:DISPlay`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 0..15
Selects the bit of the bus word. Each bit corresponds to a digital channel.

Parameters:

<Assigned> ON | OFF
*RST: OFF

BUS<m>:PARallel:THReshold<n> <Threshold>

Sets the logical threshold for the indicated channel group.

Alternatively, you can set the threshold with `BUS<m>:PARallel:TECHnology`. For the parallel bus 1, you can also use `DIGital<m>:THReshold`.

See also: DIGital<m>:THCoupling

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 1..4
Selects the channel group:
1 = dig. channels 0..3;
2 = dig. channels 4..7
3 = dig. channels 8..11
4 = dig. channels 12..15

Parameters:

<Threshold>
Range: -8.0 to 8.0
Increment: 200.0e-12
*RST: 0
Default unit: V

BUS<m>:PARallel:TECHnology <Technology>

Selects the threshold voltage for various types of integrated circuits and applies it to all digital channels.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15: TTL
V25: CMOS 5.0 V
V165: CMOS 3.3 V
V125: CMOS 2.5 V
V09: CMOS 1.85 V
VM13: ECL, -1.3 V
V38: PECL
V20: LVPECL
V0: Ground
MANual: Set a user-defined threshold value with DIGital<m>:THReshold

*RST: V165

BUS<m>:PARallel:THCoupling <LevelCoupling>

Sets the threshold for all digital channels of the selected bus to the same value. Also the hysteresis value is applied to all digital channels.

Tor parallel bus 1, the command DIGital<m>:THCoupling has the same effect.

Suffix:

<m> 1..4
The suffix is irrelevant.

Parameters:

<LevelCoupling> ON | OFF

*RST: ON

BUS<m>:PARallel:HYSTeresis<n> <Hysteresis>

Defines the size of the hysteresis for the channel group to avoid the change of signal states due to noise.

For the parallel bus 1, you can also use `DIGital<m>:HYSTeresis`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 1..4
Selects the channel group:
1 = dig. channels 0..3;
2 = dig. channels 4..7
3 = dig. channels 8..11
4 = dig. channels 12..15

Parameters:

<Hysteresis> MAXIMUM | MAXimum | ROBUST | ROBust | NORMAL | NORMal

MAXIMUM = MAXimum
Maximum value that is possible and useful for the signal and its settings

ROBUST = ROBust
Different hysteresis values for falling and rising edges to avoid an undefined state of the trigger system.

NORMAL = NORMal
The instrument sets a value suitable for the signal and its settings.

BUS<m>:PARallel:BIT<n>:DESKew <Deskew>

Sets an individual delay for each digital channel to time-align it with other digital channels. The deskew value compensates delays that are known from the circuit specifics or caused by the different length of cables.

For the parallel bus 1, you can also use `DIGital<m>:DESKew`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n>	0..15 Selects the bit of the bus word. Each bit corresponds to a digital channel.

Parameters:

<Deskew>	Range: -200E-9 to 200E-9 Increment: 200E-12 *RST: 0 Default unit: s

---

BUS<m>:PARallel:DESoffset <DeskewOffset>

Sets a general delay for all digital channels. The resulting deskew of a digital channel is the sum of this general value and the individual value set with `BUS<m>:PARallel:BIT<n>:DESKew`.

Suffix:

<m>	1..4 Selects the parallel bus.

Parameters:

<DeskewOffset>	Range: -200E-9 to 200E-9 Increment: 200E-12 *RST: 0 Default unit: s

---

BUS<m>:PARallel:BIT<n>:LABel <Label>

Sets a name for the indicated digital channel. The name is displayed in the diagram.

For the parallel bus 1, you can also use `DIGital<m>:LABel`.

Suffix:

<m>	1..4 Selects the parallel bus.
<n>	0..15 Selects the bit of the bus word. Each bit corresponds to a digital channel.

Parameters:

<Label>	String containing the channel name

BUS<m>:PARallel:DATA:FORMat <DataFormat>

Sets the number format for decoded data values on the display for the indicated parallel bus.

It sets also the number representation for remote data transfer with `BUS<m>:PARallel:DATA[:VALues]?`.

Suffix:

<m>	1..4

Parameters:

<DataFormat> HEX | OCT | BIN | ASCII | ASCii | SIGN | USIG

*RST: HEX

BUS<m>:PARallel:DISPlay:SHDI <ShwDigSigns>

If enabled, the selected digital signals are shown in the diagram. Each channel is displayed as a logic signal.

See also: `DIGital<m>:DISPlay`

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<ShwDigSigns> ON | OFF

*RST: OFF

BUS<m>:PARallel:DISPlay:SHBU <ShowBus>

Shows or hides the indicated parallel bus. If enabled, the resulting bus signal and bus values are displayed in the diagram.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<ShowBus> ON | OFF

*RST: OFF

BUS<m>:PARallel:DISPlay:BTYP <BusRepres>

Selects the display type of the indicated parallel bus.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<BusRepres> COMB | ANALog

COMB
Displays the decoded bus signal with bus values.

ANALog
Displays the bus value as amplitude, similar to an analog waveform.

*RST: COMB

BUS<m>:PARallel:CLON <Clocked>

Defines if the bus is a clocked bus - one of the digital channels serves as clock of the bus.

Suffix:

<m>	1 | 2 Selects the parallel bus. The clocked bus is available on parallel buses 1 and 2.

Parameters:

<Clocked>	ON | OFF *RST: OFF

BUS<m>:PARallel:CLOCk <ClockSource>

Selects the digital channel used as clock of the indicated parallel bus.

Suffix:

<m>	1 | 2 Selects the parallel bus. The clocked bus is available on parallel buses 1 and 2.

Parameters:

<ClockSource>	D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 Clock channel *RST: D1

BUS<m>:PARallel:CLSLope <ClockSlope>

Selects the slope of the clock signal at which all digital channels of the bus are analyzed.

Suffix:

<m>	1 | 2 Selects the parallel bus. The clocked bus is available on parallel buses 1 and 2.

Parameters:

<ClockSlope>	POSitive | NEGative | EITHer *RST: POSitive

BUS<m>:PARallel:CLEar

Removes all assigned digital channels from the bus

Suffix:

<m>	1..4 Selects the parallel bus.

Usage: Event

BUS<m>:PARallel:DECTable<n>:SHOW <ShowDecodeTable>

Available only if `BUS<m>:PARallel:CLON` is set to `ON`.

If enabled, a results box opens with decoded values of the bus signal and its time. You can query the number of values (`BUS<m>:PARallel:DECTable<n>:COUNt?`) and their data value (`BUS<m>:PARallel:DECTable<n>:DATA?`).

Suffix:

<m>	1..4
<n>	1..5

Parameters:

<ShowDecodeTable>	ON | OFF *RST: OFF

BUS<m>:PARallel:DECTable<n>:COUNt?

Returns the number of clock edges in the decode table.

Suffix:

<m>	1..4
<n>	1..5

Usage: Query only

BUS<m>:PARallel:DECTable<n>:DATA?

Returns the data of the decoded values of the bus signal and its time.

Suffix:

<m>	1..4
<n>	1..5

Usage: Query only

17.18.3 Digital Resolution

ACQuire:DRESolution?..1873
ACQuire:POINts:DVALue?..1874

ACQuire:DRESolution?

Returns the current digital resolution of the digital channels.

Return values:

<DigRes>	Default unit: s

Usage: Query only
Asynchronous command

ACQuire:POINts:DVALue?

Returns the current digital record length used by each digital channel.

Return values:

<DigReclgth>	Range:	1000 to 200E6
	Increment:	2
	*RST:	1000
	Default unit:	Sa

Usage: Query only
Asynchronous command

17.18.4 Trigger Settings for Digital Signals and Parallel Buses

In all `TRIGger<m>:PARallel` commands, the trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on digital signals and parallel buses.

- General Commands....1874
- Edge Trigger....1876
- Width Trigger....1877
- Timeout Trigger....1878
- Data2Clock Trigger....1879
- State Trigger....1880
- Pattern Trigger....1881
- Serial Pattern Trigger....1883

17.18.4.1 General Commands

TRIGger<m>:SOURce[:SELect]....1874
TRIGger<m>:PARallel:TYPE....1876
TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue]....1876
TRIGger<m>:PARallel:STATe:CSOurce:VALue....1876
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]....1876
TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]....1876

TRIGger<m>:SOURce[:SELect] <SourceDetailed>

Selects the source of the trigger signal.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
Available values depend on the selected trigger source. For input channels CHAN1...4, a trigger sequence can be configured.
For all other trigger sources, only suffix 1 is allowed.
See also: `TRIGger<m>:SEQuence:MODE`

Parameters:

<SourceDetailed> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4 | EXTernanalog | SBUS | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | LOGIC | MSOB1 | MSOB2 | MSOB3 | MSOB4 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | LINE

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channels

EXTernanalog
External analog signal connected to the External Trigger Input. For this source, only the analog edge trigger is available.

LINE
The instrument generates the trigger from the AC power input and synchronizes the signal to the AC power frequency.

SBUS
Serial bus

D0...D15
Digital channels (option R&S RTE-B1)
See also: Chapter 17.18.4, "Trigger Settings for Digital Signals and Parallel Buses", on page 1874

LOGIc
Logic combination of digital channels, used as trigger source (option R&S RTE-B1)

MSOB1 | MSOB2 | MSOB3 | MSOB4
Parallel bus (option R&S RTE-B1)

Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4
Input channels of R&S RT-ZVCmulti-channel power probe. Only available in the A-trigger with trigger type EDGE.

DIFF1 | DIFF2 | COMMON1 | COMMON2
Differential signals

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:PARallel:TYPE <Type>

Selects the trigger type to trigger on digital channels and parallel buses.

To trigger on analog channels or the external trigger input, use `TRIGger<m>:TYPE`.

Parameters:

<Type> EDGE | WIDTh | TIMeout | DATatoclock | STATe | PATTern | SERPattern

*RST: EDGE

Usage: Asynchronous command

TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue] <ClockSource>
TRIGger<m>:PARallel:STATe:CSOurce:VALue <ClockSource>
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue] <ClockSource>

Selects the digital channel of the clock signal.

Parameters:

<ClockSource> D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: D0

Usage: Asynchronous command

TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine] <LogicalExpr>

Defines a logical combination of several digital channels as trigger condition if `TRIGger<m>:SOURce[:SELect]` is set to LOGIC.

Parameters:

<LogicalExpr> String with logical expression

Example: `TRIGger:PARallel:EDGE:EXPRession 'D1 and D2'`

Usage: Asynchronous command

17.18.4.2 Edge Trigger

See also:

- `TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]` on page 1876

TRIGger<m>:PARallel:EDGE:SLOPe....................1877

TRIGger<m>:PARallel:EDGE:SLOPe <Slope>

Defines the edge - the state transition - of the signal to trigger on a single digital channel (a logic bit), or a logical combination of digital channels.

Parameters:

<Slope>	POSitive | NEGative | EITHer *RST: POSitive
Usage:	Asynchronous command

17.18.4.3 Width Trigger

See also:

- `TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]` on page 1876

TRIGger<m>:PARallel:WIDTh:POLarity....1877
TRIGger<m>:PARallel:WIDTh:RANGe....1877
TRIGger<m>:PARallel:WIDTh:WIDTh....1878
TRIGger<m>:PARallel:WIDTh:DELTa....1878

TRIGger<m>:PARallel:WIDTh:POLarity <Polarity>

Sets the polarity of a pulse. When triggering on a positive pulse, the trigger event occurs on the high to low transition of the pulse if the timing condition is true. When triggering on a negative pulse, the trigger event occurs on the low to high transition of the pulse if the timing condition is true.

Parameters:

<Polarity>	POSitive | NEGative | EITHer *RST: POSitive
Usage:	Asynchronous command

TRIGger<m>:PARallel:WIDTh:RANGe <RangeMode>

Selects how the range of a pulse width is defined:

Parameters:

<RangeMode>	WITHin | OUTSide | SHORter | LONGer **WITHin | OUTSide** Pulses inside or outside a given time range. The time range *Width ± Delta* is defined by `TRIGger<m>:PARallel:WIDTh:WIDTh` and `TRIGger<m>:PARallel:WIDTh:DELTa`. **SHORter | LONGer** Pulses shorter or longer than a given width defined by `TRIGger<m>:PARallel:WIDTh:WIDTh` *RST: WITHin
Usage:	Asynchronous command

TRIGger<m>:PARallel:WIDTh:WIDTh <Width>

Sets the limit for the pulse width.

The effect depends on `TRIGger<m>:PARallel:WIDTh:RANGe`.

- For the ranges SHORter and LONGer, the width defines the maximum and minimum pulse width, respectively.
- For the ranges WITHin and OUTSide, the width defines the center of a range which is defined by the limits `TRIGger<m>:PARallel:WIDTh:DELTa`.

Parameters:

<Width> Range: 200E-12 to 10000
Increment: 200E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:WIDTh:DELTa <WidthDelta>

Defines a range around the given width value. the setting is relevant if `TRIGger<m>:PARallel:WIDTh:RANGe` is set to WITHin or OUTSide. The width is set with `TRIGger<m>:PARallel:WIDTh:WIDTh`.

Parameters:

<WidthDelta> Range: 0 to 432
Increment: 600E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

17.18.4.4 Timeout Trigger

See also:

- `TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 1876

TRIGger<m>:PARallel:TIMeout:RANGe.. 1878
TRIGger<m>:PARallel:TIMeout:TIME..1879

TRIGger<m>:PARallel:TIMeout:RANGe <TimeoutMode>

Sets the state condition.

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH
The level of a digital channel stays above the threshold, or the logical expression for LOGic trigger source is true.

LOW
The level of a digital channel stays below the threshold, or the logical expression for LOGic trigger source is false.

EITHer
The signal state remains unchanged.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:PARallel:TIMeout:TIME <Time>

Defines the time limit for the timeout at which the instrument triggers.

Parameters:

<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

17.18.4.5 Data2Clock Trigger

See also:

- TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue] on page 1876

TRIGger<m>:PARallel:DATatoclock:CSOurce:EDGE 1879
TRIGger<m>:PARallel:DATatoclock:STIMe 1879
TRIGger<m>:PARallel:DATatoclock:HTIMe 1880

TRIGger<m>:PARallel:DATatoclock:CSOurce:EDGE <ClockEdge>

Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time reference point for the setup and hold time measurement.

Parameters:

<ClockEdge> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:DATatoclock:STIMe <SetupTime>

Sets the minimum time *before* the clock edge while data should be stable and not change its state.

See also: "Setup time" on page 884

Parameters:

<SetupTime>	Range: -99.8E-9 to 100E-9 Increment: 1E-9 *RST: 0 Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:DATatoclock:HTIMe <HoldTime>

Sets the minimum time *after* the clock edge while data should be stable and not change its state.

See also: "Hold time" on page 884

Parameters:

<HoldTime>	Range: -99.8E-9 to 100E-9 Increment: 1E-9 *RST: 0 Default unit: s

Usage: Asynchronous command

17.18.4.6 State Trigger

See also:

- `TRIGger<m>:PARallel:STATe:CSOurce:VALue` on page 1876
- `TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 1876

TRIGger<m>:PARallel:STATe:CSOurce:EDGE..1880
TRIGger<m>:PARallel:STATe:BIT<0..15>...1880

TRIGger<m>:PARallel:STATe:CSOurce:EDGE <Slope>

Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time at which the logical states and the bus value are analyzed.

Parameters:

<Slope>	POSitive | NEGative | EITHer *RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:STATe:BIT<0..15> <Bit>

Sets the required state for each digital channel that is used in the bus.

Parameters:

<Bit>	HIGH | LOW | DONTCARE | DONTcare Bit value: 1 (HIGH), 0 (LOW), or X (DONTCARE = DONTcare)

Usage: Asynchronous command

17.18.4.7 Pattern Trigger

TRIGger<m>:PARallel:PATTern:BIT<0..15> <Bit>

Sets the required state for each digital channel that is used in the bus.

Parameters:

<Bit> HIGH | LOW | DONTCARE | DONTcare

Bit value: 1 (HIGH), 0 (LOW), or X (DONTCARE = DONTcare)

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:MODE <Mode>

Sets the mode of the timing condition.

Parameters:

<Mode> OFF | TIMeout | WIDTh

OFF
No timing condition, only the logical pattern condition is relevant.

TIMeout
Defines a minimum time qualification to avoid triggering on unstable or transitional conditions. Use `TRIGger<m>:PARallel:PATTern:TIMeout:MODE` and `TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]` to specify the timeout.

WIDTh
Sets a pulse width as timing condition. The pulse starts when the pattern comes true, and the trigger event occurs when the pattern comes false during the specified time limit. Use `TRIGger<m>:PARallel:PATTern:WIDTh:RANGe`, `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]`, and `TRIGger<m>:PARallel:PATTern:WIDTh:DELTa` to specify the width.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:TIMeout:MODE <TimeoutMode>

Sets the state condition for the timeout qualification if `TRIGger<m>:PARallel:PATTern:MODE` is set to TIMeout. To set the time limit, use `TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]`.

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH: The pattern stays true for the specified time.
LOW: The pattern stays false for the specified time.
EITHER: The pattern remains unchanged for the specified time.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:TIMeout[:TIME] <Time>

Defines the time limit for the timeout at which the instrument triggers.

Parameters:

<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:WIDTh:RANGe <WidthRangeMode>

Selects how the range of a pulse width is defined if `TRIGger<m>:PARallel:PATTern:MODE` is set to WIDTh.

Parameters:

<WidthRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers when the pattern comes false inside a given time range. The time limit is defined by `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]` and `TRIGger<m>:PARallel:PATTern:WIDTh:DELTa` (*Width ± Delta*).

OUTSide
Triggers when the pattern comes false before or after the given time range. The time limit definition is the same as for WITHin range.

SHORter | LONGer
Triggers when the pattern comes false before or after the given width has expired. Width is set with `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]`.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh] <Width>

The effect depend on the setting of the `TRIGger<m>:PARallel:PATTern:WIDTh:RANGe`command.

For the ranges SHOrter and LONGer, the width defines the maximum and minimum time limit, respectively.

For the ranges WITHin and OUTSide, the width defines the center of a range which is defined by the limits "±Delta".

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:WIDTh:DELTa <WidthDelta>

Defines a range around the width value set with `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]`.

Parameters:
<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

17.18.4.8 Serial Pattern Trigger

See also:
- `TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]` on page 1876
- `TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 1876

TRIGger<m>:PARallel:SPATtern:CSOurce:EDGE <ClockEdge>

Sets the edge of the clock signal. The bit value is determined at the crossing of the clock edge and the logical threshold.

Parameters:
<ClockEdge> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:SPATtern:PATTern <Pattern>

Defines the serial bit string on which to trigger.

Parameters:
<Pattern> Numeric or string pattern, see Chapter 17.4.6, "Bit Pattern Parameter", on page 1039. The bit value X (don't care) is not allowed.

Usage: Asynchronous command

17.18.5 MSO Data

To export data of digital channels and parallel buses to file, use the following commands:

- EXPort:WAVeform:SOURce on page 1363
- EXPort:WAVeform:NAME on page 1364
- EXPort:WAVeform:SAVE on page 1365

The remote data transfer from the instrument to the controlling computer is performed using the following commands:

DIGital<m>:DATA:HEADer?... 1884
DIGital<m>:DATA[:VALues]?... 1884
BUS<m>:PARallel:DATA:FORMat...1885
BUS<m>:PARallel:DATA:HEADer?.. 1885
BUS<m>:PARallel:DATA[:VALues]?.. 1885

DIGital<m>:DATA:HEADer?

Returns the header of digital channel data

Table 17-13: Header data

Position	Meaning	Example
1	XStart, acquisition time before trigger, in s	-5E-008 = - 50 ns
2	XStop, acquisition time after trigger, in s	5E-008 = 50 ns
3	Record length of the waveform in Samples	1000
4	Number of values per sample interval. For digital data the result is 1.	1

Suffix:
<m> 0..15
Number of the digital channel

Usage: Query only

DIGital<m>:DATA[:VALues]?

Returns the data of the indicated digital channel for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Suffix:
<m> 0..15
Selects the digital channel.

Return values:
<Data> List of data (0 and 1)

Usage: Query only

BUS<m>:PARallel:DATA:FORMat <DataFormat>

Sets the number format for decoded data values on the display for the indicated parallel bus.

It sets also the number representation for remote data transfer with `BUS<m>:PARallel:DATA[:VALues]?`.

Suffix:

<m> 1..4

Parameters:

<DataFormat> HEX | OCT | BIN | ASCII | ASCii | SIGN | USIG

*RST: HEX

BUS<m>:PARallel:DATA:HEADer?

Returns the header data of the indicated bus.

For a detailed description, see `DIGital<m>:DATA:HEADer?`.

Suffix:

<m> 1..4
Selects the parallel bus.

Usage: Query only

Firmware/Software: V 2.40

BUS<m>:PARallel:DATA[:VALues]?

Returns the data of the indicated parallel bus.

Requirements:

- `BUS<m>:PARallel:STATe` is set to `ON`.
- `BUS<m>:PARallel:DISPlay:SHBU` is set to `ON`.
- Number format is set with `BUS<m>:PARallel:DATA:FORMat`.

Suffix:

<m> 1..4
Selects the parallel bus.

Return values:

<Data> List of values according to the format setting.

Example:

```
BUS:PAR:STAT ON
BUS:PAR:DISP:SHBU ON
BUS:PAR:DISP:BTYP COMB
BUS:PAR:DATA:FORMat HEX
BUS:PAR:DATA:VAL?
```

Usage: Query only

17.19 Waveform Generator (Option R&S RTE-B6)

17.19.1 Waveform Generator Setup

17.19.1.1 General

WGENerator<m>:ACOPy....1886
WGENerator<m>[:ENABle]....1886
WGENerator<m>:SOURce....1886
WGENerator<m>:PRESet....1886

WGENerator<m>:ACOPy

Copies all settings from Gen1/Gen2 and applies them to Gen2/Gen1.

Suffix:
<m> 1..2

Usage: Event

WGENerator<m>[:ENABle] <State>

Enables the waveform generator and outputs the waveform.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF

Usage: Asynchronous command

WGENerator<m>:SOURce <OperationMode>

Sets the operation mode for the the waveform generator.

Suffix:
<m> 1..2

Parameters:
<OperationMode> FUNCgen | MODulation | SWEep | ARBGenerator
*RST: FUNCgen

WGENerator<m>:PRESet

Sets the parameters of the waveform generator to their default values.

Suffix:
<m> 1..2

Usage: Event

17.19.1.2 Function Generator

WGENerator<m>:FUNCtion[:SELect] 1887
WGENerator<m>:FREQuency 1887
WGENerator<m>:PERiod 1887
WGENerator<m>:FUNCtion:PULSe[:WIDTh] 1888
WGENerator<m>:FUNCtion:RAMP[:SYMMetry] 1888
WGENerator<m>:FUNCtion:SQUare:DCYCle 1888

WGENerator<m>:FUNCtion[:SELect] <FunctionType>

Sets the type of waveform to be generated for the function generator.

Suffix:
<m> 1..2

Parameters:
<FunctionType> SINusoid | SQUare | RAMP | DC | PULSe | SINC | CARDiac | GAUSs | LORNtz | EXPRise | EXPFall

SINC: Cardial sine

*RST: SINusoid

WGENerator<m>:FREQuency <Frequency>

Sets the frequency of the waveform. The available frequency range depends on the selected function type.

Suffix:
<m> 1..2

Parameters:
<Frequency>
Range: 1E-3 to 100E+6
Increment: 1
*RST: 1E+6
Default unit: Hz

WGENerator<m>:PERiod <Period>

Sets the period of the waveform. The available period range depends on the selected function type.

Suffix:
<m> 1..2

Parameters:

<Period>	Range: 8E-9 to 1000 Increment: 1 *RST: 1E-6 Default unit: s

---

WGENerator<m>:FUNCtion:PULSe[:WIDTh] <PulseWidth>

Sets the pulse duration for a pulse waveform.

Suffix:

<m>	1..2

Parameters:

<PulseWidth>	Range: 16.5E-9 to 90E+3 Increment: 1 *RST: 500E-9 Default unit: s

---

WGENerator<m>:FUNCtion:RAMP[:SYMMetry] <RampSymmetry>

Sets the symmetry for a ramp waveform.

Suffix:

<m>	1..2

Parameters:

<RampSymmetry>	Range: 0 to 100 Increment: 1 *RST: 50 Default unit: %

WGENerator<m>:FUNCtion:SQUare:DCYCle <SquareDutyCycle>

Sets the duty cycle for a square waveform.

Suffix:

<m>	1..2

Parameters:

<SquareDutyCycle>	Range: 0.01 to 99.99 Increment: 1 *RST: 50 Default unit: %

17.19.1.3 Modulation

WGENerator<m>:MODulation:TYPE....1889
WGENerator<m>:MODulation:AM:DCYCle....1889
WGENerator<m>:MODulation:AM:DEPTh....1889
WGENerator<m>:MODulation:AM:FREQuency....1890

WGENerator<m>:MODulation:AM:SYMMetry........ 1890
WGENerator<m>:MODulation:CARRier:FREQuency........ 1890
WGENerator<m>:MODulation:CARRier:PERiod........ 1890
WGENerator<m>:MODulation:AM[:FUNCtion]........ 1891
WGENerator<m>:MODulation:FM:DCYCle........ 1891
WGENerator<m>:MODulation:FM:DEViation........ 1891
WGENerator<m>:MODulation:FM:FREQuency........ 1892
WGENerator<m>:MODulation:FM:SYMMetry........ 1892
WGENerator<m>:MODulation:FM[:FUNCtion]........ 1892
WGENerator<m>:MODulation:FSK:FONE........ 1892
WGENerator<m>:MODulation:FSK:FTWO........ 1893
WGENerator<m>:MODulation:FSK[:RATE]........ 1893
WGENerator<m>:MODulation:PWM:DCYCle........ 1893
WGENerator<m>:MODulation:PWM:DEPTh........ 1894
WGENerator<m>:MODulation:PWM:FREQuency........ 1894
WGENerator<m>:MODulation:PWM:SYMMetry........ 1894
WGENerator<m>:MODulation:PWM[:FUNCtion]........ 1894

WGENerator<m>:MODulation:TYPE <ModulationType>

Sets the modulation type.

Suffix:

<m>	1..2

Parameters:

<ModulationType>	AM | FM | PWM | FSK
	*RST: AM

WGENerator<m>:MODulation:AM:DCYCle <SquareDutyCycle>

Sets the duty cycle of a square waveform for amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:

<m>	1..2

Parameters:

<SquareDutyCycle>	Range: 10 to 90
	Increment: 1
	*RST: 50
	Default unit: %

WGENerator<m>:MODulation:AM:DEPTh <Depth>

Sets the amplitude modulation depth.

Suffix:

<m>	1..2

Parameters:

<Depth>
Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:AM:FREQuency <Frequency>

Sets the frequency of the modulation waveform for amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:

<m> 1..2

Parameters:

<Frequency>
Range: 1E-3 to 1E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:AM:SYMMetry <RampSymmetry>

Sets the symmetry, the percentage of time the ramp modulation waveform is rising, for amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:

<m> 1..2

Parameters:

<RampSymmetry>
Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:CARRier:FREQuency <CarrierFreq>

Sets the frequency of the carrier signal for a modulation waveform.

Suffix:

<m> 1..2

Parameters:

<CarrierFreq>
Range: 1E-3 to 100E+6
Increment: 10
*RST: 1E+6
Default unit: Hz

WGENerator<m>:MODulation:CARRier:PERiod <CarrierPeriod>

Sets the period of the carrier signal for a modulation waveform.

Suffix:
<m> 1..2

Parameters:
<CarrierPeriod> Range: 8E-9 to 1000
Increment: 10
*RST: 1E-6
Default unit: s

WGENerator<m>:MODulation:AM[:FUNCtion] <SignalType>

Sets the type of the modulation signal for the amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:
<m> 1..2

Parameters:
<SignalType> SINusoid | SQUare | RAMP
*RST: SINusoid

WGENerator<m>:MODulation:FM:DCYCle <SquareDutyCycle>

Sets the duty cycle of a square waveform for frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<SquareDutyCycle> Range: 10 to 90
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:FM:DEViation <Deviation>

Sets the frequency modulation deviation.

Suffix:
<m> 1..2

Parameters:
<Deviation> Range: 1E-3 to 10E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:FM:FREQuency <Frequency>

Sets the frequency of the modulation waveform for frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<Frequency> Range: 1E-3 to 1E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:FM:SYMMetry <RampSymmetry>

Sets the symmetry, the percentage of time the ramp modulation waveform is rising, for frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<RampSymmetry> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:FM[:FUNCtion] <SignalType>

Sets the type of the modulation signal for the frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<SignalType> SINusoid | SQUare | RAMP
*RST: SINusoid

WGENerator<m>:MODulation:FSK:FONE <Frequency1>

Sets the frequency of the carrier waveform, if `WGENerator<m>:MODulation:TYPE` is set to `FSK`.

Suffix:
<m> 1..2

Parameters:

<Frequency1>	Range: 1E-3 to 100E+6 Increment: 1 *RST: 1E+6 Default unit: Hz

---

WGENerator<m>:MODulation:FSK:FTWO <Frequency2>

Sets the frequency of the modulated waveform, if `WGENerator<m>:MODulation:TYPE` is set to `FSK`.

Suffix:

<m>	1..2

Parameters:

<Frequency2>	Range: 1E-3 to 100E+6 Increment: 1 *RST: 1000 Default unit: Hz

---

WGENerator<m>:MODulation:FSK[:RATE] <Rate>

Sets the hop rate, the time before a switch from the carrier frequency set with `WGENerator<m>:MODulation:FSK:FONE` and the modulation frequency set with `WGENerator<m>:MODulation:FSK:FTWO`.

Suffix:

<m>	1..2

Parameters:

<Rate>	Range: 1E-3 to 1E+6 Increment: 1 *RST: 1000 Default unit: Hz

---

WGENerator<m>:MODulation:PWM:DCYCle <SquareDutyCycle>

Sets the duty cycle of a square waveform for pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:

<m>	1..2

Parameters:

<SquareDutyCycle>	Range: 10 to 90 Increment: 1 *RST: 50 Default unit: %

WGENerator<m>:MODulation:PWM:DEPTh <Depth>

Sets the modulation depth for pulse width modulation.

Suffix:
<m> 1..2

Parameters:
<Depth> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:PWM:FREQuency <Frequency>

Sets the frequency of the modulation waveform for pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:
<m> 1..2

Parameters:
<Frequency> Range: 1E-3 to 1E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:PWM:SYMMetry <RampSymmetry>

Sets the symmetry, the percentage of time the ramp modulation waveform is rising, for pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:
<m> 1..2

Parameters:
<RampSymmetry> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:PWM[:FUNCtion] <SignalType>

Sets the type of the modulation signal for the pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:
<m> 1..2

Parameters:
<SignalType> SINusoid | SQUare | RAMP
*RST: SINusoid

17.19.1.4 Sweep

WGENerator<m>:SWEep:FSTart....................1895
WGENerator<m>:SWEep:TIME....................1895
WGENerator<m>:SWEep[:FEND]....................1895

WGENerator<m>:SWEep:FSTart <StartFrequency>

Sets the start frequency of the sweep range.

Suffix:

<m> 1..2

Parameters:

<StartFrequency> Range: 1E-3 to 100E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:SWEep:TIME <Time>

Sets the duration of the sweep.

Suffix:

<m> 1..2

Parameters:

<Time> Range: 1E-3 to 500
Increment: 1
*RST: 1E-3
Default unit: s

WGENerator<m>:SWEep[:FEND] <StopFrequency>

Sets the stop frequency of the sweep range.

Suffix:

<m> 1..2

Parameters:

<StopFrequency> Range: 1E-3 to 100E+6
Increment: 1
*RST: 1E+6
Default unit: Hz

17.19.1.5 ARB

WGENerator<m>:ARBGen:COPY....................1896
WGENerator<m>:ARBGen:MULTichannel:IMPort....................1896
WGENerator<m>:ARBGen:MULTichannel:NAME....................1896
WGENerator<m>:ARBGen:MULTichannel:OPEN....................1897

WGENerator<m>:ARBGen:NAME ... 1897
WGENerator<m>:ARBGen:OPEN ... 1897
WGENerator<m>:ARBGen:RUNMode ... 1897
WGENerator<m>:ARBGen:SAMPles? ... 1898
WGENerator<m>:ARBGen:SELect ... 1898
WGENerator<m>:ARBGen:SRATe ... 1898
WGENerator<m>:ARBGen[:SOURce] ... 1898

WGENerator<m>:ARBGen:COPY

Loads the waveform from the selected signal source (`WGENerator<m>:ARBGen:SELect`).

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

WGENerator<m>:ARBGen:MULTichannel:IMPort <SavedWfmSrc>

Assigns a waveform from the multichannel file to the arbitrary waveform generator.

Suffix:
<m> 1..2

Parameters:
<SavedWfmSrc> WFM1 | WFM2 | WFM3 | WFM4
*RST: WFM1

Example:
```
WGENerator1:ARBGen:MULTichannel:NAME
ArbMultichannelCurve_2017-02-16_01.bin
WGENerator1:ARBGen:MULTichannel:IMPort WF1
WGENerator1:ARBGen:MULTichannel:OPEN
```

WGENerator<m>:ARBGen:MULTichannel:NAME <FilePath>

Defines the path and the multichannel arbitrary waveform file to be imported. If not path is given, the default path
`C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms`
is used.

Suffix:
<m> 1..2

Parameters:
<FilePath>

Example: See `WGENerator<m>:ARBGen:MULTichannel:IMPort` on page 1896.

WGENerator<m>:ARBGen:MULTichannel:OPEN

Loads the arbitrary waveform.

Suffix:
<m> 1..2

Example: See WGENerator<m>:ARBGen:MULTichannel:IMPort on page 1896.

Usage: Event
Asynchronous command

WGENerator<m>:ARBGen:NAME <FilePath>

Sets the file path and the file for an arbitrary waveform, if WGENerator<m>:ARBGen[:SOURce] on page 1898 is set to ARBitrary.

If not path is given, the default path
C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms
is used.

This command is only relevant for files with one waveform defined. For multichannel arbitrary waveform files, see WGENerator<m>:ARBGen:MULTichannel:OPEN.

Suffix:
<m> 1..2

Parameters:
<FilePath>

WGENerator<m>:ARBGen:OPEN

Loads the arbitrary waveform, that is selected with the WGENerator<m>:ARBGen:NAME command.

This command is only relevant for files with one waveform defined. For multichannel arbitrary waveform files, see WGENerator<m>:ARBGen:MULTichannel:OPEN.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

WGENerator<m>:ARBGen:RUNMode <RunMode>

Sets the duration for which the signal of the arbitrary generator will be output after the trigger event.

Suffix:
<m> 1..2

Parameters:

<RunMode>	SINGle | CONTinuous *RST: CONTinuous

WGENerator<m>:ARBGen:SAMPles?

Displays the number of samples for the arbitrary waveform.

Suffix:

<m> 1..2

Return values:

<NumSamples> Range: 0 to 128000000
Increment: 10
*RST: 0
Default unit: Sa

Usage: Query only

WGENerator<m>:ARBGen:SELect <ScopeSignSrc>

Selects the oscilloscope source, from which the arbitrary signal is loaded, if `WGENerator<m>:ARBGen[:SOURce]` is set to `SCOPe`.

Suffix:

<m> 1..2

Parameters:

<ScopeSignSrc> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | R1 | R2 | R3 | R4
*RST: C1W1

WGENerator<m>:ARBGen:SRATe <SampleRate>

Sets the sample rate for the arbitrary waveform.

Suffix:

<m> 1..2

Parameters:

<SampleRate> Range: 1 to 250E+6
Increment: 10
*RST: 1E+6
Default unit: Sa/s

WGENerator<m>:ARBGen[:SOURce] <WaveformSource>

Sets the arbitrary waveform source.

Suffix:

<m>	1..2

Parameters:

<WaveformSource> ARBitrary | SCOPe

ARBitrary
Enables you to load an existing arbitrary file.

SCOPe
Enables you to load a scope waveform.

*RST: ARBitrary

17.19.1.6 Output

WGENerator<m>:OUTPut[:LOAD]....1899
WGENerator<m>:VOLTage[:VPP]....1899
WGENerator<m>:VOLTage:DCLevel....1900
WGENerator<m>:VOLTage:HIGH....1900
WGENerator<m>:VOLTage:INVersion....1900
WGENerator<m>:VOLTage:LOW....1900
WGENerator<m>:VOLTage:OFFSet....1901

WGENerator<m>:OUTPut[:LOAD] <Load>

Sets the load of the DUT at its connection.

Suffix:

<m>	1..2

Parameters:

<Load> FIFTy | HIZ

HIZ: high input impedance

*RST: HIZ

WGENerator<m>:VOLTage[:VPP] <Amplitude>

Sets the amplitude, peak to peak voltage, of the output waveform. This is defined as the voltage difference between the maximum (`WGENerator<m>:VOLTage:HIGH`) and the minimum (`WGENerator<m>:VOLTage:LOW`) voltage levels.

The value is set for the currently selected load (`WGENerator<m>:OUTPut[:LOAD]`). If the load is changed, the value of the amplitude is adapted to this new setting.

Suffix:

<m>	1..2

Parameters:

<Amplitude>

Range: 0.01 to 12
Increment: 0.1
*RST: 1
Default unit: V

WGENerator<m>:VOLTage:DCLevel <DCLevel>

Sets the DC level of the generated DC signal.

Suffix:

<m>	1..2

Parameters:

<DCLevel>	Range: -5.995 to 5.995 Increment: 0.1 *RST: 0 Default unit: V

WGENerator<m>:VOLTage:HIGH <High>

Sets the high signal level of the output waveform.

Suffix:

<m>	1..2

Parameters:

<High>	Range: -5.99 to 6 Increment: 0.1 *RST: 0.5 Default unit: V

WGENerator<m>:VOLTage:INVersion <Inversion>

Enables inversion, which swaps the two poles of the waveform.

Suffix:

<m>	1..2

Parameters:

<Inversion>	ON | OFF *RST: OFF

WGENerator<m>:VOLTage:LOW <Low>

Sets the low signal level of the output waveform.

Suffix:

<m>	1..2

Parameters:

<Low>	Range: -6 to 5.99 Increment: 0.1 *RST: -0.5 Default unit: V

WGENerator<m>:VOLTage:OFFSet <Offset>

Sets a voltage offset.

Suffix:

<m>	1..2

Parameters:

<Offset>	Range: -5.995 to 5.995 Increment: 0.1 *RST: 0 Default unit: V

17.19.1.7 Noise

WGENerator<m>:MODulation:NDCLevel....1901
WGENerator<m>:MODulation:NLABsolute?....1901
WGENerator<m>:MODulation:NLPCent....1901
WGENerator<m>:MODulation:NOISe....1902

WGENerator<m>:MODulation:NDCLevel <LevelDC>

Sets the DC noise level.

Suffix:

<m>	1..2

Parameters:

<LevelDC>	Range: 0 to 12 Increment: 0.1 *RST: 0.01 Default unit: V

WGENerator<m>:MODulation:NLABsolute?

Queries the level of the noise in volts.

Suffix:

<m>	1..2

Return values:

<LevelAbs>	Range: 0 to 12 Increment: 0.1 *RST: 0 Default unit: V
Usage:	Query only

WGENerator<m>:MODulation:NLPCent <LevelPct>

Sets the noise level in percentage of the amplitude.

Suffix:

<m>	1..2

Parameters:

<LevelPct>
Range: 0 to 100
Increment: 1
*RST: 0
Default unit: %

WGENerator<m>:MODulation:NOISe <State>

Enables the adding of noise to the waveform.

Suffix:

<m>	1..2

Parameters:

<State> ON | OFF
*RST: OFF

17.19.2 Pattern Generator Setup

PGENerator:BITRate....1902
PGENerator:ENABle....1902
PGENerator:FILE:OPEN....1902
PGENerator:FILE[:NAME]....1903
PGENerator:HLEVel....1903
PGENerator:PRESet....1903
PGENerator:RUNMode....1903

PGENerator:BITRate <BitRate>

Sets the number of transmitted bits per second for the pattern generator.

Parameters:

<BitRate>
Range: 1 to 40E+6
Increment: 10
*RST: 1E+6
Default unit: bps

PGENerator:ENABle <State>

Enables the pattern generator and outputs the waveform.

Parameters:

<State> ON | OFF

PGENerator:FILE:OPEN

Loads the specified pattern file to the instrument.

Usage:	Event Asynchronous command

PGENerator:FILE[:NAME] <FilePath>

Specifies a file path to open a pattern generator file.

Parameters:

<FilePath>	String with the name of the file.

PGENerator:HLEVel <HighLevel>

Sets the high level of the pattern generator signal.

Parameters:

<HighLevel>	Range: 1.2 to 5 Increment: 0.1 *RST: 1.2 Default unit: V

PGENerator:PRESet

Sets the default pattern generator settings.

Usage:	Event

PGENerator:RUNMode <RunMode>

Sets the duration for which the signal of the pattern generator will be output after the trigger event.

Parameters:

<RunMode>	SINGle | CONTinuous *RST: CONTinuous

17.19.3 Coupling & Sync

WGENerator<m>:COUPling:ALL 1903
WGENerator<m>:COUPling:AMPLitude 1904
WGENerator<m>:COUPling:OUTPut 1904
WGENerator<m>:COUPling:PHASeshift 1904
WGENerator<m>:COUPling[:FREQuency] 1904
GENerator:SYNC[:COMBination] 1905

WGENerator<m>:COUPling:ALL <CoupleAll>

Enables the coupling of all parameters of the generators, except of load and inversion.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
<CoupleAll> ON | OFF
*RST: OFF

WGENerator<m>:COUPling:AMPLitude <CoupleAmplitude>

Enables the coupling of all amplitude parameters of the generators.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
<CoupleAmplitude> ON | OFF
*RST: OFF

WGENerator<m>:COUPling:OUTPut CoupleAllOutput

Couples all parameters including output enable, but excluding inversion and load.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
CplAllWthEnab ON | OFF
*RST: OFF

WGENerator<m>:COUPling:PHASeshift <PhaseShift>

Sets the phase shift between the waveform of Gen1 and Gen2 when the frequency parameters of the two waveforms are coupled.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
<PhaseShift> Range: -180 to 180
Increment: 1
*RST: 0
Default unit: °

WGENerator<m>:COUPling[:FREQuency] <CoupleFrequency>

Enables the coupling of all frequency parameters of the generators.

Suffix:

<m>	1..2 Specifies the master generator.

Parameters:

<CoupleFrequency>	ON | OFF *RST: OFF

GENerator:SYNC[:COMBination] <Combination>

Sets which signals generated from the waveform generator are synchronized.

Parameters:

<Combination>	NONE | GEN12 | G1PG | G2PG | G12PG GEN12: Gen1 and Gen 2 GEN1PG/GEN2PG: Gen1/Gen 2 and Patt Gen GEN12PG: Gen1,Gen 2 and Patt Gen *RST: NONE

17.19.4 DC Offset Alignment

GENerator:ALIGnment:DC:ABORt..1905
GENerator:ALIGnment:DC:RESult:DATE?..1905
GENerator:ALIGnment:DC:RESult:TIME?..1905
GENerator:ALIGnment:DC:RESult[:STATe]?..1906
GENerator:ALIGnment:DC[:STARt]..1906

GENerator:ALIGnment:DC:ABORt

Aborts a DC offset alignment.

Usage: Event
Asynchronous command

GENerator:ALIGnment:DC:RESult:DATE?

Queries the date of the last performed DC offset alignment.

Return values:
<Date>

Usage: Query only

GENerator:ALIGnment:DC:RESult:TIME?

Queries the time of the last performed DC offset alignment.

Return values:
<Time>

Usage: Query only

GENerator:ALIGnment:DC:RESult[:STATe]?

Queries the result of a DC offset alignment.

Return values:

<State> PASSed | FAILed | NOALigndata

*RST: NOALigndata

Usage: Query only

GENerator:ALIGnment:DC[:STARt]

Starts a DC offset alignment.

Usage: Event
Asynchronous command

17.20 Power Analysis (Option R&S RTE-K31)

- General...1906
- Deskew...1907
- Report...1908
- Power Quality...1914
- Inrush Current...1916
- Current Harmonic...1918
- Modulation Analysis...1920
- Dynamic ON Resistance...1922
- Slew Rate...1923
- S.O.A...1926
- Turn On/Off...1928
- Switching Loss...1930
- Power Efficiency...1933
- Ripple...1934
- Transient Response...1941
- Spectrum...1944

17.20.1 General

POWer:ENABle...1907
POWer:SOURce:CURRent<1..2>...1907
POWer:SOURce:VOLTage<1..4>...1907

POWer:ENABle

Activates the power mode and initializes the power measurements. If the power mode is disabled, the instrument does not accept any `POWer` command.

Use `POWer:ENABle` after each `*RST`.

Example: See Chapter 17.3.7.1, "Auto Deskew", on page 1033

Usage: Event

POWer:SOURce:CURRent<1..2> <CurrentSource>

Sets the channel for the current source.

Parameters:

<CurrentSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4

*RST: CURRent1: CHAN2, CURRent2: CHAN4,

Usage: Asynchronous command

POWer:SOURce:VOLTage<1..4> <VoltageSource>

Sets the channel for the voltage source input.

Parameters:

<VoltageSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4

*RST: VOLTage1: CHAN1, VOLTage2: CHAN3, VOLTage3: CHAN3, VOLTage4: CHAN4

Usage: Asynchronous command

17.20.2 Deskew

Programming example: Chapter 17.3.7.1, "Auto Deskew", on page 1033

POWer:DESKew:CURRent....1907
POWer:DESKew:EXECute....1908
POWer:DESKew:RESet....1908
POWer:DESKew:TIME?....1908
POWer:DESKew:UDPReset....1908

POWer:DESKew:CURRent

Applies the result of the auto deskew to the "Skew offset" value.

Usage: Event

POWer:DESKew:EXECute

Starts the auto deskew.

Usage: Event

POWer:DESKew:RESet <OverwriteCurrSkew>

Overwrites the present skew setup.

Parameters:

<OverwriteCurrSkew> ON | OFF

*RST: ON

POWer:DESKew:TIME?

Queries the result of the auto deskew.

Return values:

<AutoDeskewOffs> Range: -100E-9 to 100E-9

*RST: 0

Default unit: s

Usage: Query only

POWer:DESKew:UDPReset <UsrDefinedPreset>

Activates or deactivates a user defined setup. If ON, the instrument setup including probe setup and the deskew values are written to a user defined preset file (saveset) that can be loaded using `MMEMory:RCL` on page 1359.

The default path is:

`C:\Users\Public\Documents\Rohde-Schwarz\RTE\SaveSets\`

Parameters:

<UsrDefinedPreset> ON | OFF

*RST: ON

17.20.3 Report

POWer:REPort:CONTent:HSETup 1909
POWer:REPort:CONTent:MSETup 1909
POWer:REPort:CONTent:MSIGnal 1909
POWer:REPort:CONTent:RESU 1909
POWer:REPort:CONTent:SETTings 1909
POWer:REPort:CONTent:TSETup 1909
POWer:REPort:CONTent:VSETup 1909

POWer:REPort:CONTent:TITLe....................1909
POWer:REPort:DESCription....................1910
POWer:REPort:DUT....................1910
POWer:REPort:SITE....................1910
POWer:REPort:TEMPerature....................1910
POWer:REPort:USER....................1910
POWer:REPort:FONT:COLO....................1910
POWer:REPort:FONT:FAMI....................1910
POWer:REPort:FONT:SIZE....................1910
POWer:REPort:LOGO....................1910
POWer:REPort:PAPersize....................1910
POWer:REPort:FILE:NAME....................1911
POWer:REPort:FILE:DELete....................1911
POWer:REPort:FILE:NEW....................1911
POWer:REPort:FILE:SAVE....................1911
POWer:REPort:TEST:ADD....................1911
POWer:REPort:TEST:INSert....................1911
POWer:REPort:TEST:REMove....................1911
POWer:REPort:INVert....................1912
POWer:REPort:TEST:DSEA....................1912
POWer:REPort:TEST:ISE....................1912
POWer:REPort:TEST:SEA....................1912
POWer:REPort:TEST:RSE....................1912
POWer:REPort:TEST:DIRectory....................1913
POWer:REPort:TEST:COMMent....................1913
POWer:REPort:TEST:COUNt....................1913
POWer:REPort:TEST:LSENd?....................1914

POWer:REPort:CONTent:HSETup <ContentHorizSet>
POWer:REPort:CONTent:MSETup <ContentMeasSet>
POWer:REPort:CONTent:MSIGnal <ContentMeasuredSigns>
POWer:REPort:CONTent:RESU <ContentResults>
POWer:REPort:CONTent:SETTings <ContentSettings>
POWer:REPort:CONTent:TSETup <ContentTrigSet>
POWer:REPort:CONTent:VSETup <ContentVertSet>

Sets how often the respective content is shown in the final report.

Parameters:

<ContentVertSet> ALWAYS | NEVER | ONCE

*RST: ONCE

POWer:REPort:CONTent:TITLe <ContentTitle>

Includes the title page in the report.

Parameters:

<ContentTitle> ON | OFF

*RST: ON

POWer:REPort:DESCription <String>
POWer:REPort:DUT <String>
POWer:REPort:SITE <String>
POWer:REPort:TEMPerature <String>
POWer:REPort:USER <String>

The content of the strings is shown at the title page of a report if the title page is included in the report.

Parameters:
<String>

POWer:REPort:FONT:COLO <FontColor>

Sets the font color.

Parameters:

<FontColor>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

POWer:REPort:FONT:FAMI <FontFamily>

Selects the font family. You can choose between Arial and Helvetica.

Parameters:

<FontFamily>	ARIAL | HELV	
	*RST:	ARIAL

POWer:REPort:FONT:SIZE <FontSize>

Sets the font size.

Parameters:

<FontSize>	Range:	10 to 30
	Increment:	1
	*RST:	12

POWer:REPort:LOGO <LogoFile>

Selects a path to a logo picture file.

Parameters:
<LogoFile>

POWer:REPort:PAPersize <PaperSize>

Set the layout of your report.

Parameters:
<PaperSize> A4 | USL
*RST: A4

POWer:REPort:FILE:NAME <Path>

Defines the path and file name of the report file that is to be created, saved, or deleted.

Parameters:
<Path> String containing path and file name

POWer:REPort:FILE:DELete

Deletes the selected report file.

Usage: Event

POWer:REPort:FILE:NEW

Creates a new report file.

Usage: Event

POWer:REPort:FILE:SAVE

Saves the report file.

Usage: Event

POWer:REPort:TEST:ADD <MeasType>
POWer:REPort:TEST:INSert <MeasType>, <Index>
POWer:REPort:TEST:REMove <MeasType>, <Index>

Manage reports.

Setting parameters:
<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

QUAL
Power Quality

RUSH
Inrush Current

HARM
Current Harmonic

MODU
Modulation Analysis

DONR
Dynamic ON Resistance

SLEW
Slew Rate

SOA
Safe Operating Area (S.O.A.)

TURN
Turn On/Off

SWIT
Switching Loss

EFF
Power Efficiency

RIPP
Ripple

TRANS
Transient Response

SPEC
Spectrum

<Index>

Usage: Setting only

POWer:REPort:INVert <InvertScreenshotClr>
POWer:REPort:TEST:DSEA <MeasType>
POWer:REPort:TEST:ISE <MeasType>
POWer:REPort:TEST:SEA <MeasType>
POWer:REPort:TEST:RSE <MeasType>

Manage the selection of reports.

Parameters:

<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

QUAL
Power Quality

RUSH
Inrush Current

HARM
Current Harmonic

MODU
Modulation Analysis

DONR
Dynamic ON Resistance

SLEW
Slew Rate

SOA
Safe Operating Area (S.O.A.)

TURN
Turn On/Off

SWIT
Switching Loss

EFF
Power Efficiency

RIPP
Ripple

TRANS
Transient Response

SPEC
Spectrum

POWer:REPort:TEST:DIRectory <MeasType>, <DirectoryPath>
POWer:REPort:TEST:DIRectory? <MeasType>

Selects the directory, in which the reports are saved.

Setting parameters:
<DirectoryPath>

Parameters for setting and query:

<MeasType>	QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

POWer:REPort:TEST:COMMent <MeasType>, <Comment>
POWer:REPort:TEST:COMMent? <MeasType>

Sets a comment for the report.

Setting parameters:
<Comment>

Parameters for setting and query:

<MeasType>	QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

POWer:REPort:TEST:COUNt <MeasType>

Parameters:

<MeasType>	QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

Return values:
<Count>

POWer:REPort:TEST:LSENd? <MeasType>

Query parameters:

<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

Usage: Query only

17.20.4 Power Quality

POWer:QUALity:AUTO....1914
POWer:QUALity:EXECute....1914
POWer:QUALity:FREQ....1914
POWer:QUALity:FCUS....1915
POWer:QUALity:REPort:ADD....1915
POWer:QUALity:RESult:CURRent:CREStfactor?....1915
POWer:QUALity:RESult:CURRent:FREQuency?....1915
POWer:QUALity:RESult:CURRent:PEAK?....1915
POWer:QUALity:RESult:CURRent:RMS?....1915
POWer:QUALity:RESult:POWer:APParent?....1915
POWer:QUALity:RESult:POWer:PFACtor?....1915
POWer:QUALity:RESult:POWer:PHASe?....1915
POWer:QUALity:RESult:POWer:REACtive?....1915
POWer:QUALity:RESult:POWer:REALpower?....1915
POWer:QUALity:RESult:VOLTage:CREStfactor?....1915
POWer:QUALity:RESult:VOLTage:FREQuency?....1915
POWer:QUALity:RESult:VOLTage:PEAK?....1915
POWer:QUALity:RESult:VOLTage:RMS?....1915

POWer:QUALity:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:QUALity:EXECute

Starts the power quality measurement.

Usage: Event

POWer:QUALity:FREQ <Frequency>

Sets the input frequency of the source signal in Hz.

Parameters:

<Frequency>	F50 | F60 | F360 | F400 | F650 | F800 | NFF650 | WFF800 | FCUS NFF650: 360 to 650 Hz WFF800: 360 to 800 Hz FCUS: user-defined frequency to be set using `POWer:QUALity:FCUS`. *RST: F50

POWer:QUALity:FCUS <CustomFrequency>

Sets the user-defined frequency if `POWer:QUALity:FREQ` is set to `FCUS`.

Parameters:

<CustomFrequency>	Range: 1 to 5000 Increment: 1 *RST: 16.666 Default unit: Hz

Firmware/Software: Version 2.70

POWer:QUALity:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:QUALity:RESult:CURRent:CREStfactor?
POWer:QUALity:RESult:CURRent:FREQuency?
POWer:QUALity:RESult:CURRent:PEAK?
POWer:QUALity:RESult:CURRent:RMS?
POWer:QUALity:RESult:POWer:APParent?
POWer:QUALity:RESult:POWer:PFACtor?
POWer:QUALity:RESult:POWer:PHASe?
POWer:QUALity:RESult:POWer:REACtive?
POWer:QUALity:RESult:POWer:REALpower?
POWer:QUALity:RESult:VOLTage:CREStfactor?
POWer:QUALity:RESult:VOLTage:FREQuency?
POWer:QUALity:RESult:VOLTage:PEAK?
POWer:QUALity:RESult:VOLTage:RMS?

Returns the value of the respective result.

Return values:

<Value>	Range: Depends on the measured quantity. Default unit: Depends on the measured quantity.

Usage: Query only

17.20.5 Inrush Current

This measurement is a single shot measurement. To start the measurement, use the RUNS command.

POWer:INRush:ADD ... 1916
POWer:INRush:INSert ... 1916
POWer:INRush:REMove ... 1916
POWer:INRush:COUNt? ... 1916
POWer:INRush:EXECute ... 1917
POWer:INRush:GATE<m>:STARt ... 1917
POWer:INRush:GATE<m>:STOP ... 1917
POWer:INRush:GATE<m>:VALue ... 1917
POWer:INRush:MAXCurrent ... 1917
POWer:INRush:TRIGger ... 1917
POWer:INRush:REPort:ADD ... 1918

POWer:INRush:ADD

Adds a gate.

Usage: Event

POWer:INRush:INSert <GateIndex>

Inserts a gate.

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:INRush:REMove <GateIndex>

Removes a gate

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:INRush:COUNt?

Queiries the number of inrush current gates.

Return values:
<Count>

Usage: Query only

POWer:INRush:EXECute

Starts the inrush current measurement.

Usage: Event

POWer:INRush:GATE<m>:STARt <StartTime>
POWer:INRush:GATE<m>:STOP <StopTime>

Sets the measuring time for the selected gate.

Suffix:
<m> *

Parameters:
<StopTime> Range: 0 to 10
Increment: 0
*RST: 100E-6
Default unit: s

POWer:INRush:GATE<m>:VALue <Value>

Returns the value of the inrush current.

Suffix:
<m> *

Parameters:
<Value> Range: -1000 to 1000
Increment: 0
*RST: 0
Default unit: A

POWer:INRush:MAXCurrent <MaxExpCurr>

Sets the maximum expected current for the vertical scale.

Parameters:
<MaxExpCurr> Range: -1000 to 1000
Increment: 0
*RST: 10
Default unit: A

POWer:INRush:TRIGger <CurrentValue>

Sets the current value for the trigger.

Parameters:
<CurrentValue> Range: -1000 to 1000
Increment: 0
*RST: 1
Default unit: A

POWer:INRush:REPort:ADD

Adds the result to the report list.

Usage: Event

17.20.6 Current Harmonic

POWer:HARMonics:AUTO....1918
POWer:HARMonics:DOFR....1918
POWer:HARMonics:ENFR....1918
POWer:HARMonics:MIFR....1919
POWer:HARMonics:EVAL....1919
POWer:HARMonics:EXECute....1919
POWer:HARMonics:REPort:ADD....1919
POWer:HARMonics:RESult<m>:STDinuse?....1919
POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?....1919
POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?....1919
POWer:HARMonics:RESult<m>:STDValue<n>:VALue?....1919
POWer:HARMonics:RESult<m>:VALue<n>:VALue?....1919
POWer:HARMonics:STAN....1920
POWer:HARMonics:VOLT....1920

POWer:HARMonics:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:HARMonics:DOFR <FrequencyDOA160>

Selects the frequency of the input signal.

Parameters:

<FrequencyDOA160> F360 | F400 | F650 | F800

*RST: F400

POWer:HARMonics:ENFR <FreqEN61000>

Selects the frequency of the input signal.

Parameters:

<FreqEN61000> F50 | F60

*RST: F50

POWer:HARMonics:MIFR <FreqMIL1399>

Selects the frequency of the input signal.

Parameters:
<FreqMIL1399> F400 | F60
*RST: F400

POWer:HARMonics:EVAL <AnalysisRevised>

Sets the evaluation of the results for "Standard" > "RTCA DO-160".

Parameters:
<AnalysisRevised> REVISED | NOREVISED
*RST: NOREVISED

POWer:HARMonics:EXECute

Starts the current harmonic measurement.

Usage: Event

POWer:HARMonics:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:HARMonics:RESult<m>:STDinuse?

Returns the used standard.

Suffix:
<m> 1..2

Return values:
<StandardInUse> ENA | ENB | ENC | END | MIL | RTC
*RST: ENA

Usage: Query only

POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?
POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?
POWer:HARMonics:RESult<m>:STDValue<n>:VALue?
POWer:HARMonics:RESult<m>:VALue<n>:VALue?

Returns the value of the respective result.

Suffix:
<m> 1..2

<n> *

Return values:

<Value> Range: -1000 to 1000
*RST: 0
Default unit: A

Usage: Query only

POWer:HARMonics:STAN <StandardInUse>

Sets a standard for the current harmonic measurement.

Parameters:

<StandardInUse> ENA | ENB | ENC | END | MIL | RTC
*RST: ENA

POWer:HARMonics:VOLT <DOADisplayResult>

Selects if the voltage results are displayed or not for "Standard" > "RTCA DO-160" and enabled "Evaluation with voltage source and revised current law".

Parameters:

<DOADisplayResult> VOLTDISP | NOVOLTDISP
*RST: NOVOLTDISP

Example:

```
POW:HARM:STAN RTC
POW:HARM:EVAL REVISED
POW:HARM:VOLT NOVOLTDISP
```

Selects an evaluation with the revised current law and no voltage display.

17.20.7 Modulation Analysis

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

POWer:MODulation:AUTO....1921
POWer:MODulation:DHIStogram....1921
POWer:MODulation:EXECute....1921
POWer:MODulation:REPort:ADD....1921
POWer:MODulation:RESult:ACTual?....1921
POWer:MODulation:RESult:AVG?....1921
POWer:MODulation:RESult:EVTCount?....1921
POWer:MODulation:RESult:NPEak?....1921
POWer:MODulation:RESult:PPEak?....1921
POWer:MODulation:RESult:RMS?....1921
POWer:MODulation:RESult:STDDev?....1921

POWer:MODulation:RESult:WFMCount?....................1921
POWer:MODulation:SOURce....................1922
POWer:MODulation:TYPE....................1922

POWer:MODulation:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:MODulation:DHIStogram <DispHistg>

Activates or deactivates the display of a histogram.

Parameters:

<DispHistg> ON | OFF

*RST: ON

POWer:MODulation:EXECute

Starts the modulation analysis measurement.

Usage: Event
Asynchronous command

POWer:MODulation:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:MODulation:RESult:ACTual? <MeasType>
POWer:MODulation:RESult:AVG? <MeasType>
POWer:MODulation:RESult:EVTCount? <MeasType>
POWer:MODulation:RESult:NPEak? <MeasType>
POWer:MODulation:RESult:PPEak? <MeasType>
POWer:MODulation:RESult:RMS? <MeasType>
POWer:MODulation:RESult:STDDev? <MeasType>
POWer:MODulation:RESult:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results

- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<MeasType>	FREQ | DUTY

Usage: Query only

POWer:MODulation:SOURce <Source>

Selects the source for the measurement.

Parameters:

<Source>	CURRENT | VOLTAGE
	*RST: VOLTAGE

POWer:MODulation:TYPE <AnalysisType>

Sets the type of measurement.

Parameters:

<AnalysisType>	TURNON | CONT
	*RST: CONT

17.20.8 Dynamic ON Resistance

POWer:DONRes:AUTO....1922
POWer:DONRes:AVG....1922
POWer:DONRes:EXECute....1923
POWer:DONRes:GATE<m>:STARt....1923
POWer:DONRes:GATE<m>:STOP....1923
POWer:DONRes:REPort:ADD....1923
POWer:DONRes:RESult:RESistance?....1923

POWer:DONRes:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale>	AUTO | MANUAL
	*RST: MANUAL

POWer:DONRes:AVG <Average>

Enables/disables averaging.

Parameters:

<Average>	ON | OFF
	*RST: ON

POWer:DONRes:EXECute

Starts the dynamic on resistance measurement.

Usage: Event
Asynchronous command

POWer:DONRes:GATE<m>:STARt <Start>
POWer:DONRes:GATE<m>:STOP <Stop>

Sets the value for the cursor.

Suffix:
<m> 1..2

Parameters:
<Stop>

POWer:DONRes:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:DONRes:RESult:RESistance?

Returns the the dynamic on resistance value.

Return values:
<Resistance> Range: -100E+24 to 100E+24
*RST: 0
Default unit: \x2126

Usage: Query only

17.20.9 Slew Rate

POWer:SLEWrate:SOURce 1924
POWer:SLEWrate:AUTO 1924
POWer:SLEWrate:AVGDeriv 1924
POWer:SLEWrate:EXECute 1924
ACQuire:ARESet:MODE 1924
ACQuire:ARESet:TIME 1925
ACQuire:ARESet:COUNt 1925
POWer:SLEWrate:GATE:STARt 1925
POWer:SLEWrate:GATE:STOP 1925
POWer:SLEWrate:REPort:ADD 1925
POWer:SLEWrate:RESult:ACTual? 1925
POWer:SLEWrate:RESult:AVG? 1925
POWer:SLEWrate:RESult:EVTCount? 1925
POWer:SLEWrate:RESult:NPEak? 1925

POWer:SLEWrate:RESult:PPEak?..1925
POWer:SLEWrate:RESult:RMS?... 1925
POWer:SLEWrate:RESult:STDDev?... 1926
POWer:SLEWrate:RESult:WFMCount?... 1926

POWer:SLEWrate:SOURce <Source>

Selects the source for the slew rate measurement.

Parameters:

<Source> CURRENT | VOLTAGE

*RST: VOLTAGE

POWer:SLEWrate:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:SLEWrate:AVGDeriv <AvgDerivative>

Activates or deactivates average.

Parameters:

<AvgDerivative> ON | OFF

*RST: ON

POWer:SLEWrate:EXECute

Starts the slew rate measurement.

Usage: Event

ACQuire:ARESet:MODE <ArtmRst>

Defines when the envelope and average evaluation restarts.

Parameters:

<ArtmRst> NONE | TIME | WFMS

TIME
Restarts the envelope and average calculation after the time defined with `ACQuire:ARESet:TIME`.

WFMS
Restarts the envelope and average calculation after a number of acquired waveforms defined with `ACQuire:ARESet:COUNt` on page 1925.

*RST: NONE

Usage: Asynchronous command

ACQuire:AREset:TIME <EnvelopeTimeout>

Defines the time after which the envelope and average evaluation restarts.

The setting is relevant if `ACQuire:AREset:MODE` is set to `TIME`.

Parameters:
<EnvelopeTimeout> Range: 0.1 to 10000
Increment: 0.01
*RST: 0.1
Default unit: s

Usage: Asynchronous command

ACQuire:AREset:COUNt <NofWaveforms>

Defines the number of acquired waveforms after which the envelope and average evaluation restarts.

The setting is relevant if `ACQuire:AREset:MODE` is set to `WFMS`.

Parameters:
<NofWaveforms> Range: 2 to 16777215
Increment: 10
*RST: 1000

Usage: Asynchronous command

POWer:SLEWrate:GATE:STARt <T0>
POWer:SLEWrate:GATE:STOP <T1>

Sets the value for the cursor.

Parameters:
<T1>

POWer:SLEWrate:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SLEWrate:RESult:ACTual? <MeasType>
POWer:SLEWrate:RESult:AVG? <MeasType>
POWer:SLEWrate:RESult:EVTCount? <MeasType>
POWer:SLEWrate:RESult:NPEak? <MeasType>
POWer:SLEWrate:RESult:PPEak? <MeasType>
POWer:SLEWrate:RESult:RMS? <MeasType>

POWer:SLEWrate:RESult:STDDev? <MeasType>
POWer:SLEWrate:RESult:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> MIN | MAX

Usage: Query only

17.20.10 S.O.A

POWer:SOA:EXECute....1926
POWer:SOA:LINear:ADD....1926
POWer:SOA:LOGarithmic:ADD....1926
POWer:SOA:LINear:COUNt?....1927
POWer:SOA:LOGarithmic:COUNt?....1927
POWer:SOA:LINear:REMove....1927
POWer:SOA:LOGarithmic:REMove....1927
POWer:SOA:LINear:INSert....1927
POWer:SOA:LOGarithmic:INSert....1927
POWer:SOA:LINear:POINt<m>:CURRent....1927
POWer:SOA:LOGarithmic:POINt<m>:CURRent....1927
POWer:SOA:LINear:POINt<m>:VOLTage....1927
POWer:SOA:LOGarithmic:POINt<m>:VOLTage....1927
POWer:SOA:MASK....1928
POWer:SOA:REPort:ADD....1928
POWer:SOA:SCALe....1928
POWer:SOA:SWITch....1928

POWer:SOA:EXECute

Starts the safe operating area measurement.

Usage: Event

POWer:SOA:LINear:ADD
POWer:SOA:LOGarithmic:ADD

Adds a point.

Usage: Event

POWer:SOA:LINear:COUNt?
POWer:SOA:LOGarithmic:COUNt?

Queries the number of points.

Return values:
<Count>

Usage: Query only

POWer:SOA:LINear:REMove <GateIndex>
POWer:SOA:LOGarithmic:REMove <GateIndex>

Removes a point.

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:SOA:LINear:INSert <GateIndex>
POWer:SOA:LOGarithmic:INSert <GateIndex>

Inserts a point.

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:SOA:LINear:POINt<m>:CURRent <Amp>
POWer:SOA:LOGarithmic:POINt<m>:CURRent <Amp>

Sets the current value for the respective point.

Suffix:
<m> *

Parameters:
<Amp> Range: 0.01 to 1000
Increment: 0
*RST: 0.01
Default unit: A

POWer:SOA:LINear:POINt<m>:VOLTage <Volt>
POWer:SOA:LOGarithmic:POINt<m>:VOLTage <Volt>

Sets the voltage value for the respective point.

Suffix:
<m> *

Parameters:

<Volt>	Range:	1E-3 to 1000
	Increment:	0
	*RST:	1E-3
	Default unit:	V

POWer:SOA:MASK <EnableMaskTest>

Activates or deactivates a mask.

Parameters:

<EnableMaskTest>	ON | OFF	
	*RST:	OFF

POWer:SOA:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SOA:SCALe <Scale>

Sets the scale for the measurement.

Parameters:

<Scale>	LOG | LINEAR	
	*RST:	LOG

POWer:SOA:SWITch <Switch>

Switches between linear and logarithmic scale.

Parameters:

<Switch>	LOGLINEAR | LINEARLOG	
	*RST:	LOGLINEAR

17.20.11 Turn On/Off

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

POWer:ONOFf:ATOFf ... 1929
POWer:ONOFf:ATON ... 1929
POWer:ONOFf:DTOFf ... 1929
POWer:ONOFf:DTON ... 1929
POWer:ONOFf:DSOFf ... 1929
POWer:ONOFf:DSON ... 1929
POWer:ONOFf:EXECute ... 1929
POWer:ONOFf:INPut ... 1929

POWer:ONOFf:REPort:ADD ... 1929
POWer:ONOFf:RESult:TOFF? ... 1930
POWer:ONOFf:RESult:TON? ... 1930
POWer:ONOFf:TIME ... 1930
POWer:ONOFf:TYPE ... 1930

POWer:ONOFf:ATOFf <ACTrigLevOff>
POWer:ONOFf:ATON <ACTrigLevOn>

Triggers the beginning of the measurements at the moment the AC input voltage reaches the set value.

Parameters:

<ACTrigLevOn> Range: -1E+6 to 1E+6
Increment: 1E-3
*RST: 10
Default unit: V

POWer:ONOFf:DTOFf <ACTrigLevOff>
POWer:ONOFf:DTON <ACTrigLevOn>
POWer:ONOFf:DSOFf <StateLevelOff>
POWer:ONOFf:DSON <StateLevelOn>

Sets the percentage of the steady state level of the DC output that has to be reached.

Parameters:

<StateLevelOn> Range: 0 to 100
Increment: 1
*RST: 90
Default unit: %

POWer:ONOFf:EXECute

Starts the turn on/off measurement.

Usage: Event

POWer:ONOFf:INPut <InputType>

Sets the input type.

Parameters:

<InputType> AC | DC
*RST: AC

POWer:ONOFf:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:ONOFf:RESult:TOFF?
POWer:ONOFf:RESult:TON?

Returns the result time.

Return values:

<TurnOnTime>	Range: -100E+24 to 100E+24 Increment: 100E-12 *RST: 0 Default unit: s
Usage:	Query only

POWer:ONOFf:TIME <Time>

Sets the time, the start of the measurement of the turn off time is delay with, after the trigger point.

Parameters:

<Time>	Range: 100E-12 to 10000 Increment: 100E-9 *RST: 0.1 Default unit: s

POWer:ONOFf:TYPE <MeasType>

Selects the measurement type.

Parameters:

<MeasType>	TON | TOFF TON - "Turn on" measurement TOFF - "Turn off" measurement *RST: TON

17.20.12 Switching Loss

POWer:SWITching:AUTO 1931
POWer:SWITching:EXECute 1931
POWer:SWITching:REPort:ADD 1931
POWer:SWITching:SWIFrequency 1931
POWer:SWITching:SWIT 1932
POWer:SWITching:COND 1932
POWer:SWITching:NCON 1932
POWer:SWITching:TON 1932
POWer:SWITching:TOFF 1932
POWer:SWITching:TOTal 1932
POWer:SWITching:GATE:COND:STARt 1932
POWer:SWITching:GATE:COND:STOP 1932
POWer:SWITching:GATE:NCON:STARt 1932
POWer:SWITching:GATE:TOFF:STARt 1932

POWer:SWITching:GATE:TOFF:STOP....1932
POWer:SWITching:GATE:TON:STARt....1932
POWer:SWITching:GATE:TON:STOP....1932
POWer:SWITching:RESult:ENERgy:ACTual?....1932
POWer:SWITching:RESult:ENERgy:AVG?....1932
POWer:SWITching:RESult:ENERgy:EVTCount?....1932
POWer:SWITching:RESult:ENERgy:NPEak?....1932
POWer:SWITching:RESult:ENERgy:PPEak?....1932
POWer:SWITching:RESult:ENERgy:RMS?....1932
POWer:SWITching:RESult:ENERgy:STDDev?....1932
POWer:SWITching:RESult:ENERgy:WFMCount?....1932
POWer:SWITching:RESult:POWer:ACTual?....1933
POWer:SWITching:RESult:POWer:AVG?....1933
POWer:SWITching:RESult:POWer:EVTCount?....1933
POWer:SWITching:RESult:POWer:NPEak?....1933
POWer:SWITching:RESult:POWer:PPEak?....1933
POWer:SWITching:RESult:POWer:RMS?....1933
POWer:SWITching:RESult:POWer:STDDev?....1933
POWer:SWITching:RESult:POWer:WFMCount?....1933

POWer:SWITching:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:SWITching:EXECute

Starts the switching loss measurement.

Usage: Event
Asynchronous command

POWer:SWITching:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SWITching:SWIFrequency <SwitchingFreq>

Sets the switching frequency.

Parameters:

<SwitchingFreq> Range: 1 to 500E+9
Increment: 1000
*RST: 10E+6
Default unit: Hz

POWer:SWITching:SWIT <MeasSwitchingFreq>

Activates or deactivates the measurements of the switching frequency.

Parameters:
<MeasSwitchingFreq>ON | OFF
*RST: ON

POWer:SWITching:COND <MeasureConduction>
POWer:SWITching:NCON <MeasureNonConduction>
POWer:SWITching:TON <MeasureTurnOn>
POWer:SWITching:TOFF <MeasureTurnOff>
POWer:SWITching:TOTal <MeasureTotal>

Enables the measurement during the respective period.

Parameters:
<MeasureTotal> ON | OFF
*RST: ON

POWer:SWITching:GATE:COND:STARt <T1>
POWer:SWITching:GATE:COND:STOP <T2>
POWer:SWITching:GATE:NCON:STARt <T3>
POWer:SWITching:GATE:TOFF:STARt <T2>
POWer:SWITching:GATE:TOFF:STOP <T3>
POWer:SWITching:GATE:TON:STARt <T0>
POWer:SWITching:GATE:TON:STOP <T1>

Sets the value for the respective cursor.

Parameters:
<T1>

POWer:SWITching:RESult:ENERgy:ACTual? <MeasType>
POWer:SWITching:RESult:ENERgy:AVG? <MeasType>
POWer:SWITching:RESult:ENERgy:EVTCount? <MeasType>
POWer:SWITching:RESult:ENERgy:NPEak? <MeasType>
POWer:SWITching:RESult:ENERgy:PPEak? <MeasType>
POWer:SWITching:RESult:ENERgy:RMS? <MeasType>
POWer:SWITching:RESult:ENERgy:STDDev? <MeasType>
POWer:SWITching:RESult:ENERgy:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results

- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> SWF | TON | TOF | CON | NCO | TOT

Usage: Query only

POWer:SWITching:RESult:POWer:ACTual? <MeasType>
POWer:SWITching:RESult:POWer:AVG? <MeasType>
POWer:SWITching:RESult:POWer:EVTCount? <MeasType>
POWer:SWITching:RESult:POWer:NPEak? <MeasType>
POWer:SWITching:RESult:POWer:PPEak? <MeasType>
POWer:SWITching:RESult:POWer:RMS? <MeasType>
POWer:SWITching:RESult:POWer:STDDev? <MeasType>
POWer:SWITching:RESult:POWer:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> SWF | TON | TOF | CON | NCO | TOT

Usage: Query only

17.20.13 Power Efficiency

POWer:EFFiciency:AUTO....1933
POWer:EFFiciency:EXECute....1934
POWer:EFFiciency:REPort:ADD....1934
POWer:EFFiciency:RESult<m>:ACTual?....1934
POWer:EFFiciency:RESult<m>:AVG?....1934
POWer:EFFiciency:RESult<m>:EVTCount?....1934
POWer:EFFiciency:RESult<m>:NPEak?....1934
POWer:EFFiciency:RESult<m>:PPEak?....1934
POWer:EFFiciency:RESult<m>:RMS?....1934
POWer:EFFiciency:RESult<m>:STDDev?....1934
POWer:EFFiciency:RESult<m>:WFMCount?....1934

POWer:EFFiciency:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale>	AUTO | MANUAL	
	*RST:	MANUAL

POWer:EFFiciency:EXECute

Starts the power efficiency measurement.

Usage: Event

POWer:EFFiciency:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:EFFiciency:RESult<m>:ACTual?
POWer:EFFiciency:RESult<m>:AVG?
POWer:EFFiciency:RESult<m>:EVTCount?
POWer:EFFiciency:RESult<m>:NPEak?
POWer:EFFiciency:RESult<m>:PPEak?
POWer:EFFiciency:RESult<m>:RMS?
POWer:EFFiciency:RESult<m>:STDDev?
POWer:EFFiciency:RESult<m>:WFMCount?

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Suffix:

<m>	1..3

Return values:

<WaveformsCount>	Range:	0 to 4E+9
	*RST:	0

Usage: Query only

17.20.14 Ripple

POWer:RIPPle:AUToscale....................1936
POWer:RIPPle:CURRent....................1936
POWer:RIPPle:EXECute....................1936

POWer:RIPPle:FREQuency....1936
POWer:RIPPle:REPort:ADD....1937
POWer:RIPPle:RESult:FREQuency[:ACTual]?....1937
POWer:RIPPle:RESult:FREQuency:AVG?....1937
POWer:RIPPle:RESult:FREQuency:EVTCount?....1937
POWer:RIPPle:RESult:FREQuency:NPEak?....1937
POWer:RIPPle:RESult:FREQuency:PPEak?....1937
POWer:RIPPle:RESult:FREQuency:RMS?....1937
POWer:RIPPle:RESult:FREQuency:STDDev?....1937
POWer:RIPPle:RESult:FREQuency:WFMCount?....1937
POWer:RIPPle:RESult:MAXimum[:ACTual]?....1937
POWer:RIPPle:RESult:MAXimum:AVG?....1937
POWer:RIPPle:RESult:MAXimum:EVTCount?....1937
POWer:RIPPle:RESult:MAXimum:NPEak?....1937
POWer:RIPPle:RESult:MAXimum:PPEak?....1937
POWer:RIPPle:RESult:MAXimum:RMS?....1937
POWer:RIPPle:RESult:MAXimum:STDDev?....1937
POWer:RIPPle:RESult:MAXimum:WFMCount?....1937
POWer:RIPPle:RESult:MINimum[:ACTual]?....1938
POWer:RIPPle:RESult:MINimum:AVG?....1938
POWer:RIPPle:RESult:MINimum:EVTCount?....1938
POWer:RIPPle:RESult:MINimum:NPEak?....1938
POWer:RIPPle:RESult:MINimum:PPEak?....1938
POWer:RIPPle:RESult:MINimum:RMS?....1938
POWer:RIPPle:RESult:MINimum:STDDev?....1938
POWer:RIPPle:RESult:MINimum:WFMCount?....1938
POWer:RIPPle:RESult:NDCYcle[:ACTual]?....1938
POWer:RIPPle:RESult:NDCYcle:AVG?....1938
POWer:RIPPle:RESult:NDCYcle:EVTCount?....1938
POWer:RIPPle:RESult:NDCYcle:NPEak?....1938
POWer:RIPPle:RESult:NDCYcle:PPEak?....1938
POWer:RIPPle:RESult:NDCYcle:RMS?....1938
POWer:RIPPle:RESult:NDCYcle:STDDev?....1939
POWer:RIPPle:RESult:NDCYcle:WFMCount?....1939
POWer:RIPPle:RESult:PDCYcle[:ACTual]?....1939
POWer:RIPPle:RESult:PDCYcle:AVG?....1939
POWer:RIPPle:RESult:PDCYcle:EVTCount?....1939
POWer:RIPPle:RESult:PDCYcle:NPEak?....1939
POWer:RIPPle:RESult:PDCYcle:PPEak?....1939
POWer:RIPPle:RESult:PDCYcle:RMS?....1939
POWer:RIPPle:RESult:PDCYcle:STDDev?....1939
POWer:RIPPle:RESult:PDCYcle:WFMCount?....1939
POWer:RIPPle:RESult:PDEL[:ACTual]?....1940
POWer:RIPPle:RESult:PDEL:AVG?....1940
POWer:RIPPle:RESult:PDEL:EVTCount?....1940
POWer:RIPPle:RESult:PDEL:NPEak?....1940
POWer:RIPPle:RESult:PDEL:PPEak?....1940
POWer:RIPPle:RESult:PDEL:RMS?....1940
POWer:RIPPle:RESult:PDEL:STDDev?....1940
POWer:RIPPle:RESult:PDEL:WFMCount?....1940

POWer:RIPPle:RESult:PERiod[:ACTual]?....1940
POWer:RIPPle:RESult:PERiod:AVG?....1940
POWer:RIPPle:RESult:PERiod:EVTCount?....1940
POWer:RIPPle:RESult:PERiod:NPEak?....1940
POWer:RIPPle:RESult:PERiod:PPEak?....1940
POWer:RIPPle:RESult:PERiod:RMS?....1940
POWer:RIPPle:RESult:PERiod:STDDev?....1940
POWer:RIPPle:RESult:PERiod:WFMCount?....1940
POWer:RIPPle:RESult:STDDev[:ACTual]?....1941
POWer:RIPPle:RESult:STDDev:AVG?....1941
POWer:RIPPle:RESult:STDDev:EVTCount?....1941
POWer:RIPPle:RESult:STDDev:NPEak?....1941
POWer:RIPPle:RESult:STDDev:PPEak?....1941
POWer:RIPPle:RESult:STDDev:RMS?....1941
POWer:RIPPle:RESult:STDDev:STDDev?....1941
POWer:RIPPle:RESult:STDDev:WFMCount?....1941

POWer:RIPPle:AUToscale <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale>	AUTO | MANUAL	
	*RST:	MANUAL

POWer:RIPPle:CURRent <TwoChMeas>

Activates or deactivates the input current.

Parameters:

<TwoChMeas>	ON | OFF	
	*RST:	ON

POWer:RIPPle:EXECute

Starts the ripple measurement.

Usage: Event

POWer:RIPPle:FREQuency <SmpsFrequency>

Sets the SMPS switching frequency.

Parameters:

<SmpsFrequency>	Range:	1 to 100E+9
	Increment:	1
	*RST:	1E+6
	Default unit:	Hz

POWer:RIPPle:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:RIPPle:RESult:FREQuency[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:WFMCount? [<VoltageMeasType>]

Return the specified statistic result of the frequency of the signal. The result is based on the period measurement.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result of the frequency

Usage: Query only

POWer:RIPPle:RESult:MAXimum[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the maximum value of the waveform.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results

- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:MINimum[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the minimum value of the waveform.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:NDCYcle[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:RMS? [<VoltageMeasType>]

POWer:RIPPle:RESult:NDCYcle:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the negative duty cycle. The measurement requires at least one complete period of a triggered signal.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:PDCYcle[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the positive duty cycle. The measurement requires at least one complete period of a triggered signal.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:PDEL[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the peak to peak measurement.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:

<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:PERiod[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the period, the length of the left-most signal period of the waveform.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent

*RST: VOLTage

Return values:

<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:STDDev[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the standard deviation of the long-term measurement results.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent

*RST: VOLTage

Return values:

<Result> Statistic result

Usage: Query only

17.20.15 Transient Response

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

Progragramming example: Chapter 17.3.7.2, "Transient Response Measurement", on page 1034

POWer:TRANsient:AUToscale....1942
POWer:TRANsient:EXECute....1942
POWer:TRANsient:FREQuency....1942
POWer:TRANsient:HYSTeresis....1942
POWer:TRANsient:INPut....1942

POWer:TRANsient:REPort:ADD......1943
POWer:TRANsient:RESult[:ACTual]?......1943
POWer:TRANsient:SIGHigh......1943
POWer:TRANsient:SIGLow......1943
POWer:TRANsient:TRGChannel......1943
POWer:TRANsient:TRGLevel......1944
POWer:TRANsient:TRGSlope......1944

POWer:TRANsient:AUToscale <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale>	AUTO | MANUAL	
	*RST:	MANUAL

POWer:TRANsient:EXECute

Starts the transient response measurement.

Usage: Event

POWer:TRANsient:FREQuency <SmpsFrequency>

Sets the SMPS switching frequency.

Parameters:

<SmpsFrequency>	Range:	1 to 100E+6
	Increment:	1
	*RST:	1E+6
	Default unit:	Hz

POWer:TRANsient:HYSTeresis <ToleranceTube>

Specifies a tolerated error band for the signal level.

Parameters:

<ToleranceTube>	Range:	0 to 50
	Increment:	1
	*RST:	10
	Default unit:	%

POWer:TRANsient:INPut <ThreeChMeas>

Activates or deactivates the input voltage.

Parameters:

<ThreeChMeas>	ON | OFF	
	*RST:	OFF

POWer:TRANsient:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:TRANsient:RESult[:ACTual]? <MeasType>

Returns the results of the transient response measurement.

Query parameters:

<MeasType> OVERshoot | RTIMe | DELay | PEAKtime | SETTling

Usage: Query only

POWer:TRANsient:SIGHigh <ExpHighOutputSignLev>

Sets the expected signal high voltage value.

Parameters:

<ExpHighOutputSignLev>
Range: -1000 to 1000
Increment: 1E-3
*RST: 1
Default unit: V

POWer:TRANsient:SIGLow <ExpLowOutputSignLev>

Sets the expected signal low voltage value.

Parameters:

<ExpLowOutputSignLev>
Range: -1000 to 1000
Increment: 1E-3
*RST: 0
Default unit: V

POWer:TRANsient:TRGChannel <TriggerSource>

Sets the source channel of the trigger.

Parameters:

<TriggerSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Only the measurement source channels can be used as trigger source.

*RST: CHAN1

POWer:TRANsient:TRGLevel <TriggerLevel>

Parameters:

<TriggerLevel>	Range:	-1000 to 1000
	Increment:	1E-3
	*RST:	0
	Default unit:	V

POWer:TRANsient:TRGSlope <TriggerSlope>

Sets the edge type for the trigger event.

Parameters:

<TriggerSlope>	POSitive | NEGative | EITHer	
	*RST:	POSitive

17.20.16 Spectrum

POWer:SPECtrum:AUToscale....................1944
POWer:SPECtrum:EXECute....................1944
POWer:SPECtrum:FREQuency....................1944
POWer:SPECtrum:REPort:ADD....................1945
POWer:SPECtrum:RCOunt?....................1945
POWer:SPECtrum:RESult<m>:FREQuency?....................1945
POWer:SPECtrum:RESult<m>:LEVel?....................1945

POWer:SPECtrum:AUToscale <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale>	AUTO | MANUAL	
	*RST:	MANUAL

POWer:SPECtrum:EXECute

Starts the spectrum measurement.

Usage: Event

POWer:SPECtrum:FREQuency <SmpsFrequency>

Sets the SMPS switching frequency.

Parameters:

<SmpsFrequency>	Range:	1 to 100E+6
	Increment:	1
	*RST:	1E+6
	Default unit:	Hz

POWer:SPECtrum:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SPECtrum:RCOunt?

Returns the total number of harmonics.

Return values:
<ResultCount>

Usage: Query only

POWer:SPECtrum:RESult<m>:FREQuency?

Returns the result frequency of the m-th result value.

Suffix:
<m> 1..*

Return values:
<Frequency>

Usage: Query only

POWer:SPECtrum:RESult<m>:LEVel?

Returns the result level of the m-th result value.

Suffix:
<m> 1..*

Return values:
<Level>

Usage: Query only

17.21 Maintenance

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- Chapter B.3, "Command Sequence and Synchronization", on page 1980.

- Chapter 17.3, "Programming Examples", on page 1018

DIAGnostic:SERVice:WFAModel?....................1946
DIAGnostic:SERVice:WFASeries?....................1946
DIAGnostic:SERVice:WFAType?....................1946
CALibration:DATE?....................1946
CALibration:TIME?....................1947
CALibration:RESult?....................1947
DIAGnostic:SERVice:STST:EXECute....................1947
DIAGnostic:SERVice:STST:STATe?....................1947
DIAGnostic:SERVice:PWD....................1947

DIAGnostic:SERVice:WFAModel?

Returns the model name of the oscilloscope.

Return values:
<WFAModel> Product type

Usage: Query only

DIAGnostic:SERVice:WFASeries?

Returns the model series of the oscilloscope.

Return values:
<WFASeries> RTO | RTE | RTP
*RST: RTO

Usage: Query only

DIAGnostic:SERVice:WFAType?

Returns the instrument family of the oscilloscope.

Return values:
<WFAType> RTO | RTO2000 | RTEM | RTE | RTP
*RST: RTO2000

Usage: Query only

CALibration:DATE?

Returns the date of the last selfalignment.

Return values:
<Date>

Usage: Query only
Asynchronous command

CALibration:TIME?

Returns the time of the last selfalignment.

Return values:
<Time>

Usage: Query only
Asynchronous command

CALibration:RESult?

Returns the result of the last selfalignment and the current alignment status. In remote mode, `*CAL?` provides more detailed information.

Return values:
<ResultState> PASSed | FAILed | NOALigndata
*RST: FAILed

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:STST:EXECute

Starts the selftest.

Usage: Event
Asynchronous command

DIAGnostic:SERVice:STST:STATe?

Returns the summary result of the selftest.

Return values:
<State> PSSD | FAILed | UNDefined
*RST: UNDefined

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:PWD <Password>

Sets the password to enter the service mode.

Setting parameters:
<Password> Password string

Usage: Setting only

17.22 Status Reporting

This chapter describes the remote commands that are used to read the status registers.

For information on structure, hierarchy, and contents of the status registers, see Chapter C, "Remote Control - Status Reporting System", on page 1983.

● STATus:OPERation Register....1948
● STATus:QUEStionable Registers....1949

17.22.1 STATus:OPERation Register

`STATus:OPERation` commands provide information on the activity of the instrument.

See also: Chapter C.3.4, "STATus:OPERation Register", on page 1988

STATus:OPERation:CONDition?....1948
STATus:OPERation[:EVENt]?....1948
STATus:OPERation:ENABle....1948

STATus:OPERation:CONDition?
STATus:OPERation[:EVENt]?

The CONDition command returns information on actions the instrument is currently executing. The contents of the register is retained.

The EVENt command returns information on actions the instrument has executed since the last reading. Reading the EVENt register deletes its contents.

Bits:

- 0 = ALIGnment
- 2 = AUToset
- 3= WTRIgger (wait for trigger)
- 4= MEASuring

Usage: Query only

STATus:OPERation:ENABle <Enable>

Controls the ENABle part of the STATus:OPERation register. The ENABle defines which events in the EVENt part of the status register are forwarded to the OPERation summary bit (bit 7) of the status byte. The status byte can be used to create a service request.

Parameters:

<Enable> Range: 1 to 65535
Increment: 1

Example: `STATus:OPERation:ENABle 5`
The ALIGnment event (bit 0) and AUToset event (bit 2) are forwarded to the OPERation summary bit of the status byte.

17.22.2 STATus:QUEStionable Registers

The commands of the `STATus:QUEStionable` subsystem control the status reporting structures of the STATus:QUEStionable registers.

See also: Chapter C.3.5, "STATus:QUEStionable Register", on page 1989

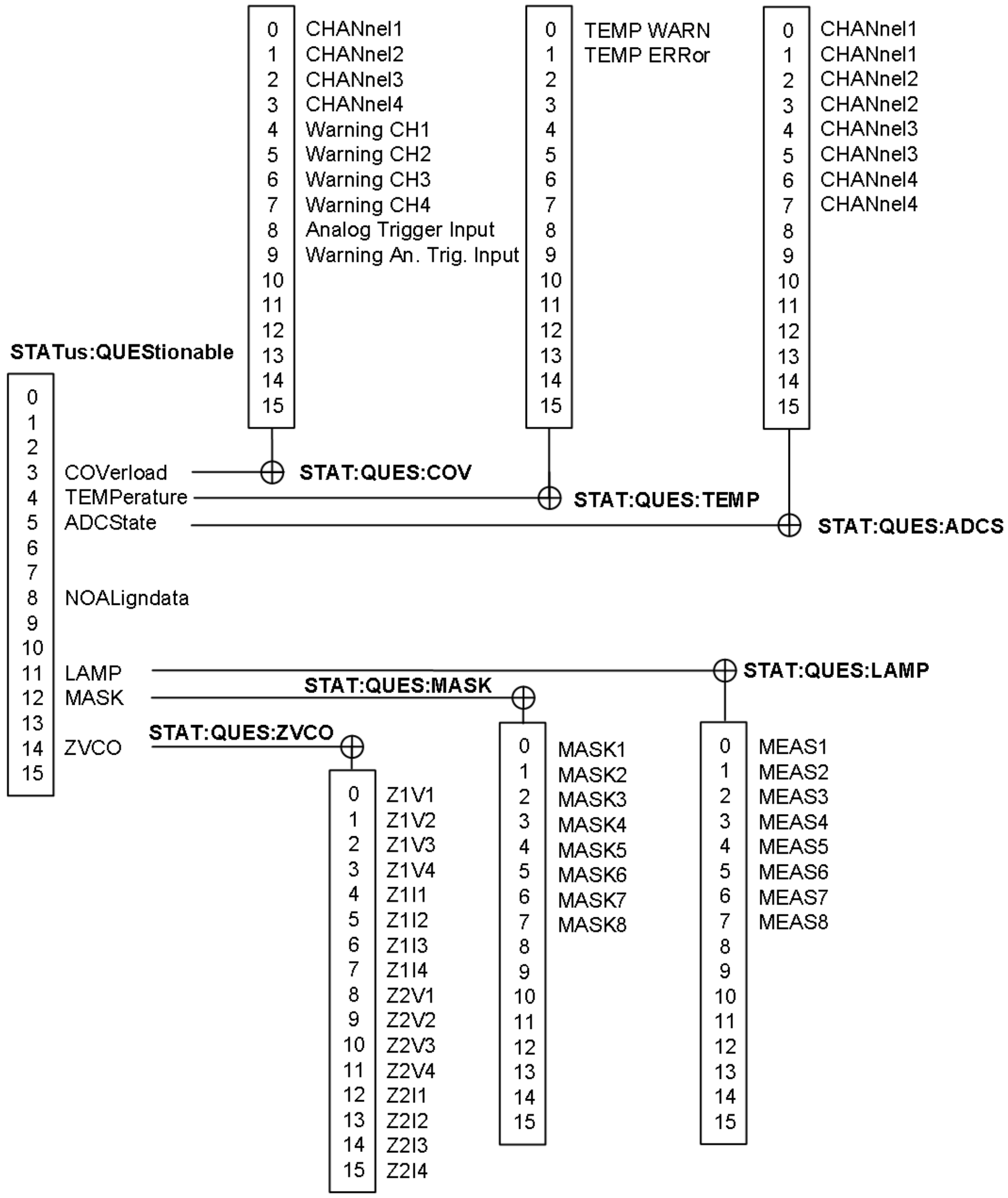


Figure 17-1: Overview of the STATus:QUEStionable register

The following commands are available:

STATus:QUEStionable:COVerload:CONDition?..1950
STATus:QUEStionable:TEMPerature:CONDition?... 1950
STATus:QUEStionable:ADCState:CONDition?... 1950
STATus:QUEStionable:LAMPlitude:CONDition?... 1950

STATus:QUEStionable:ZVCoverload:CONDition?........1950
STATus:QUEStionable:MASK:CONDition?........1950
STATus:QUEStionable:COVerload:ENABle........1950
STATus:QUEStionable:TEMPerature:ENABle........1950
STATus:QUEStionable:ADCState:ENABle........1950
STATus:QUEStionable:LAMPlitude:ENABle........1950
STATus:QUEStionable:ZVCoverload:ENABle........1951
STATus:QUEStionable:MASK:ENABle........1951
STATus:QUEStionable:COVerload[:EVENt]?........1951
STATus:QUEStionable:TEMPerature[:EVENt]?........1951
STATus:QUEStionable:ADCState[:EVENt]?........1951
STATus:QUEStionable:LAMPlitude[:EVENt]?........1951
STATus:QUEStionable:ZVCoverload[:EVENt]?........1951
STATus:QUEStionable:MASK[:EVENt]?........1951
STATus:QUEStionable:COVerload:NTRansition........1951
STATus:QUEStionable:TEMPerature:NTRansition........1951
STATus:QUEStionable:ADCState:NTRansition........1951
STATus:QUEStionable:LAMPlitude:NTRansition........1951
STATus:QUEStionable:ZVCoverload:NTRansition........1951
STATus:QUEStionable:MASK:NTRansition........1951
STATus:QUEStionable:COVerload:PTRansition........1951
STATus:QUEStionable:TEMPerature:PTRansition........1951
STATus:QUEStionable:ADCState:PTRansition........1951
STATus:QUEStionable:LAMPlitude:PTRansition........1951
STATus:QUEStionable:ZVCoverload:PTRansition........1952
STATus:QUEStionable:MASK:PTRansition........1952
MTESt:SBITnumber?........1952

STATus:QUEStionable:COVerload:CONDition?
STATus:QUEStionable:TEMPerature:CONDition?
STATus:QUEStionable:ADCState:CONDition?
STATus:QUEStionable:LAMPlitude:CONDition?
STATus:QUEStionable:ZVCoverload:CONDition?
STATus:QUEStionable:MASK:CONDition?

Returns the contents of the CONDition part of the status register to check for questionable instrument or measurement states. Reading the CONDition registers does not delete the contents.

Usage: Query only

STATus:QUEStionable:COVerload:ENABle <Value>
STATus:QUEStionable:TEMPerature:ENABle <Value>
STATus:QUEStionable:ADCState:ENABle <Value>
STATus:QUEStionable:LAMPlitude:ENABle <Value>

STATus:QUEStionable:ZVCoverload:ENABle <Value>
STATus:QUEStionable:MASK:ENABle <Value>

Sets the ENABle part that allows true conditions in the EVENt part to be reported in the summary bit. If a bit is set to 1 in the enable part and its associated event bit transitions to true, a positive transition occurs in the summary bit and is reported to the next higher level.

Parameters:

<Value> Bit mask in decimal representation

Example: `STATus:QUEStionable:MASK:ENABle 24`
Set bits no. 3 and 4 of the STATus:QUEStionable:MASK:ENABle register part: $24 = 8 + 16 = 2^3 + 2^4$

STATus:QUEStionable:COVerload[:EVENt]?
STATus:QUEStionable:TEMPerature[:EVENt]?
STATus:QUEStionable:ADCState[:EVENt]?
STATus:QUEStionable:LAMPlitude[:EVENt]?
STATus:QUEStionable:ZVCoverload[:EVENt]?
STATus:QUEStionable:MASK[:EVENt]?

Returns the contents of the EVENt part of the status register to check whether an event has occurred since the last reading. Reading an EVENt register deletes its contents.

Usage: Query only

STATus:QUEStionable:COVerload:NTRansition <Value>
STATus:QUEStionable:TEMPerature:NTRansition <Value>
STATus:QUEStionable:ADCState:NTRansition <Value>
STATus:QUEStionable:LAMPlitude:NTRansition <Value>
STATus:QUEStionable:ZVCoverload:NTRansition <Value>
STATus:QUEStionable:MASK:NTRansition <Value>

Sets the negative transition filter. If a bit is set, a 1 to 0 transition in the corresponding bit of the condition register causes a 1 to be written in the corresponding bit of the event register.

Parameters:

<Value> Bit mask in decimal representation

Example: `STATus:QUEStionable:MASK:NTRansition 24`
Set bits no. 3 and 4 of the STATus:QUEStionable:MASK:NTRansition register part: $24 = 8 + 16 = 2^3 + 2^4$

STATus:QUEStionable:COVerload:PTRansition <Value>
STATus:QUEStionable:TEMPerature:PTRansition <Value>
STATus:QUEStionable:ADCState:PTRansition <Value>
STATus:QUEStionable:LAMPlitude:PTRansition <Value>

STATus:QUEStionable:ZVCoverload:PTRansition <Value>
STATus:QUEStionable:MASK:PTRansition <Value>

Sets the positive transition filter. If a bit is set, a 0 to 1 transition in the corresponding bit of the condition register causes a 1 to be written in the corresponding bit of the event register.

Parameters:

<Value> Bit mask in decimal representation

Example: `STATus:QUEStionable:MASK:PTRansition 24`
Set bits no. 3 and 4 of the STATus:QUEStionable:MASK:PTRansition register part: $24 = 8 + 16 = 2^3 + 2^4$

MTESt:SBITnumber? <MaskTestName>

Returns the number of the status bit that belongs to the indicated mask test in the `STATus:QUEStionable:MASK` register. The status bits are assigned in the order of mask creation.

Query parameters:

<MaskTestName> String with the name of the mask test

Return values:

<StatusBitNumber> Bit number, see Chapter C.3.5.5, "STATus:QUEStionable:MASK Register", on page 1993.

Range: 0 to 7
Increment: 1
*RST: 0

Example:

```
:MTESt:SBITnumber? 'MT1'
<--0
:MTESt:SBITnumber? 'MT2'
<--1
:MTESt:SBITnumber? 'MT3'
<--2
```

Usage: Query only
Asynchronous command

17.23 Remote Trace

The commands in this section configure tracing of the remote control interface and of events. They also configure the display of the SCPI remote trace.

Before you start tracing, configure all settings as desired. Modifying settings while tracing is active may result in loss of already traced data. Useful exception: Selecting a new target file while tracing is allowed. For start mode `EXPLicit` a restart of the instrument resets the settings to the documented default values.

If you want to start tracing already during startup of the instrument, configure all settings (including start mode AUTO). Then restart your instrument. Tracing will be started automatically during the restart, using the already configured settings.

If you use an XML file as trace file, ensure that tracing is stopped properly. If tracing is aborted instead of stopped, for example by shutting down the instrument for stop mode EXPLicit, the XML file will be invalid, because some tags are not closed.

When the maximum file size is reached (except for stop mode BUFFerfull) or if tracing is started with an already existing trace file, a backup of the trace file is created and the file itself is reset and overwritten. When the file is full for the second time or when tracing is started the next time, the first backup file is lost because it is overwritten by the next backup. In order to prevent loss of data, set a sufficient file size, select an appropriate stop mode and archive/copy completed trace files if you want to keep them.

17.23.1 Standard Commands

TRACe:REMote:MODE:FILE:NAME....................1953
TRACe:REMote:MODE:FILE:FORMat....................1953
TRACe:REMote:MODE:FILE:SIZE....................1953
TRACe:REMote:MODE:FILE:STARtmode....................1954
TRACe:REMote:MODE:FILE:STOPmode....................1954
TRACe:REMote:MODE:FILE:ENABle....................1954
TRACe:REMote:MODE:FILE:FILTer....................1954

TRACe:REMote:MODE:FILE:NAME <FilePath>

Sets the directory and file name where the remote trace file is stored.

Parameters:

<FilePath> String parameter

TRACe:REMote:MODE:FILE:FORMat <Format>

Sets the file format of the remote trace file.

Parameters:

<Format> ASCii | XML

TRACe:REMote:MODE:FILE:SIZE <FileSize>

Sets the maximum size of the remote trace file.

Parameters:

<FileSize> Integer value

TRACe:REMote:MODE:FILE:STARtmode <StartMode>

Defines how the remote trace is started.

Parameters:

<StartMode> AUTO | EXPLicit

AUTO
Starts the remote trace immediately.

EXPLicit
Starts remote trace with `TRACe:REMote:MODE:FILE:ENABle ON`

TRACe:REMote:MODE:FILE:STOPmode <StopMode>

Defines when the remote trace is stopped.

Parameters:

<StopMode> AUTO | EXPLicit | ERRor | BUFFerfull

AUTO
Ends the remote trace on device shutdown.

EXPLicit
Ends remote trace with `TRACe:REMote:MODE:FILE:ENABle OFF`

ERRor
Ends remote trace when a SCPI error occurs.

BUFFerfull
Ends remote trace when the maximum file size is reached.

TRACe:REMote:MODE:FILE:ENABle <Enable>

Enables and disables the remote trace to file.

Parameters:

<Enable> ON | OFF

TRACe:REMote:MODE:FILE:FILTer <Input>, <Output>, <Error>, <Trigger>, <DeviceClear>, <StatusRegister>, <Connection>, <RemoteLocalEvents>, <Locking>

Defines the content of the remote trace file.

Parameters:

<Input> ON | OFF

Input data

<Output> ON | OFF

Output data

<Error>	ON | OFF New SCPI error queue entries
<Trigger>	ON | OFF Trigger events
<DeviceClear>	ON | OFF Device clear events
<StatusRegister>	ON | OFF Status register conditions
<Connection>	ON | OFF Open/close connection events
<RemoteLocalEvents>	ON | OFF Local/remote transition events
<Locking>	ON | OFF Remote locking events

17.23.2 Diagnostic Remote Trace Commands

TRACe:REMote:MODE:FILE:DEXecution:DURation....1955
TRACe:REMote:MODE:FILE:RPC....1955
TRACe:REMote:MODE:FILE:PARSer....1955
TRACe:REMote:MODE:FILE:FUNCtions....1956

TRACe:REMote:MODE:FILE:DEXecution:DURation <Enable>

Traces the device execution time of a command

Parameters:
<Enable> ON | OFF

TRACe:REMote:MODE:FILE:RPC <Enable>

Enables and disables output of rpc calls to remote trace.

Parameters:
<Enable> ON | OFF

TRACe:REMote:MODE:FILE:PARSer <Enable>

Enables and disables output of parser transitions to remote trace.

Parameters:
<Enable> ON | OFF

TRACe:REMote:MODE:FILE:FUNCtions <Enable>

Enables and disables output of function names to remote trace.

Parameters:
<Enable> ON | OFF

17.24 Deprecated Commands

The following commands are provided for compatibility to previous oscilloscope versions only. For new remote control programs, use the specified alternative commands.

17.24.1 Base Instrument

Legacy command	Replaced by	Comment
ACQuire:ARESet:WFMCount	ACQuire:ARESet:COUNt on page 1925	Replaced since FW 2.70
BUSFormat	For parallel buses: BUS<m>:PARallel:DATA:FORMat on page 1870 For serial buses: BUS<m>:FORMat on page 1384	Replaced since FW 4.70
MEASurement<m>:SPECtrum:NREJect		Functionality is not supported.
MEASurement<m>:SPECtrum:THReshold	MEASurement<m>:SPECtrum:ATHReshold on page 1235	Replaced since FW 3.30
SEARch:TRIGger:INTerval:POLarity	SEARch:TRIGger:INTerval:SLOPe on page 1320	Replaced since FW 2.70
TRIGger<m>:INTerval:POLarity	TRIGger<m>:INTerval:SLOPe on page 1142	Replaced since FW 2.70
SYSTem:RESet	*RST on page 1043	Replaced since FW 4.60

17.24.2 Options

Legacy command	Replaced by	Comment
CAN, CAN FD		
SEARch:TRIGger:CAN[:STATe]	SEARch:TRIGger:CAN[:SSOFrame] on page 1472 SEARch:TRIGger:CAN:SFTYpe on page 1472 SEARch:TRIGger:CAN:SFIDentifier on page 1472 SEARch:TRIGger:CAN:SIDData on page 1473 SEARch:TRIGger:CAN:SERRor on page 1473 SEARch:TRIGger:CAN:SSYMbolic on page 1489	Replaced since FW 2.25
Ethernet 100BASE-T1		
TRIGger<m>:HBTO:RSTatus		Functionality is not supported.
SEARch:TRIGger:HBTO:RSTatus		Functionality is not supported.
FlexRay		
SEARch:TRIGger:FLXRay[:STATe]		Functionality is not supported.
Manchester and NRZ		
SEARch:TRIGger:CMSB:TYPE		Functionality is not supported.
SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:BIT SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:DMAX SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:DMIN SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:DOPerator SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:ENABle SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:IMAX SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:IMIN SEARch:TRIGger:CMSB:FRAMe<m>:FIELd<n>:IOPerator	SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT on page 1728 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX on page 1728 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN on page 1729 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator on page 1729 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle on page 1730 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX on page 1730 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN on page 1731 SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator on page 1731	Replaced since FW 4.15
MDIO		
BUS<m>:MDIO:MAXGap		Functionality is not supported.
SpaceWire		
BUS<m>:SWIRe:MINGap	BUS<m>:SWIRe:MGAP on page 1819	

Legacy command	Replaced by	Comment
BUS<m>:SWIRe:THRCoupling	BUS<m>:SWIRe:COUPling on page 1821	Replaced since FW 3.30
BUS<m>:SWIRe:THRPreset	BUS<m>:SWIRe:PRESet on page 1821	Replaced since FW 3.30
USBPD		
SEARch:TRIGger:USBPd:TYPE TRIGger<m>:USBPd:TYPE		Functionality is not supported.
SEARch:TRIGger:USBPd:FRAMe:SELect TRIGger<m>:USBPd:FRAMe:SELect		Functionality is not supported.
SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:BIT SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:DMAX SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:DMIN SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:DOPerator SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:ENABle SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:IMAX SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:IMIN SEARch:TRIGger:USBPd:FRAMe<m>:FIELd<n>:IOPerator	SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT on page 1809 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX on page 1810 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN on page 1810 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator on page 1811 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle on page 1809 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX on page 1811 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN on page 1812 SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator on page 1812	Replaced since FW 4.15
TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:BIT TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:DMAX TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:DMIN TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:DOPerator TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:ENABle TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:IMAX TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:IMIN TRIGger<m>:USBPd:FRAMe<n>:FIELd<o>:IOPerator	TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT on page 1799 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX on page 1800 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN on page 1800 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator on page 1800 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle on page 1799 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX on page 1801 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN on page 1801 TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator on page 1802	Replaced since FW 4.15

18 Maintenance and Support

The instrument does not need periodic maintenance. Only the cleaning of the instrument is essential.

To protect the front panel and to transport the instrument to another workplace safely and easily, various accessories are provided. Refer to the data sheet for available covers and cases and their order numbers.

The addresses of Rohde & Schwarz support centers can be found at www.customer-support.rohde-schwarz.com. A list of all service centers is available on www.services.rohde-schwarz.com.

18.1 Cleaning

1. Clean the outside of the instrument using a soft, dry, lint-free dust cloth.

   Note: Do not use any liquids for cleaning. Cleaning agents, solvents (thinners, acetone), acids and bases can damage the front panel labeling, plastic parts and display.

2. Check and clean the fans regularly to ensure that they always operate properly.

3. Clean the touchscreen as follows:
   a) Apply a small amount of standard screen cleaner to a soft cloth.
   b) Wipe the screen gently with the moist, but not wet, cloth.
   c) If necessary, remove any excess moisture with a dry, soft cloth.

18.2 Contacting Customer Support

Technical support – where and when you need it

For quick, expert help with any Rohde & Schwarz product, contact our customer support center. A team of highly qualified engineers provides support and works with you to find a solution to your query on any aspect of the operation, programming or applications of Rohde & Schwarz products.

Contact information

Contact our customer support center at www.rohde-schwarz.com/support, or follow this QR code:

Figure 18-1: QR code to the Rohde & Schwarz support page

18.3 Information for Technical Support

If you encounter problems that you cannot solve yourself, contact your Rohde & Schwarz support center, see Chapter 18.2, "Contacting Customer Support", on page 1959. Our support center staff is optimally trained to assist you in solving problems.

The support center finds solutions more quickly and efficiently, if you provide them with information on the instrument and an error description. To create, collect and save the required information you can use the RTxServiceReporter. The RTxServiceReporter creates a ZIP file with a complete bug report, all relevant setup information, reporting and log files, and the instrument configuration (device footprint).

1. Open the App Cockpit: tap .
2. On the "R&S Apps" tab, tap "Service".

   The RTxServiceReporter creates the report and saves it as ZIP file directly on the Windows desktop.
3. Attach the report file to an email in which you describe the problem. Send the email to the customer support address for your region as listed in the internet.

On the instrument, you can find log files, the device footprint and report files, and other information in the "AppData" folder on the Windows desktop (`C:\ProgramData\Rohde-Schwarz\RTx`).

The "Maintenance" dialog box also provides information on your instrument configuration which can be helpful in case you need support.

18.4 Data Security

If you have to send the instrument to the service, or if the instrument is used in a secured environment, consider the document "Instrument Security Procedures" that is delivered on the R&S RTE web page.

Instrument configuration data and user data are stored on a removable hard disk only. Thus it is sufficient to remove the hard disk before the instrument leaves a secured environment. Details are given in the document mentioned above.

18.5 Storing and Packing

The storage temperature range of the instrument is given in the data sheet. If the instrument is to be stored for a longer period of time, it must be protected against dust.

Repack the instrument as it was originally packed when transporting or shipping. The two protective foam plastic parts prevent the control elements and connectors from being damaged. The antistatic packing foil avoids any undesired electrostatic charging to occur.

If you do not use the original packaging, use a sturdy cardboard box of suitable size and provide for sufficient padding to prevent the instrument from slipping inside the package. Wrap antistatic packing foil around the instrument to protect it from electrostatic charging.

18.6 Maintenance Information

"File" menu > "Maintenance"

The "Maintenance" dialog box provides information on your R&S RTE configuration, which can be helpful in case you need support.

System Info
This tab provides general information on the hardware configuration, and indicates where system information can be found on the instrument. Here you can also show the content of the device footprint file.

Remote command:
`DIAGnostic:SERVice:PARTnumber` on page 1050
`DIAGnostic:SERVice:SERialnumber?` on page 1050
`SYSTem:DFPRint` on page 1051

Mainboard
This tab provides information on the mainboard configuration in your instrument.

Frontend
This tab provides information on the frontend configuration in your instrument.

Frontpanel
This tab provides information on the front panel module installed in your instrument.

MSO
This tab is only relevant if the MSO option R&S RTE-B1 is installed. The tab provides information on the MSO hardware module that is installed in your instrument.

AWG

This tab is only relevant if the waveform generator option R&S RTE-B6 is installed. The tab provides information on the generator hardware module that is installed in your instrument.

PSC

This tab is only relevant if the pulse source option R&S RTE-B7 is installed. The tab provides information on the pulse source hardware module that is installed in your instrument.

Service

This tab allows the service personnel to enter a password that activates further service functions.

Remote command:

- `DIAGnostic:SERVice:PWD` on page 1947

Annex

A Menu Overview

This section provides an overview of the menus together with a short description or link to the description.

- File Menu....1963
- Horizontal Menu....1964
- Trigger Menu....1964
- Vertical Menu....1964
- Math Menu....1965
- Cursor Menu....1965
- Meas Menu....1966
- Masks Menu....1966
- Analysis Menu....1966
- Display Menu....1967
- WaveGen Menu (with Option R&S RTE-B6)....1968

A.1 File Menu

Menu item	Description	Corresponding key
Save Recall	Chapter 11.1, "Instrument Settings", on page 430 Chapter 11.2.1, "Waveform Export Files", on page 439 Chapter 11.3, "Autonaming", on page 459	[FILE]
Setup	Chapter 3, "Instrument Setup", on page 94	[SETUP]
Print Setup	Chapter 11.4, "Screenshots", on page 461 Chapter 11.4.2, "Printing Screenshots", on page 466	[PRINT]
Report Setup	Chapter 11.5, "Reports", on page 467	
Frontpanel Setup	Chapter 3.3, "Frontpanel Setup", on page 101	
External Setup	Chapter 3.5, "External Application", on page 120	
Preset Setup	Chapter 11.6, "Preset Setup", on page 469	
Help	Chapter 2.4.12, "Getting Information and Help", on page 91	[HELP]
Mode	Chapter 3.9.4, "Options in Beta State", on page 127	[MODE]
Maintenance	Chapter 18.6, "Maintenance Information", on page 1961	
Selfalignment	Chapter 3.6, "Self-alignment", on page 120	
Selftest	Chapter 3.7, "Self-test", on page 122	

Menu item	Description	Corresponding key
Demo Board	For internal use only. Opens a setup dialog box for the demo board If a demo board is connected to the instrument.	
Minimize Application	Shows the Windows desktop with the application icon of the R&S RTE firmware.	
Exit	Shuts down the firmware.	

A.2 Horizontal Menu

Menu item	Description	Corresponding key
Setup	Chapter 4.2.1, "Setup", on page 138	[HORIZONTAL]
Acquisition	Chapter 4.2.2, "Acquisition", on page 143	[ACQUISITION]
Fast Segmentation	Chapter 4.2.3, "Fast Segmentation", on page 145	
Skew	Chapter 4.9.1, "Skew", on page 194	

A.3 Trigger Menu

Menu item	Description	Corresponding key
Setup	Chapter 5.3, "Trigger Types", on page 203	[TRIGGER]
Noise Reject	Chapter 5.5, "Noise Reject", on page 227	
Holdoff	Chapter 5.4, "Holdoff", on page 226	
Ctrl/Action	Chapter 5.6, "Control / Action", on page 229	
Extern		
Digital Filter	Chapter 4.8, "Digital Filter Setup", on page 192	
Acquisition Info	Shows the current number of acquisitions that have been acquired.	

A.4 Vertical Menu

Menu item	Description	Corresponding key
Channels	Chapter 4.3.1, "Channels", on page 147	[CH ×]
Coupled Channels		
Power Calculation	Chapter 4.3.3, "Power Calculation", on page 150	

Menu item	Description	Corresponding key
Probe Setup	Chapter 4.5, "Probes", on page 155	
Probe Attributes	Chapter 4.5.9, "Probe Attributes", on page 177	
Calibration Results	Chapter 4.5.10, "Calibration Results", on page 178	
Digital Filter	Chapter 4.8, "Digital Filter Setup", on page 192	
ZVC Multi-Channel Probe	Chapter 4.6, "R&S RT-ZVC Probe", on page 179	

A.5 Math Menu

Menu item	Description	Key / Icon
Math Setup	Chapter 6.3.2, "Math Setup - General Settings", on page 258	[MATH]
FFT Setup	Chapter 8.1.3.1, "FFT Setup", on page 366	
FFT Overlap		
FFT Gating	Chapter 8.1.3.3, "FFT Gating", on page 372	
FFT Y-Units		
Reference Waveform	Submenu: Setup, Scaling, Original Attributes: Chapter 6.2.2, "Settings for Reference Waveforms", on page 254	

A.6 Cursor Menu

Menu item	Description	Key / Icon
Setup	Chapter 7.1.3.1, "Cursor Setup", on page 288	[CURSOR] 
Label	Chapter 7.1.3.2, "Cursor Labels and Display", on page 291	
Peak Search	Chapter 7.1.3.3, "Peak Search ", on page 292	

A.7 Meas Menu

Menu item	Description	Key / Icon
"Setup"	Chapter 7.2.1.3, "General Measurement Settings", on page 297	[MEAS]
Parameters	• Chapter 7.2.5.2, "Settings for Amplitude/Time Measurements ", on page 315 • Chapter 7.2.6.2, "Settings for Eye Diagram Measurements", on page 324 • Chapter 7.2.7.2, "Settings for Spectrum Measurements ", on page 326 • Chapter 7.2.8.5, "Settings for Histogram Measurement ", on page 336	
Result Analysis	Chapter 7.2.10.5, "Settings for Result Analysis", on page 348	
Gate/Display	Chapter 7.2.3.2, "Gate Settings for Measurements", on page 303 Chapter 7.2.2.3, "Display Settings for Results", on page 301	
Histogram	Chapter 7.2.8.4, "Histogram Setup", on page 334	
Reference Level	Chapter 7.2.4.2, "Level Settings", on page 307	

A.8 Masks Menu

Menu item	Description	Key / Icon
Test Definition	Chapter 9.2.1, "Test Definition", on page 385	[MASKS] Opens the last selected tab in the "Masks" dialog box.
Mask Definition	Chapter 9.2.2.1, "Mask Definition: User Mask", on page 387	
Event Actions / Reset	Chapter 9.2.3, "Event Actions /Reset ", on page 393	
Mask Display	Chapter 9.2.4, "Mask Display", on page 396	

A.9 Analysis Menu

The content of the menu depends on the installed options.

Menu item	Description	Corresponding key
Power	Chapter 15, "Power Analysis (Option R&S RTE-K31)", on page 922	
Parallel bus	Chapter 13.1.1, "Parallel Buses - Configuration", on page 868	[LOGIC]
Search	Chapter 10, "Search Functions", on page 406	[SEARCH]
Serial Bus	Chapter 12, "Protocol Analysis", on page 473	[PROTOCOL]

A.10 Display Menu

Menu item	Description	Key / Icon
Signal Colors / Persistence	Chapter 3.4.2.1, "Colors / Persistence", on page 105	[DISPLAY] Opens the last selected tab in the "Display" dialog box.
Color Tables	Chapter 3.4.2.2, "Color Tables", on page 108	
Diagram Layout	Chapter 3.4.2.3, "Diagram Layout", on page 109	
XY-Diagram	Chapter 6.5, "XY-Diagram", on page 280	
Labels		
Zoom	Chapter 6.1, "Zoom", on page 240	[ZOOM] 
Show history	Enables the history mode and opens the quick-access "History" dialog box. Chapter 6.4, "History", on page 273	[HISTORY]
History setup	Opens the "History" configuration dialog box without starting the history mode. Chapter 6.4.2, "History Setup", on page 275	
Show performance	Displays the current performance values of the instrument.	
Clear all	Chapter 3.4.2.6, "Clear Results", on page 114	
Toolbar	Chapter 2.4.7.2, "Configuring the Toolbar", on page 80	
Show Signal Bar	Switches the signal bar on or off. Chapter 2.4.6, "Using the Signal bar", on page 78	

A.11 WaveGen Menu (with Option R&S RTE-B6)

Menu item	Description	Corresponding key
Setup	Chapter 14.1, "Setup of the Waveform Generator", on page 893	
Coupling & Sync	Chapter 14.3, "Coupling and Sync Settings", on page 916	
DC offset alignment	Chapter 14.5, "DC Offset Alignment", on page 921	

B Remote Control - Basics

- Messages 1969
- SCPI Command Structure 1971
- Command Sequence and Synchronization 1980
- General Programming Recommendations 1982

B.1 Messages

B.1.1 Instrument Messages

Instrument messages are employed in the same way for all interfaces, if not indicated otherwise in the description.

There are different types of instrument messages, depending on the direction they are sent:

- Commands
- Instrument responses

Structure and syntax of the instrument messages are described in Chapter B.2, "SCPI Command Structure", on page 1971.

Commands

Commands (program messages) are messages that the controller sends to the instrument. They operate the instrument functions and request information. The commands are subdivided according to two criteria:

- According to the effect on the instrument:
  – **Setting commands** cause instrument settings such as a reset of the instrument or setting the frequency.
  – **Queries** cause data to be provided for remote control, e.g. for identification of the instrument or polling a parameter value. Queries are formed by directly appending a question mark to the command header.
- According to their definition in standards:
  – **Common commands**: their function and syntax are precisely defined in standard IEEE 488.2. They are employed identically on all instruments (if implemented). They refer to functions such as management of the standardized status registers, reset and self-test.
  – **Instrument control commands** refer to functions depending on the features of the instrument such as frequency settings. Many of these commands have also been standardized by the SCPI committee. These commands are marked as "SCPI compliant" in the command reference chapters. Commands without this SCPI label are device-specific, however, their syntax follows SCPI rules as permitted by the standard.

Instrument responses

Instrument responses (response messages and service requests) are messages that the instrument sends to the controller after a query. They can contain measurement results, instrument settings and information on the instrument status.

B.1.2 Interface Massages

Interface messages are transmitted to the instrument on the data lines. They are used to communicate between the controller and the instrument.

B.1.2.1 GPIB Interface Messages

Interface messages are transmitted to the instrument on the data lines, with the attention line (ATN) being active (LOW). They are used for communication between the controller and the instrument and can only be sent by a computer which has the function of a GPIB bus controller. GPIB interface messages can be further subdivided into:

- **Universal commands**: act on all instruments connected to the GPIB bus without previous addressing
- **Addressed commands**: only act on instruments previously addressed as listeners

Universal Commands

Universal commands are encoded in the range 10 through 1F hex. They affect all instruments connected to the bus and do not require addressing.

Command	Effect on the instrument
DCL (Device Clear)	Aborts the processing of the commands just received and sets the command processing software to a defined initial state. Does not change the instrument settings.
IFC (Interface Clear) *)	Resets the interfaces to the default setting.
LLO (Local Lockout)	The "Local" softkey is disabled. Manual operation is no longer available until `GTL` is executed.
SPE (Serial Poll Enable)	Ready for serial poll.
SPD (Serial Poll Disable)	End of serial poll.
PPU (Parallel Poll Unconfigure)	End of the parallel-poll state.
*) IFC is not a real universal command, it is sent via a separate line; however, it also affects all instruments connected to the bus and does not require addressing	

Addressed Commands

Addressed commands are encoded in the range 00 through 0F hex. They only affect instruments addressed as listeners.

Command	Effect on the instrument
GET (Group Execute Trigger)	Triggers a previously active instrument function (e.g. a sweep). The effect of the command is the same as with that of a pulse at the external trigger signal input.
GTL (Go to Local)	Transition to the "local" state (manual control).
GTR (Go to Remote)	Transition to the "remote" state (remote control).
PPC (Parallel Poll Configure)	Configures the instrument for parallel poll.
SDC (Selected Device Clear)	Aborts the processing of the commands just received and sets the command processing software to a defined initial state. Does not change the instrument setting.

B.1.2.2 LAN Interface Messages

In the LAN connection, the interface messages are called low–level control messages. These messages can be used to emulate interface messages of the GPIB bus.

Command	Long term	Effect on the instrument
&ABO	Abort	Aborts processing of the commands just received.
&DCL	Device Clear	Aborts processing of the commands just received and sets the command processing software to a defined initial state. Does not change the instrument setting.
&GTL	Go to Local	Transition to the "local" state (manual control). (The instrument automatically returns to remote state when a remote command is sent UNLESS `&NREN` was sent before.)
&GTR	Go to Remote	Enables automatic transition from local state to remote state by a subsequent remote command (after `&NREN` was sent).
&GET	Group Execute Trigger	Triggers a previously active instrument function (e.g. a sweep). The effect of the command is the same as with that of a pulse at the external trigger signal input.
&LLO	Local Lockout	Disables transition from remote control to manual control by means of the front panel keys.
&NREN	Not Remote Enable	Disables automatic transition from local state to remote state by a subsequent remote command. (To re-activate automatic transition use `&GTR`.)
&POL	Serial Poll	Starts a serial poll.

B.2 SCPI Command Structure

SCPI commands consist of a header and, in most cases, one or more parameters. The header and the parameters are separated by a "white space" (ASCII code 0 to 9, 11 to 32 decimal, e.g. blank). The headers may consist of several mnemonics (keywords). Queries are formed by appending a question mark directly to the header.

The commands can be either device-specific or device-independent (common commands). Common and device-specific commands differ in their syntax.

B.2.1 Syntax for Common Commands

Common (= device-independent) commands consist of a header preceded by an asterisk (*), and possibly one or more parameters.

Table B-1: Examples of common commands

*RST	RESET	Resets the instrument.
*ESE	EVENT STATUS ENABLE	Sets the bits of the event status enable registers.
*ESR?	EVENT STATUS QUERY	Queries the contents of the event status register.
*IDN?	IDENTIFICATION QUERY	Queries the instrument identification string.

B.2.2 Syntax for Device-Specific Commands

Not all commands used in the following examples are necessarily implemented in the instrument. For demonstration purposes only, assume the existence of the following commands for this section:

- `DISPlay[:WINDow<1...4>]:MAXimize <Boolean>`
- `FORMat:READings:DATA <type>[,<length>]`
- `HCOPy:DEVice:COLor <Boolean>`
- `HCOPy:DEVice:CMAP:COLor:RGB <red>,<green>,<blue>`
- `HCOPy[:IMMediate]`
- `HCOPy:ITEM:ALL`
- `HCOPy:ITEM:LABel <string>`
- `HCOPy:PAGE:DIMensions:QUADrant[<N>]`
- `HCOPy:PAGE:ORIentation LANDscape | PORTrait`
- `HCOPy:PAGE:SCALe <numeric value>`
- `MMEMory:COPY <file_source>,<file_destination>`
- `SENSE:BANDwidth|BWIDth[:RESolution] <numeric_value>`
- `SENSe:FREQuency:STOP <numeric value>`
- `SENSe:LIST:FREQuency <numeric_value>{,<numeric_value>}`

- Long and short form....................1973
- Numeric Suffixes....................1973
- Optional Mnemonics....................1973

B.2.2.1 Long and short form

The mnemonics feature a long form and a short form. The short form is marked by upper case letters, the long form corresponds to the complete word. Either the short form or the long form can be entered; other abbreviations are not permitted.

Example:

`HCOPy:DEVice:COLor ON` is equivalent to `HCOP:DEV:COL ON`.

Case-insensitivity

Upper case and lower case notation only serves to distinguish the two forms in the manual, the instrument itself is case-insensitive.

B.2.2.2 Numeric Suffixes

If a command can be applied to multiple instances of an object, e.g. specific channels or sources, the required instances can be specified by a suffix added to the command. Numeric suffixes are indicated by angular brackets (<1...4>, <n>, <i>) and are replaced by a single value in the command. Entries without a suffix are interpreted as having the suffix 1.

Example:

Definition: `HCOPy:PAGE:DIMensions:QUADrant[<N>]`

Command: `HCOP:PAGE:DIM:QUAD2`

This command refers to the quadrant 2.

Different numbering in remote control

For remote control, the suffix may differ from the number of the corresponding selection used in manual operation. SCPI prescribes that suffix counting starts with 1. Suffix 1 is the default state and used when no specific suffix is specified.

Some standards define a fixed numbering, starting with 0. If the numbering differs in manual operation and remote control, it is indicated for the corresponding command.

B.2.2.3 Optional Mnemonics

Some command systems permit certain mnemonics to be inserted into the header or omitted. These mnemonics are marked by square brackets in the description. The instrument must recognize the long command to comply with the SCPI standard. Some commands are considerably shortened by these optional mnemonics.

Example:

Definition: `HCOPy[:IMMediate]`

Command: `HCOP:IMM` is equivalent to `HCOP`

Optional mnemonics with numeric suffixes

Do not omit an optional mnemonic if it includes a numeric suffix that is relevant for the effect of the command.

Example:

Definition:`DISPlay[:WINDow<1...4>]:MAXimize <Boolean>`

Command: `DISP:MAX ON` refers to window 1.

In order to refer to a window other than 1, you must include the optional `WINDow` parameter with the suffix for the required window.

`DISP:WIND2:MAX ON` refers to window 2.

B.2.3 SCPI Parameters

Many commands are supplemented by a parameter or a list of parameters. The parameters must be separated from the header by a "white space" (ASCII code 0 to 9, 11 to 32 decimal, e.g. blank).

The parameters required for each command and the allowed range of values are specified in the command description.

Allowed parameters are:

- Numeric Values....................1974
- Special Numeric Values....................1975
- Boolean Parameters....................1976
- Text Parameters....................1976
- Character Strings....................1976
- Block Data....................1976

B.2.3.1 Numeric Values

Numeric values can be entered in any form, i.e. with sign, decimal point and exponent. Values exceeding the resolution of the instrument are rounded up or down. The mantissa may comprise up to 255 characters, the exponent must lie inside the value range -32000 to 32000. The exponent is introduced by an "E" or "e". Entry of the exponent alone is not allowed.

Example:

`SENS:FREQ:STOP 1500000` = `SENS:FREQ:STOP 1.5E6`

Units

For physical quantities, the unit can be entered. If the unit is missing, the basic unit is used. Allowed unit prefixes are:

- G (giga)
- MA (mega), MOHM, MHZ
- K (kilo)
- M (milli)

- U (micro)
- N (nano)

Example:

`SENSe:FREQ:STOP 1.5GHz` = `SENSe:FREQ:STOP 1.5E9`

Some settings allow relative values to be stated in percent. According to SCPI, this unit is represented by the `PCT` string.

Example:

`HCOP:PAGE:SCAL 90PCT`

B.2.3.2 Special Numeric Values

The following mnemonics are special numeric values. In the response to a query, the numeric value is provided.

- **MIN and MAX**: denote the minimum and maximum value.
- **DEF**: denotes a preset value which has been stored in the EPROM. This value conforms to the default setting, as it is called by the *RST command.
- **UP and DOWN**: increases or reduces the numeric value by one step. The step width can be specified via an allocated step command for each parameter which can be set via UP and DOWN.
- **INF and NINF**: INFinity and negative INFinity (NINF) represent the numeric values 9.9E37 or -9.9E37, respectively. INF and NINF are only sent as instrument responses.
- **NAN**: Not A Number (NAN) represents the value 9.91E37. NAN is only sent as a instrument response. This value is not defined. Possible causes are the division of zero by zero, the subtraction of infinite from infinite and the representation of missing values.

Example:

Setting command: `SENSe:LIST:FREQ MAXimum`

Query: `SENS:LIST:FREQ?`

Response: `3.5E9`

Queries for special numeric values

The numeric values associated to `MAXimum/MINimum/DEFault` can be queried by adding the corresponding mnemonic after the quotation mark.

Example: `SENSe:LIST:FREQ? MAXimum`

Returns the maximum numeric value as a result.

B.2.3.3 Boolean Parameters

Boolean parameters represent two states. The "ON" state (logically true) is represented by "ON" or a numeric value 1. The "OFF" state (logically untrue) is represented by "OFF" or the numeric value 0. The numeric values are provided as the response for a query.

Example:

Setting command: `HCOPy:DEV:COL ON`

Query: `HCOPy:DEV:COL?`

Response: `1`

B.2.3.4 Text Parameters

Text parameters observe the syntactic rules for mnemonics, i.e. they can be entered using a short or long form. Like any parameter, they have to be separated from the header by a white space. In the response to a query, the short form of the text is provided.

Example:

Setting command: `HCOPy:PAGE:ORIentation LANDscape`

Query: `HCOP:PAGE:ORI?`

Response: `LAND`

B.2.3.5 Character Strings

Strings must always be entered in quotation marks (' or ").

Example:

```
HCOP:ITEM:LABel "Test1"
HCOP:ITEM:LABel 'Test1'
```

B.2.3.6 Block Data

Block data is a format which is suitable for the transmission of large amounts of data. For example, a command using a block data parameter has the following structure:

```
FORMat:READings:DATA #45168xxxxxxxx
```

The ASCII character # introduces the data block. The next number indicates how many of the following digits describe the length of the data block. In the example the 4 following digits indicate the length to be 5168 bytes. The data bytes follow. During the transmission of these data bytes all end or other control signs are ignored until all bytes are transmitted.

#0 specifies a data block of indefinite length. The use of the indefinite format requires a `NL^END` message to terminate the data block. This format is useful when the length of

the transmission is not known or if speed or other considerations prevent segmentation of the data into blocks of definite length.

B.2.4 Overview of Syntax Elements

The following tables provide an overview of the syntax elements and special characters.

Table B-2: Syntax elements

:	The colon separates the mnemonics of a command.
;	The semicolon separates two commands of a command line. It does not alter the path.
,	The comma separates several parameters of a command.
?	The question mark forms a query.
*	The asterisk marks a common command.
' ' "	Quotation marks introduce a string and terminate it (both single and double quotation marks are possible).
#	The hash symbol introduces binary, octal, hexadecimal and block data. ● Binary: #B10110 ● Octal: #O7612 ● Hexa: #HF3A7 ● Block: #21312
	A "white space" (ASCII-Code 0 to 9, 11 to 32 decimal, e.g. blank) separates the header from the parameters.

Table B-3: Special characters

|	**Parameters**
	A vertical stroke in parameter definitions indicates alternative possibilities in the sense of "or". The effect of the command differs, depending on which parameter is used.
	Example:
	Definition:`HCOPy:PAGE:ORIentation LANDscape | PORTrait`
	Command `HCOP:PAGE:ORI LAND` specifies landscape orientation
	Command `HCOP:PAGE:ORI PORT` specifies portrait orientation
	Mnemonics
	A selection of mnemonics with an identical effect exists for several commands. These mnemonics are indicated in the same line; they are separated by a vertical stroke. Only one of these mnemonics needs to be included in the header of the command. The effect of the command is independent of which of the mnemonics is used.
	Example:
	Definition`SENSE:BANDwidth|BWIDth[:RESolution] <numeric_value>`
	The two following commands with identical meaning can be created:
	`SENS:BAND:RES 1`
	`SENS:BWID:RES 1`

[ ]	Mnemonics in square brackets are optional and may be inserted into the header or omitted. Example: `HCOPy[:IMMediate]` `HCOP:IMM` is equivalent to `HCOP`

{ }	Parameters in curly brackets are optional and can be inserted once or several times, or omitted.
	Example: `SENSe:LIST:FREQuency <numeric_value>{,<numeric_value>}`
	The following are valid commands:
	`SENS:LIST:FREQ 10`
	`SENS:LIST:FREQ 10,20`
	`SENS:LIST:FREQ 10,20,30,40`

B.2.5 Structure of a Command Line

A command line may consist of one or several commands. It is terminated by one of the following:

- <New Line>
- <New Line> with EOI
- EOI together with the last data byte

Several commands in a command line must be separated by a semicolon ";".

Example:

```
MMEM:COPY "Test1","MeasurementXY";:HCOP:ITEM ALL
```

This command line contains two commands. The first command belongs to the MMEM system, the second command belongs to the HCOP system. If the next command belongs to a different command system, the semicolon is followed by a colon.

Example:

`HCOP:ITEM ALL;:HCOP:IMM`

This command line contains two commands. Both commands are part of the `HCOP` command system, i.e. they have one level in common.

If the successive commands belong to the same system, having one or several levels in common, the command line can be abbreviated. When abbreviating the command line, the second command begins with the level below HCOP. The colon after the semicolon is omitted. The abbreviated form of the command line reads as follows:

`HCOP:ITEM ALL;IMM`

Example:

`HCOP:ITEM ALL`

`HCOP:IMM`

A new command line always begins with the complete path.

B.2.6 Responses to Queries

A query is defined for each setting command unless explicitly specified otherwise. It is formed by adding a question mark to the associated setting command. According to SCPI, the responses to queries are partly subject to stricter rules than in standard IEEE 488.2.

- The requested parameter is transmitted without a header.
  Example: `HCOP:PAGE:ORI?`, Response: `LAND`
- Maximum values, minimum values and all other quantities that are requested via a special text parameter are returned as numeric values.
  Example: `SENSe:FREQuency:STOP? MAX`, Response: `3.5E9`
- Numeric values are output without a unit. Physical quantities are referred to the basic units or to the units set using the `Unit` command. The response `3.5E9` in the previous example stands for 3.5 GHz.
- Truth values (Boolean values) are returned as `0` (for OFF) and `1` (for ON).
  Example:
  Setting command: `HCOPy:DEV:COL ON`
  Query: `HCOPy:DEV:COL?`
  Response: `1`
- Text (character data) is returned in a short form.
  Example:
  Setting command: `HCOPy:PAGE:ORIentation LANDscape`
  Query: `HCOP:PAGE:ORI?`
  Response: `LAND`
- Invalid numerical results
  In some cases, particularly when a result consists of multiple numeric values, invalid values are returned as 9.91E37 (not a number).

B.3 Command Sequence and Synchronization

IEEE 488.2 defines a distinction between overlapped (asynchronous) and sequential commands:

- A sequential command finishes executing before the next command starts executing. Commands that are processed quickly are usually implemented as sequential commands.
- An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. Usually, overlapping commands take longer to process and allow the program to do other tasks while being executed. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially. This method is called synchronization between the controller and the instrument.

As a rule, send commands and queries in different program messages, i.e. in separate command lines.

Do not combine queries with commands that affect the queried value in one program message because the response to the query is not predictable.

The following messages always return correct results:

`:CHAN:SCAL 0.01;POS 1`

`:CHAN:SCAL?`

Result: `0.01` (10 mV/div)

Reason: Setting commands within one command line, even though they are implemented as sequential commands, are not necessarily serviced in the order in which they have been received.

For further information, refer to:

- rohde-schwarz.com/rckb: Rohde & Schwarz web page that provides information on instrument drivers and remote control.
- "Automatic Measurement Control - A tutorial on SCPI and IEEE 488.2" from John M. Pieper (R&S order number 0002.3536.00). The book offers detailed information on concepts and definitions of SCPI.

B.3.1 Preventing Overlapping Execution

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used. All three commands cause a certain action only to be carried out after the hardware has been set. The controller can be forced to wait for the corresponding action to occur.

*Table B-4: Synchronization using *OPC, *OPC? and *WAI*

Com-mand	Action	Programming the controller
*OPC	Sets the Operation Complete bit in the ESR after all previous commands have been executed.	• Setting bit 0 in the ESE • Setting bit 5 in the SRE • Waiting for service request (SRQ)
*OPC?	Stops command processing until 1 is returned. This occurs when all pending operations are completed.	Send *OPC? directly after the command whose processing must be terminated before other commands can be executed.
*WAI	Stops further command processing until all commands sent before *WAI have been executed.	Send *WAI directly after the command whose processing must be terminated before other commands are executed.

Command synchronization using `*WAI` or `*OPC?` is a good choice if the overlapped command takes only little time to process. The two synchronization commands simply block overlapped execution of the command. Append the synchronization command to the overlapping command, for example:

```
SINGle; *OPC?
```

For time consuming overlapped commands, you can allow the controller or the instrument to do other useful work while waiting for command execution. Use one of the following methods:

*OPC with a service request

1. Set the OPC mask bit (bit no. 0) in the ESE: *ESE 1
2. Set bit no. 5 in the SRE: `*SRE 32` to enable ESB service request.
3. Send the overlapped command with `*OPC` .
4. Wait for a service request.

   The service request indicates that the overlapped command has finished.

*OPC? with a service request

1. Set bit no. 4 in the SRE: `*SRE 16` to enable MAV service request.
2. Send the overlapped command with `*OPC?`.
3. Wait for a service request.

   The service request indicates that the overlapped command has finished.

Event status register (ESE)

1. Set the OPC mask bit (bit no. 0) in the ESE: `*ESE 1`
2. Send the overlapped command without `*OPC`, `*OPC?` or `*WAI`.

3. Poll the operation complete state periodically (with a timer) using the sequence: `*OPC; *ESR?`

   A return value (LSB) of 1 indicates that the overlapped command has finished.

B.4 General Programming Recommendations

Initial instrument status before changing settings

Manual operation is designed for maximum possible operating convenience. In contrast, the priority of remote control is the "predictability" of the instrument status. Thus, when a command attempts to define incompatible settings, the command is ignored and the instrument status remains unchanged, i.e. other settings are not automatically adapted. Therefore, control programs should always define an initial instrument status (e.g. using the *RST command) and then implement the required settings.

Command sequence

As a general rule, send commands and queries in different program messages. Otherwise, the result of the query may vary depending on which operation is performed first (see also Preventing Overlapping Execution).

Reacting to malfunctions

The service request is the only possibility for the instrument to become active on its own. Each controller program should instruct the instrument to initiate a service request in case of malfunction. The program should react appropriately to the service request.

Error queues

The error queue should be queried after every service request in the controller program as the entries describe the cause of an error more precisely than the status registers. Especially in the test phase of a controller program the error queue should be queried regularly since faulty commands from the controller to the instrument are recorded there as well.

C Remote Control - Status Reporting System

The status reporting system stores all information on the current operating state of the instrument, and on errors which have occurred. This information is stored in the status registers and in the error queue. Both can be queried via GPIB bus or LAN interface (`STATus`... commands).

C.1 Structure of a SCPI Status Register

Each standard SCPI register consists of 5 parts. Each part has a width of 16 bits and has different functions. The individual bits are independent of each other, i.e. each hardware status is assigned a bit number which is valid for all five parts. Bit 15 (the most significant bit) is set to zero for all parts. Thus the contents of the register parts can be processed by the controller as positive integers.

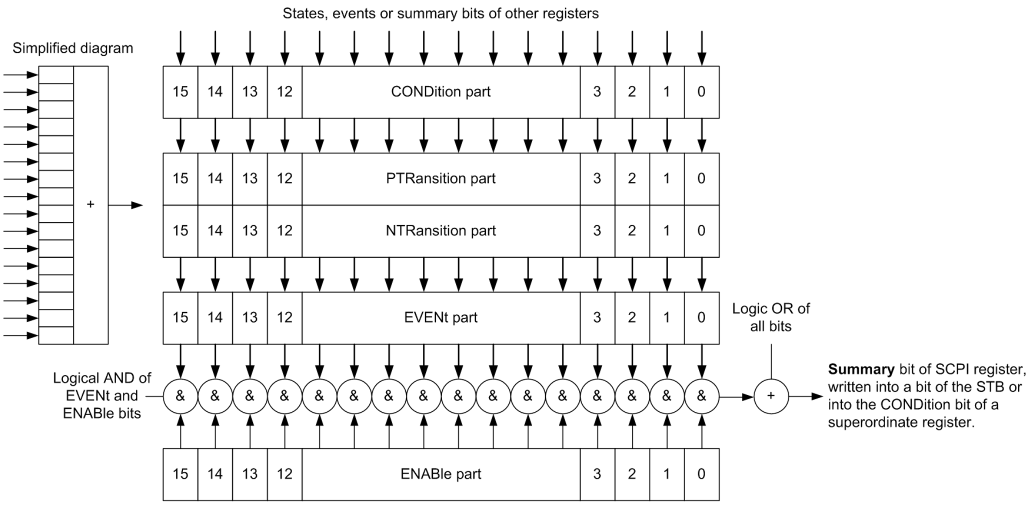


Figure C-1: The status-register model

Description of the five status register parts

The five parts of a SCPI register have different properties and functions:

- **CONDition**
  The `CONDition` part is written into directly by the hardware or the sum bit of the next lower register. Its contents reflect the current instrument status. This register part can only be read, but not written into or cleared. Its contents are not affected by reading.
- **PTRansition / NTRansition**
  The two transition register parts define which state transition of the `CONDition` part (none, 0 to 1, 1 to 0 or both) is stored in the `EVENt` part.
  The **Positive-TRansition** part acts as a transition filter. When a bit of the `CONDition` part is changed from 0 to 1, the associated `PTR` bit decides whether the `EVENt` bit is set to 1.

– PTR bit =1: the EVENt bit is set.
– PTR bit =0: the EVENt bit is not set.

This part can be written into and read as required. Its contents are not affected by reading.
The **Negative-TRansition** part also acts as a transition filter. When a bit of the CONDition part is changed from 1 to 0, the associated NTR bit decides whether the EVENt bit is set to 1.

– NTR bit =1: the EVENt bit is set.
– NTR bit =0: the EVENt bit is not set.

This part can be written into and read as required. Its contents are not affected by reading.

- **EVENt**
The EVENt part indicates whether an event has occurred since the last reading, it is the "memory" of the condition part. It only indicates events passed on by the transition filters. It is permanently updated by the instrument. This part can only be read by the user. Reading the register clears it. This part is often equated with the entire register.
- **ENABle**
The ENABle part determines whether the associated EVENt bit contributes to the sum bit (see below). Each bit of the EVENt part is "ANDed" with the associated ENABle bit (symbol '&'). The results of all logical operations of this part are passed on to the sum bit via an "OR" function (symbol '+').
ENABle bit = 0: the associated EVENt bit does not contribute to the sum bit
ENABle bit = 1: if the associated EVENt bit is "1", the sum bit is set to "1" as well.
This part can be written into and read by the user as required. Its contents are not affected by reading.

Sum bit

The sum bit is obtained from the EVENt and ENABle part for each register. The result is then entered into a bit of the CONDition part of the higher-order register.

The instrument automatically generates the sum bit for each register. Thus an event can lead to a service request throughout all levels of the hierarchy.

C.2 Hierarchy of Status Registers

As shown in the following figure, the status information is of hierarchical structure.

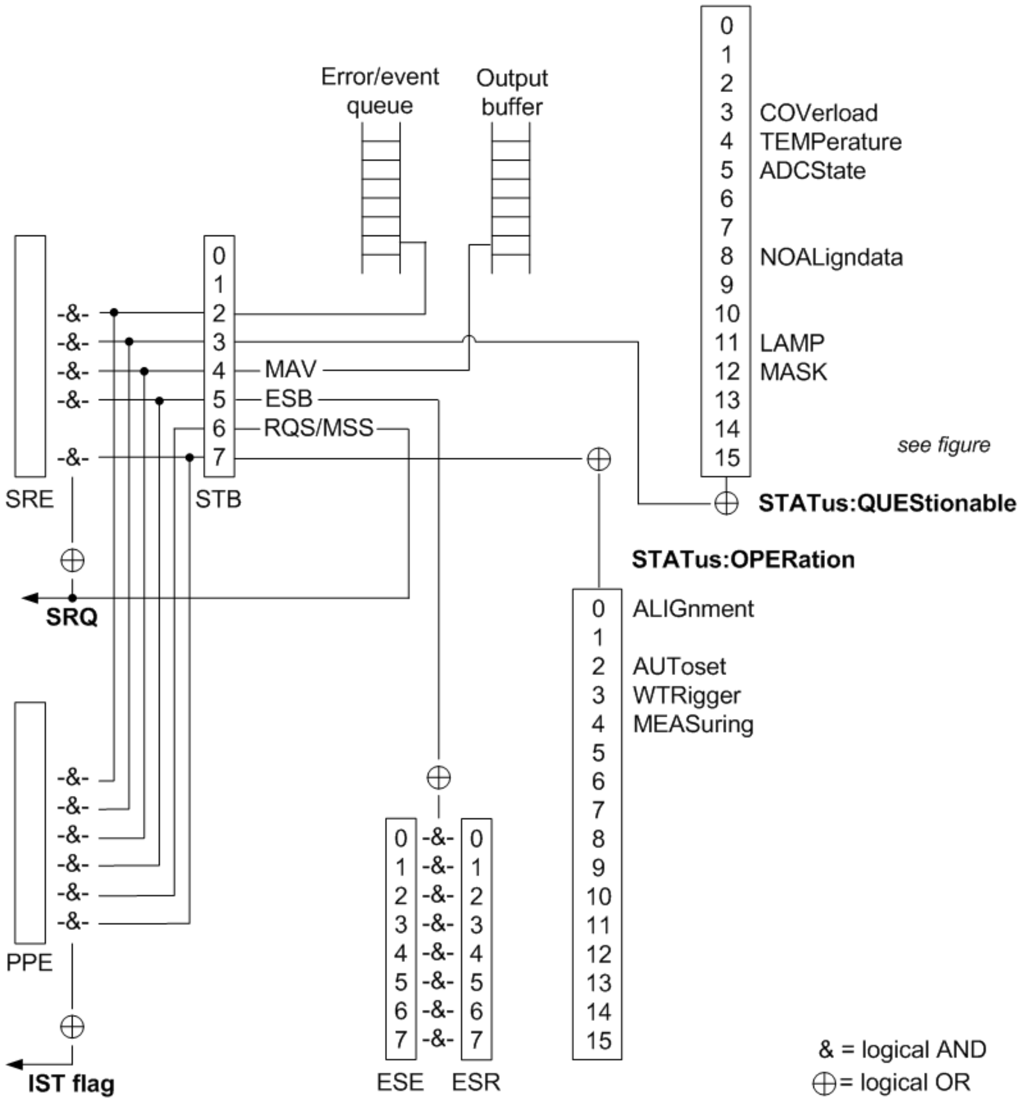


Figure C-2: Overview of the status registers hierarchy

- **STB, SRE**
  The STatus Byte (`STB`) register and its associated mask register Service Request Enable (`SRE`) form the highest level of the status reporting system. The `STB` provides a rough overview of the instrument status, collecting the information of the lower-level registers.
- **ESR, SCPI registers**
  The STB receives its information from the following registers:
  – The Event Status Register (`ESR`) with the associated mask register standard Event Status Enable (`ESE`).
  – The `STATus:OPERation` and `STATus:QUEStionable` registers which are defined by SCPI and contain detailed information on the instrument.
- **IST, PPE**

The IST flag ("Individual STatus"), like the SRQ, combines the entire instrument status in a single bit. The PPE fulfills the same function for the IST flag as the SRE for the service request.

- **Output buffer**
  The output buffer contains the messages the instrument returns to the controller. It is not part of the status reporting system but determines the value of the MAV bit in the STB and thus is represented in the overview.

All status registers have the same internal structure.

SRE, ESE

The service request enable register SRE can be used as ENABle part of the STB if the STB is structured according to SCPI. By analogy, the ESE can be used as the ENABle part of the ESR.

C.3 Contents of the Status Registers

In the following sections, the contents of the status registers are described in more detail.

C.3.1 Status Byte (STB) and Service Request Enable Register (SRE)

The STatus Byte (STB) is already defined in IEEE 488.2. It provides a rough overview of the instrument status by collecting the pieces of information of the lower registers. A special feature is that bit 6 acts as the sum bit of the remaining bits of the status byte.

The STB can thus be compared with the CONDition part of an SCPI register and assumes the highest level within the SCPI hierarchy.

The STB is read using the command *STB? or a serial poll.

The STatus Byte (STB) is linked to the Service Request Enable (SRE) register. Each bit of the STB is assigned a bit in the SRE. Bit 6 of the SRE is ignored. If a bit is set in the SRE and the associated bit in the STB changes from 0 to 1, a service request (SRQ) is generated. The SRE can be set using the command *SRE and read using the command *SRE?.

Table C-1: Meaning of the bits used in the status byte

Bit No.	Meaning
0...1	Not used
2	Error Queue not empty The bit is set when an entry is made in the error queue. If this bit is enabled by the SRE, each entry of the error queue generates a service request. Thus an error can be recognized and specified in greater detail by polling the error queue. The poll provides an informative error message. This procedure is to be recommended since it considerably reduces the problems involved with remote control.

Bit No.	Meaning
3	QUEStionable status register summary bit The bit is set if an EVENt bit is set in the QUEStionable status register and the associated ENABle bit is set to 1. A set bit indicates a questionable instrument status, which can be specified in greater detail by querying the STATus:QUEStionable status register.
4	MAV bit (message available) The bit is set if a message is available in the output queue which can be read. This bit can be used to enable data to be automatically read from the instrument to the controller.
5	ESB bit Sum bit of the event status register. It is set if one of the bits in the event status register is set and enabled in the event status enable register. Setting of this bit indicates a serious error which can be specified in greater detail by polling the event status register.
6	MSS bit (master status summary bit) The bit is set if the instrument triggers a service request. This is the case if one of the other bits of this registers is set together with its mask bit in the service request enable register SRE.
7	STATus:OPERation status register summary bit The bit is set if an EVENt bit is set in the OPERation status register and the associated ENABle bit is set to 1. A set bit indicates that the instrument is just performing an action. The type of action can be determined by querying the STATus:OPERation status register.

C.3.2 IST Flag and Parallel Poll Enable Register (PPE)

As with the SRQ, the IST flag combines the entire status information in a single bit. It can be read by means of a parallel poll (see Chapter C.4.3, "Parallel Poll", on page 1995) or using the command *IST?.

The parallel poll enable register (PPE) determines which bits of the STB contribute to the IST flag. The bits of the STB are "ANDed" with the corresponding bits of the PPE, with bit 6 being used as well in contrast to the SRE. The IST flag results from the "ORing" of all results. The PPE can be set using commands *PRE and read using command *PRE?.

C.3.3 Event Status Register (ESR) and Event Status Enable Register (ESE)

The ESR is defined in IEEE 488.2. It can be compared with the EVENt part of a SCPI register. The event status register can be read out using command *ESR?.

The ESE corresponds to the ENABle part of a SCPI register. If a bit is set in the ESE and the associated bit in the ESR changes from 0 to 1, the ESB bit in the STB is set. The ESE register can be set using the command *ESE and read using the command *ESE?.

Table C-2: Meaning of the bits used in the event status register

Bit No.	Meaning
0	Operation Complete This bit is set on receipt of the command *OPC exactly when all previous commands have been executed.
1	Not used
2	Query Error This bit is set if either the controller wants to read data from the instrument without having sent a query, or if it does not fetch requested data and sends new instructions to the instrument instead. The cause is often a query which is faulty and hence cannot be executed.
3	Device-dependent Error This bit is set if a device-dependent error occurs. An error message with a number between -300 and -399 or a positive error number, which denotes the error in greater detail, is entered into the error queue.
4	Execution Error This bit is set if a received command is syntactically correct but cannot be performed for other reasons. An error message with a number between -200 and -300, which denotes the error in greater detail, is entered into the error queue.
5	Command Error This bit is set if a command is received, which is undefined or syntactically incorrect. An error message with a number between -100 and -200, which denotes the error in greater detail, is entered into the error queue.
6	User Request This bit is set when the instrument is switched over to manual control.
7	Power On (supply voltage on) This bit is set on switching on the instrument.

C.3.4 STATus:OPERation Register

In the `CONDition` part, this register contains information on which actions the instrument is executing. In the `EVENt` part, it contains information on which actions the instrument has executed since the last reading.

It can be read using the commands `STATus:OPERation:CONDition?` or `STATus:OPERation[:EVENt]?`, see also Chapter 17.22.1, "STATus:OPERation Register", on page 1948.

Table C-3: Bits in the STATus:OPERation register

Bit No.	Meaning
0	`ALIGnment` This bit is set as long as the instrument is performing a self-alignment.
1	Not used
2	`AUToset` This bit is set while the instrument is performing an auto setup.

Bit No.	Meaning
3	`WTRIgger` The wait for trigger status bit indicates that the instrument is ready to trigger, and the pre-trigger time is expired. The bit is set if the instrument did not trigger for more than 10 ms. The bit is only valid if the trigger event is initiated by the user, for example, using an external generator. If the instrument triggers on signals or if it is in auto trigger mode, the bit status is undefined.
4	`MEASuring` The bit is set as long as an acquisition - sampling and postprocessing - is running. In run continuous mode, the bit is always set.
5 - 15	Not used

C.3.5 STATus:QUEStionable Register

This register contains information about indefinite states which may occur if the unit is operated without meeting the specifications. It can be read using the commands `STATus:QUEStionable:CONDition?` and `STATus:QUEStionable[:EVENt]?`

The remote commands for the STATus:QUEStionable register are described in Chapter 17.22.2, "STATus:QUEStionable Registers", on page 1949.

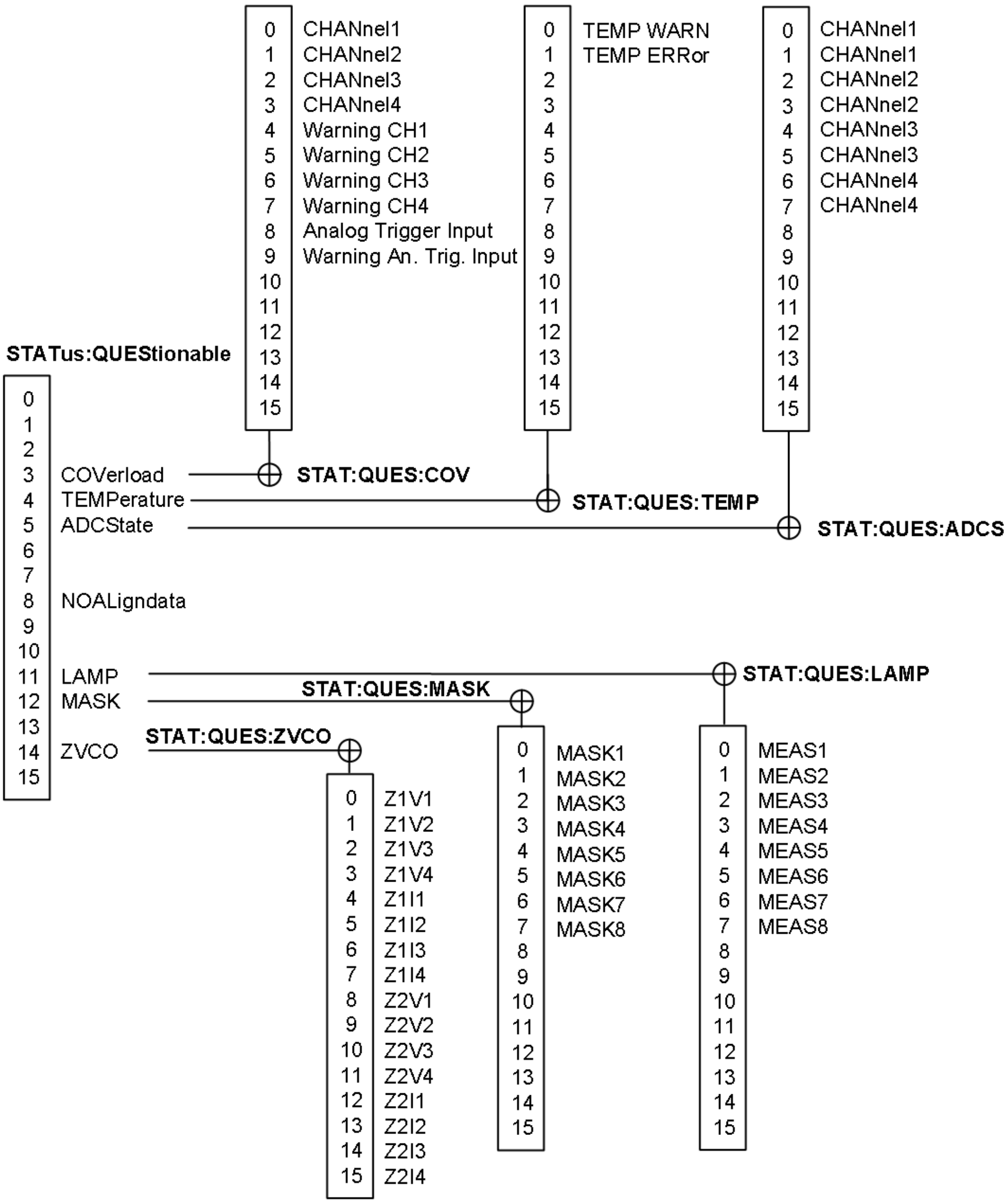


Figure C-3: Overview of the STATus:QUEStionable register

Table C-4: Bits in the STATus:QUEStionable register

Bit No.	Meaning
0 to 2	not used
3	`COVerload` This bit is set if a questionable channel overload occurs (see Chapter C.3.5.1, "STATus:QUEStionable:COVerload Register", on page 1991).
4	`TEMPerature` This bit is set if a questionable temperature occurs (see Chapter C.3.5.2, "STATus:QUEStionable:TEMPerature Register", on page 1992).

Bit No.	Meaning
5	ADCState The bit is set if the signal is clipped on the upper or lower edge of the screen -overflow of the ADC occurs (see Chapter C.3.5.3, "STATus:QUEStionable:ADCState Register", on page 1992).
6	Not used
7	Not used
8	NOALigndata This bit is set if no alignment data is available - the instrument is uncalibrated.
9 to 10	not used
11	LAMP (Low AMPlitude) This bit is set if the magnitude of the signal is too low to get reliable measurement results. See Chapter C.3.5.4, "STATus:QUEStionable:LAMP Register", on page 1992.
12	MASK This bit is set if a mask value is violated (see Chapter C.3.5.5, "STATus:QUEStionable:MASK Register", on page 1993
13	Not used
14	ZVCO: This bit is set if a channel overload occurs at one or more input channels of the R&S RT-ZVC multi-channel power probe. See Chapter C.3.5.6, "STATus:QUEStionable:ZVCO Register", on page 1993.
15	This bit is always 0.

C.3.5.1 STATus:QUEStionable:COVerload Register

This register contains all information about overload of the channels. The bit is set if the assigned channel is overloaded, or if an overload risk occurred (overload warning).

Table C-5: Bits in the STATus:QUEStionable:COVerload register

Bit No.	Meaning
0	Overload on CHANnel1
1	Overload on CHANnel2
2	Overload on CHANnel3
3	Overload on CHANnel4
4	Overload warning for CHANnel1
5	Overload warning for CHANnel2
6	Overload warning for CHANnel3
7	Overload warning for CHANnel4
8	Overload on external trigger input
9	Overload warning for external trigger input

C.3.5.2 STATus:QUEStionable:TEMPerature Register

This register contains information about the instrument's temperature.

Table C-6: Bits in the STATus:QUEStionable:TEMPerature register

Bit No.	Meaning
0	`TEMP WARN` This bit is set if a temperature warning on channel 1, 2, 3 or 4 occurred.
1	`TEMP ERRor` This bit is set if a temperature error on channel 1, 2, 3 or 4 occurred.

C.3.5.3 STATus:QUEStionable:ADCState Register

This register contains all information about overflow of the ADC.

The bit is set if the assigned channel signal is clipped on the upper or lower edge of the screen. In this case, the signal does not fit in the range of the ADC and overflow occurs.

Table C-7: Bits in the STATus:QUEStionable:ADCState register

Bit No.	Meaning
0	`CHANnel1`, clipping on the upper limit
1	`CHANnel1`, clipping on the lower limit
2	`CHANnel2`, clipping on the upper limit
3	`CHANnel2`, clipping on the lower limit
4	`CHANnel3`, clipping on the upper limit
5	`CHANnel3`, clipping on the lower limit
6	`CHANnel4`, clipping on the upper limit
7	`CHANnel4`, clipping on the lower limit

C.3.5.4 STATus:QUEStionable:LAMP Register

The LAMP (Low AMPlitude) bit is set if the magnitude of the signal is too low to get reliable measurement results.

Table C-8: Bits in the STATus:QUEStionable:LAMP register

Bit No.	Meaning
0	`MEAS1`
1	`MEAS2`
2	`MEAS3`
3	`MEAS4`
4	`MEAS5`

Bit No.	Meaning
5	MEAS6
6	MEAS7
7	MEAS8

C.3.5.5 STATus:QUEStionable:MASK Register

This register contains information about the violation of masks. This bit is set if the assigned mask is violated.

Table C-9: Bits in the STATus:QUEStionable:MASK register

Bit No.	Meaning
0	MASK1
1	MASK2
2	MASK3
3	MASK4
4	MASK5
5	MASK6
6	MASK7
7	MASK8

C.3.5.6 STATus:QUEStionable:ZVCO Register

This register contains all information about overload of the R&S RT-ZVC input channels. The bit is set if the assigned channel is overloaded.

Table C-10: Bits in the STATus:QUEStionable:ZVCO register

Bit No.	Meaning
0	Overload on Z1V1
1	Overload on Z1V2
2	Overload on Z1V3
3	Overload on Z1V4
4	Overload on Z1I1
5	Overload on Z1I2
6	Overload on Z1I3
7	Overload on Z1I4
8	Overload on Z2V1
9	Overload on Z2V2

Bit No.	Meaning
10	Overload on Z2V3
11	Overload on Z2V4
12	Overload on Z2I1
13	Overload on Z2I2
14	Overload on Z2I3
15	Overload on Z2I4

C.4 Application of the Status Reporting System

The purpose of the status reporting system is to monitor the status of one or several devices in a measuring system. To do this and react appropriately, the controller must receive and evaluate the information of all devices. The following standard methods are used:

- **Service request** (SRQ) initiated by the instrument
- **Serial poll** of all devices in the bus system, initiated by the controller to find out who sent an SRQ and why
- **Parallel poll** of all devices
- Query of a **specific instrument status** by commands
- Query of the **error queue**

C.4.1 Service Request

Under certain circumstances, the instrument can send a service request (SRQ) to the controller. Usually this service request initiates an interrupt at the controller, to which the control program can react appropriately. As evident from Figure C-2, an SRQ is always initiated if one or several of bits 2, 3, 4, 5 or 7 of the status byte are set and enabled in the SRE. Each of these bits combines the information of a further register, the error queue or the output buffer. The ENABle parts of the status registers can be set such that arbitrary bits in an arbitrary status register initiate an SRQ. To make use of the possibilities of the service request effectively, all bits should be set to "1" in enable registers SRE and ESE.

The SRQ is the only possibility for the instrument to become active on its own. Each controller program should cause the instrument to initiate a service request if errors occur. The program should react appropriately to the service request.

C.4.2 Serial Poll

In a serial poll, just as with command *STB, the status byte of an instrument is queried. However, the query is realized via interface messages and is thus clearly faster.

The serial poll method is defined in IEEE 488.1 and used to be the only standard possibility for different instruments to poll the status byte. The method also works for instruments which do not adhere to SCPI or IEEE 488.2.

The serial poll is mainly used to obtain a fast overview of the state of several instruments connected to the controller.

C.4.3 Parallel Poll

In a parallel poll, up to eight instruments are simultaneously requested by the controller using a single command to transmit 1 bit of information each on the data lines, i.e., to set the data line allocated to each instrument to a logical "0" or "1".

In addition to the SRE register, which determines the conditions under which an SRQ is generated, there is a Parallel Poll Enable register (PPE) which is ANDed with the STB bit by bit, considering bit 6 as well. This register is ANDed with the STB bit by bit, considering bit 6 as well. The results are ORed, the result is possibly inverted and then sent as a response to the parallel poll of the controller. The result can also be queried without parallel poll using the command `*IST?`.

The instrument first has to be set for the parallel poll using the command `PPC`. This command allocates a data line to the instrument and determines whether the response is to be inverted. The parallel poll itself is executed using `PPE`.

The parallel poll method is mainly used to find out quickly which one of the instruments connected to the controller has sent a service request. To this effect, SRE and PPE must be set to the same value.

C.4.4 Query of an instrument status

Each part of any status register can be read using queries. There are two types of commands:

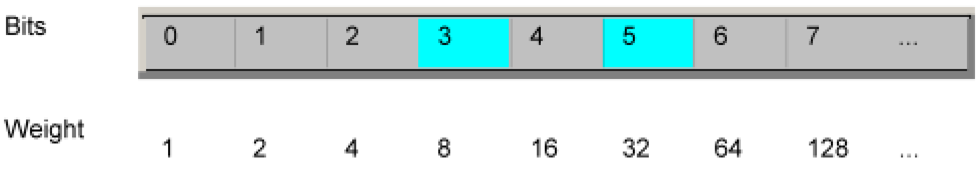
- The common commands `*ESR?`, `*IDN?`, `*IST?`, `*STB?` query the higher-level registers.
- The commands of the `STATus` system query the SCPI registers (`STATus:QUEStionable`...)

The returned value is always a decimal number that represents the bit pattern of the queried register. This number is evaluated by the controller program.

Queries are usually used after an SRQ in order to obtain more detailed information on the cause of the SRQ.

C.4.4.1 Decimal representation of a bit pattern

The STB and ESR registers contain 8 bits, the SCPI registers 16 bits. The contents of a status register are specified and transferred as a single decimal number. To make this possible, each bit is assigned a weighted value. The decimal number is calculated as the sum of the weighted values of all bits in the register that are set to 1.



Example:

The decimal value 40 = 32 + 8 indicates that bits no. 3 and 5 in the status register (e.g. the `QUEStionable` status summary bit and the `ESB` bit in the `STatus` Byte) are set.

C.4.5 Error Queue

Each error state in the instrument leads to an entry in the error queue. The entries of the error queue are detailed plain text error messages that can be looked up in the Error Log or queried via remote control using `SYSTem:ERRor[:NEXT]?`. Each call of `SYSTem:ERRor[:NEXT]?` provides one entry from the error queue. If no error messages are stored there any more, the instrument responds with 0, "No error".

The error queue should be queried after every SRQ in the controller program as the entries describe the cause of an error more precisely than the status registers. Especially in the test phase of a controller program the error queue should be queried regularly since faulty commands from the controller to the instrument are recorded there as well.

C.5 Reset Values of the Status Reporting System

The following table contains the different commands and events causing the status reporting system to be reset. None of the commands, except `*RST` and `SYSTem:PRESet`, influence the functional instrument settings. In particular, `DCL` does not change the instrument settings.

Table C-11: Resets of the status reporting system

Event	**Switching on supply voltage Power-On-Status-Clear**		**DCL, SDC (Device Clear, Selected Device Clear)**	***RST or SYSTem:PRESet**	**STATus:PRESet**	***CLS**
Effect	**0**	**1**				
Clear STB, ESR	-	yes	-	-	-	yes
Clear SRE, ESE	-	yes	-	-	-	-
Clear PPE	-	yes	-	-	-	-
Clear EVENt parts of the registers	-	yes	-	-	-	yes

Event	Switching on supply voltage Power-On-Status-Clear		DCL, SDC (Device Clear, Selected Device Clear)	*RST or SYS-Tem:PRESet	STA-Tus:PRE-Set	*CLS
Effect	**0**	**1**				
Clear ENABle parts of all OPERation and QUEStionable registers; Fill ENABle parts of all other registers with "1".	-	yes	-	-	yes	-
Fill PTRansition parts with "1"; Clear NTRansition parts	-	yes	-	-	yes	-
Clear error queue	yes	yes	-	-	-	yes
Clear output buffer	yes	yes	yes	1)	1)	1)
Clear command processing and input buffer	yes	yes	yes	-	-	-
1) The first command in a command line that immediately follows a <PROGRAM MESSAGE TERMINATOR> clears the output buffer.						

List of commands

*CAL?...1040
*CLS...1040
*ESE...1041
*ESR?...1041
*IDN?...1041
*IST?...1041
*OPC...1041
*OPT?...1042
*PCB...1042
*PRE...1042
*PSC...1042
*RCL...1043
*RST...1043
*SAV...1043
*SRE...1043
*STB?...1043
*TRG...1044
*TST?...1044
*WAI...1044
ACQuire:ARESet:COUNt...1925
ACQuire:ARESet:IMMediate...1078
ACQuire:ARESet:MODE...1924
ACQuire:ARESet:TIME...1925
ACQuire:AVAilable?...1194
ACQuire:COUNt...1078
ACQuire:CURRent?...1173
ACQuire:DRESolution?...1873
ACQuire:POINts:AADJust...1073
ACQuire:POINts:ARATe?...1074
ACQuire:POINts:AUTO...1073
ACQuire:POINts:DVALue?...1874
ACQuire:POINts:MAXimum...1074
ACQuire:POINts:ZVALue?...1118
ACQuire:POINts[:VALue]...1075
ACQuire:RESolution...1075
ACQuire:SEGMented:AUToreplay...1079
ACQuire:SEGMented:MAX...1079
ACQuire:SEGMented:STATe...1078
ACQuire:SRATe...1074
ACQuire:SRReal...1074
ACQuire:ZRESolution?...1118
AUToscale...1073
BUS<m>:ARINc:BRMode...1595
BUS<m>:ARINc:BRValue...1595
BUS<m>:ARINc:MAXGap:BITS...1596
BUS<m>:ARINc:MAXGap:SELect...1595
BUS<m>:ARINc:MINGap:BITS...1596

BUS<m>:ARINc:MINGap:SELect....1596
BUS<m>:ARINc:POLarity....1596
BUS<m>:ARINc:PRESet....1597
BUS<m>:ARINc:SOURce....1595
BUS<m>:ARINc:THReshold:HIGH....1597
BUS<m>:ARINc:THReshold:LOW....1597
BUS<m>:ARINc:WCOunt?....1601
BUS<m>:ARINc:WORD<n>:DATA?....1601
BUS<m>:ARINc:WORD<n>:LABel?....1602
BUS<m>:ARINc:WORD<n>:PATTern?....1602
BUS<m>:ARINc:WORD<n>:SDI?....1602
BUS<m>:ARINc:WORD<n>:SSM?....1603
BUS<m>:ARINc:WORD<n>:STARt?....1603
BUS<m>:ARINc:WORD<n>:STATe?....1603
BUS<m>:ARINc:WORD<n>:STOP?....1604
BUS<m>:ARINc:WORD<n>:SYMBol?....1604
BUS<m>:CAN:BITRate....1452
BUS<m>:CAN:DATA:SOURce....1450
BUS<m>:CAN:DATA:THReshold....1451
BUS<m>:CAN:FCOunt?....1463
BUS<m>:CAN:FDATa:DBITrate....1453
BUS<m>:CAN:FDATa:ENABle....1452
BUS<m>:CAN:FDATa:FRAMe<n>:SCValue?....1470
BUS<m>:CAN:FDATa:FRAMe<n>:STANdard?....1463
BUS<m>:CAN:FDATa:JWIDth....1454
BUS<m>:CAN:FDATa:PSTandard....1451
BUS<m>:CAN:FDATa:SAMPlepoint....1453
BUS<m>:CAN:FDATa:T1Segment....1453
BUS<m>:CAN:FDATa:T2Segment....1454
BUS<m>:CAN:FRAMe<n>:ACKState?....1466
BUS<m>:CAN:FRAMe<n>:ACKValue?....1467
BUS<m>:CAN:FRAMe<n>:BITRate?....1467
BUS<m>:CAN:FRAMe<n>:BSEPosition?....1469
BUS<m>:CAN:FRAMe<n>:BYTE<o>:STATe?....1470
BUS<m>:CAN:FRAMe<n>:BYTE<o>:VALue?....1471
BUS<m>:CAN:FRAMe<n>:CSSTate?....1466
BUS<m>:CAN:FRAMe<n>:CSValue?....1468
BUS<m>:CAN:FRAMe<n>:DATA?....1466
BUS<m>:CAN:FRAMe<n>:DLCState?....1466
BUS<m>:CAN:FRAMe<n>:DLCValue?....1468
BUS<m>:CAN:FRAMe<n>:FERCause?....1469
BUS<m>:CAN:FRAMe<n>:IDSTate?....1466
BUS<m>:CAN:FRAMe<n>:IDTYpe?....1468
BUS<m>:CAN:FRAMe<n>:IDValue?....1469
BUS<m>:CAN:FRAMe<n>:NDBYtes?....1465
BUS<m>:CAN:FRAMe<n>:SDATa?....1488
BUS<m>:CAN:FRAMe<n>:SDEXport?....1470
BUS<m>:CAN:FRAMe<n>:STARt?....1465
BUS<m>:CAN:FRAMe<n>:STATus?....1464
BUS<m>:CAN:FRAMe<n>:STOP?....1465

BUS<m>:CAN:FRAMe<n>:SYMBol?....1465
BUS<m>:CAN:FRAMe<n>:TYPE?....1466
BUS<m>:CAN:JWIDth....1454
BUS<m>:CAN:SAMPlepoint....1453
BUS<m>:CAN:T1Segment....1453
BUS<m>:CAN:T2Segment....1454
BUS<m>:CAN:TECHnology....1452
BUS<m>:CAN:TYPE....1451
BUS<m>:CMSB:ADDFrame....1705
BUS<m>:CMSB:BITRate:ENABle....1704
BUS<m>:CMSB:BITRate:VALue....1704
BUS<m>:CMSB:CLR....1705
BUS<m>:CMSB:CODing....1695
BUS<m>:CMSB:EXRBits....1708
BUS<m>:CMSB:FCOunt?....1705
BUS<m>:CMSB:FILTer:BIT....1716
BUS<m>:CMSB:FILTer:DMAX....1717
BUS<m>:CMSB:FILTer:DMIN....1717
BUS<m>:CMSB:FILTer:DOPerator....1717
BUS<m>:CMSB:FILTer:ENABle....1718
BUS<m>:CMSB:FILTer:ERENable....1718
BUS<m>:CMSB:FILTer:ERRor<n>:ENABle....1718
BUS<m>:CMSB:FILTer:FIENable....1718
BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle....1719
BUS<m>:CMSB:FILTer:FRENable....1719
BUS<m>:CMSB:FILTer:IMAX....1719
BUS<m>:CMSB:FILTer:IMIN....1720
BUS<m>:CMSB:FILTer:IOPerator....1720
BUS<m>:CMSB:FRAMe<n>:APPend....1706
BUS<m>:CMSB:FRAMe<n>:CCOunt?....1706
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount....1706
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder....1708
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN....1708
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition....1707
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB....1708
BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat....1707
BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME....1706
BUS<m>:CMSB:FRAMe<n>:TYPE....1705
BUS<m>:CMSB:GAPTime:ENABle....1704
BUS<m>:CMSB:GAPTime:VALue....1704
BUS<m>:CMSB:LOAD....1709
BUS<m>:CMSB:MANChester:CPHase....1698
BUS<m>:CMSB:MANChester:DATA....1696
BUS<m>:CMSB:MANChester:POLarity....1696
BUS<m>:CMSB:MANChester:THReshold:COUPling....1697
BUS<m>:CMSB:MANChester:THReshold:HIGH....1696
BUS<m>:CMSB:MANChester:THReshold:LOW....1697
BUS<m>:CMSB:MANChester:THReshold:PRESet....1697
BUS<m>:CMSB:NRZ:CLCK....1698
BUS<m>:CMSB:NRZ:CPHase....1700

BUS<m>:CMSB:NRZ:CPOLarity....1699
BUS<m>:CMSB:NRZ:DATA....1699
BUS<m>:CMSB:NRZ:ENAPolarity....1701
BUS<m>:CMSB:NRZ:ENBLe....1700
BUS<m>:CMSB:NRZ:HYSTeresis:CLCK....1701
BUS<m>:CMSB:NRZ:HYSTeresis:DATA....1702
BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe....1702
BUS<m>:CMSB:NRZ:IDLPolarity....1699
BUS<m>:CMSB:NRZ:POLarity....1701
BUS<m>:CMSB:NRZ:THReshold:CLCK....1701
BUS<m>:CMSB:NRZ:THReshold:COUPling....1703
BUS<m>:CMSB:NRZ:THReshold:DATA....1702
BUS<m>:CMSB:NRZ:THReshold:ENBLe....1702
BUS<m>:CMSB:NRZ:THReshold:PRESet....1703
BUS<m>:CMSB:RCOunt?....1722
BUS<m>:CMSB:RESult<n>:CCOunt?....1724
BUS<m>:CMSB:RESult<n>:CELL<o>:NAME?....1725
BUS<m>:CMSB:RESult<n>:CELL<o>:STARt?....1725
BUS<m>:CMSB:RESult<n>:CELL<o>:STATe?....1726
BUS<m>:CMSB:RESult<n>:CELL<o>:STOP?....1725
BUS<m>:CMSB:RESult<n>:CELL<o>:VALue?....1726
BUS<m>:CMSB:RESult<n>:CONE?....1724
BUS<m>:CMSB:RESult<n>:CTHRee?....1724
BUS<m>:CMSB:RESult<n>:CTWO?....1724
BUS<m>:CMSB:RESult<n>:STARt?....1723
BUS<m>:CMSB:RESult<n>:STATe?....1722
BUS<m>:CMSB:RESult<n>:STOP?....1723
BUS<m>:CMSB:RESult<n>:TYPE?....1723
BUS<m>:CMSB:SAVE....1709
BUS<m>:CXPI:BITRate:ENABle....1836
BUS<m>:CXPI:BITRate:VALue....1836
BUS<m>:CXPI:DORD....1837
BUS<m>:CXPI:HYSTeresis....1837
BUS<m>:CXPI:IBS:MAX....1838
BUS<m>:CXPI:IBS:MIN....1838
BUS<m>:CXPI:IFS:MAX....1839
BUS<m>:CXPI:IFS:MIN....1839
BUS<m>:CXPI:POLarity....1837
BUS<m>:CXPI:RESult:BITRate?....1836
BUS<m>:CXPI:RESult:FCOunt?....1848
BUS<m>:CXPI:RESult:FRAMe<n>:DATA?....1848
BUS<m>:CXPI:RESult:FRAMe<n>:DLCV?....1848
BUS<m>:CXPI:RESult:FRAMe<n>:STARt?....1849
BUS<m>:CXPI:RESult:FRAMe<n>:STATe?....1849
BUS<m>:CXPI:RESult:FRAMe<n>:STOP?....1849
BUS<m>:CXPI:RESult:FRAMe<n>:TYPE?....1850
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:STATus?....1850
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:TYPE?....1850
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:VALue?....1851
BUS<m>:CXPI:SDATa....1838

BUS<m>:CXPI:THReshold....1837
BUS<m>:ETHernet:BITRate....1613
BUS<m>:ETHernet:POLarity....1611
BUS<m>:ETHernet:PRESet....1612
BUS<m>:ETHernet:SOURce....1611
BUS<m>:ETHernet:THReshold:HIGH....1612
BUS<m>:ETHernet:THReshold:LOW....1612
BUS<m>:ETHernet:VARiant....1611
BUS<m>:ETHernet:WCOunt?....1619
BUS<m>:ETHernet:WORD<n>:BITRate?....1623
BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?....1623
BUS<m>:ETHernet:WORD<n>:CRC?....1622
BUS<m>:ETHernet:WORD<n>:DATA?....1622
BUS<m>:ETHernet:WORD<n>:DESTaddress?....1621
BUS<m>:ETHernet:WORD<n>:DSYMbol?....1622
BUS<m>:ETHernet:WORD<n>:FTYPe?....1619
BUS<m>:ETHernet:WORD<n>:NUMWords?....1624
BUS<m>:ETHernet:WORD<n>:SRCaddress?....1621
BUS<m>:ETHernet:WORD<n>:SSYMbol?....1623
BUS<m>:ETHernet:WORD<n>:STARt?....1620
BUS<m>:ETHernet:WORD<n>:STATe?....1620
BUS<m>:ETHernet:WORD<n>:STOP?....1620
BUS<m>:ETHernet:WORD<n>:TYPE?....1621
BUS<m>:FAUToset....1383
BUS<m>:FLXRay:BITRate....1523
BUS<m>:FLXRay:CHTYpe....1523
BUS<m>:FLXRay:FCOunt?....1533
BUS<m>:FLXRay:FRAMe<n>:ADID?....1536
BUS<m>:FLXRay:FRAMe<n>:CSSTate?....1537
BUS<m>:FLXRay:FRAMe<n>:CSValue?....1537
BUS<m>:FLXRay:FRAMe<n>:CYCount?....1536
BUS<m>:FLXRay:FRAMe<n>:DATA?....1535
BUS<m>:FLXRay:FRAMe<n>:FCSTate?....1537
BUS<m>:FLXRay:FRAMe<n>:FCValue?....1538
BUS<m>:FLXRay:FRAMe<n>:FLAGs?....1535
BUS<m>:FLXRay:FRAMe<n>:PAYLength?....1536
BUS<m>:FLXRay:FRAMe<n>:STARt?....1534
BUS<m>:FLXRay:FRAMe<n>:STATus?....1533
BUS<m>:FLXRay:FRAMe<n>:STOP?....1534
BUS<m>:FLXRay:FRAMe<n>:SYMBol?....1534
BUS<m>:FLXRay:FRAMe<n>:TYPE?....1534
BUS<m>:FLXRay:POLarity....1523
BUS<m>:FLXRay:PRDiff....1522
BUS<m>:FLXRay:PRLogic....1522
BUS<m>:FLXRay:PRSingle....1522
BUS<m>:FLXRay:SEHB....1524
BUS<m>:FLXRay:SOURce<n>....1520
BUS<m>:FLXRay:SRCType....1520
BUS<m>:FLXRay:THData....1521
BUS<m>:FLXRay:THENable....1521

BUS<m>:FLXRay:THReshold<n>....1521
BUS<m>:FORMat....1384
BUS<m>:HBTO:ATTN....1634
BUS<m>:HBTO:FDAM....1635
BUS<m>:HBTO:FDAP....1635
BUS<m>:HBTO:FDIF....1635
BUS<m>:HBTO:FTYP....1636
BUS<m>:HBTO:MODE....1637
BUS<m>:HBTO:POLarity....1637
BUS<m>:HBTO:RDAM....1636
BUS<m>:HBTO:RDAP....1636
BUS<m>:HBTO:RDIF....1636
BUS<m>:HBTO:RESult:FCOunt?....1646
BUS<m>:HBTO:RESult:FRAMe<n>:BITRate?....1650
BUS<m>:HBTO:RESult:FRAMe<n>:CRC?....1649
BUS<m>:HBTO:RESult:FRAMe<n>:DATA?....1648
BUS<m>:HBTO:RESult:FRAMe<n>:DESTaddress?....1648
BUS<m>:HBTO:RESult:FRAMe<n>:DSYMbol?....1649
BUS<m>:HBTO:RESult:FRAMe<n>:FTYPe?....1646
BUS<m>:HBTO:RESult:FRAMe<n>:NUMWords?....1649
BUS<m>:HBTO:RESult:FRAMe<n>:SRCaddress?....1648
BUS<m>:HBTO:RESult:FRAMe<n>:SSYMbol?....1649
BUS<m>:HBTO:RESult:FRAMe<n>:STARt?....1647
BUS<m>:HBTO:RESult:FRAMe<n>:STATe?....1646
BUS<m>:HBTO:RESult:FRAMe<n>:STOP?....1647
BUS<m>:HBTO:RESult:FRAMe<n>:TYPE?....1648
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:TYPE?....1650
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VALue?....1650
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VSTR?....1651
BUS<m>:HBTO:RTYP....1636
BUS<m>:HBTO:SYMRate....1637
BUS<m>:HBTO:THReshold....1637
BUS<m>:I2C:FCOunt?....1397
BUS<m>:I2C:FRAMe<n>:AACCess?....1397
BUS<m>:I2C:FRAMe<n>:ACCess?....1398
BUS<m>:I2C:FRAMe<n>:ACOMplete?....1398
BUS<m>:I2C:FRAMe<n>:ADBStart?....1398
BUS<m>:I2C:FRAMe<n>:ADDRess?....1399
BUS<m>:I2C:FRAMe<n>:ADEVice?....1399
BUS<m>:I2C:FRAMe<n>:AMODe?....1399
BUS<m>:I2C:FRAMe<n>:ASTart?....1400
BUS<m>:I2C:FRAMe<n>:BCOunt?....1402
BUS<m>:I2C:FRAMe<n>:BITRate?....1400
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACCess?....1403
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACKStart?....1403
BUS<m>:I2C:FRAMe<n>:BYTE<o>:COMPlete?....1403
BUS<m>:I2C:FRAMe<n>:BYTE<o>:STARt?....1404
BUS<m>:I2C:FRAMe<n>:BYTE<o>:VALue?....1404
BUS<m>:I2C:FRAMe<n>:DATA?....1397
BUS<m>:I2C:FRAMe<n>:RWBStart?....1400

BUS<m>:I2C:FRAMe<n>:STARt?....1401
BUS<m>:I2C:FRAMe<n>:STATus?....1401
BUS<m>:I2C:FRAMe<n>:STOP?....1402
BUS<m>:I2C:FRAMe<n>:SYMBol?....1402
BUS<m>:I2C:RWBit....1390
BUS<m>:I2C:SCL:SOURce....1388
BUS<m>:I2C:SCL:THReshold....1389
BUS<m>:I2C:SDA:SOURce....1388
BUS<m>:I2C:SDA:THReshold....1389
BUS<m>:I2C:TECHnology....1389
BUS<m>:I2S:AVARiant....1554
BUS<m>:I2S:BORDer....1558
BUS<m>:I2S:CHANnel:LENGth....1559
BUS<m>:I2S:CHANnel:OFFSet....1558
BUS<m>:I2S:CHANnel:ORDer....1557
BUS<m>:I2S:CHANnel:TDMCount....1559
BUS<m>:I2S:CLOCk:POLarity....1555
BUS<m>:I2S:CLOCk:SOURce....1554
BUS<m>:I2S:CLOCk:THReshold....1557
BUS<m>:I2S:DATA:POLarity....1556
BUS<m>:I2S:DATA:SOURce....1556
BUS<m>:I2S:DATA:THReshold....1557
BUS<m>:I2S:FCOunt?....1564
BUS<m>:I2S:FOFFset....1559
BUS<m>:I2S:FRAMe<n>:LEFT:STATe?....1565
BUS<m>:I2S:FRAMe<n>:LEFT:VALue?....1565
BUS<m>:I2S:FRAMe<n>:RIGHt:STATe?....1565
BUS<m>:I2S:FRAMe<n>:RIGHt:VALue?....1565
BUS<m>:I2S:FRAMe<n>:STARt?....1564
BUS<m>:I2S:FRAMe<n>:STATe?....1564
BUS<m>:I2S:FRAMe<n>:STOP?....1564
BUS<m>:I2S:FRAMe<n>:TDM<o>:STATe?....1565
BUS<m>:I2S:FRAMe<n>:TDM<o>:VALue?....1566
BUS<m>:I2S:TCOupling....1556
BUS<m>:I2S:TRACk:LEFT....1567
BUS<m>:I2S:TRACk:RIGHt....1567
BUS<m>:I2S:TRACk:TD1Ch....1567
BUS<m>:I2S:TRACk:TD2Ch....1567
BUS<m>:I2S:TRACk:TD3Ch....1567
BUS<m>:I2S:TRACk:TD4Ch....1567
BUS<m>:I2S:TRACk:TD5Ch....1567
BUS<m>:I2S:TRACk:TD6Ch....1567
BUS<m>:I2S:TRACk:TD7Ch....1567
BUS<m>:I2S:TRACk:TD8Ch....1567
BUS<m>:I2S:WLENgth....1558
BUS<m>:I2S:WSELect:POLarity....1555
BUS<m>:I2S:WSELect:SOURce....1555
BUS<m>:I2S:WSELect:THReshold....1557
BUS<m>:LABel....1383
BUS<m>:LIN:BITRate....1493

BUS<m>:LIN:DATA:SOURce 1491
BUS<m>:LIN:DATA:THReshold 1492
BUS<m>:LIN:FCOunt? 1500
BUS<m>:LIN:FRAMe<n>:BITRate? 1505
BUS<m>:LIN:FRAMe<n>:BYTE<o>:STATe? 1506
BUS<m>:LIN:FRAMe<n>:BYTE<o>:VALue? 1506
BUS<m>:LIN:FRAMe<n>:CSSTate? 1505
BUS<m>:LIN:FRAMe<n>:CSValue? 1505
BUS<m>:LIN:FRAMe<n>:DATA? 1503
BUS<m>:LIN:FRAMe<n>:IDPValue? 1504
BUS<m>:LIN:FRAMe<n>:IDSTate? 1503
BUS<m>:LIN:FRAMe<n>:IDValue? 1503
BUS<m>:LIN:FRAMe<n>:SDATa? 1501
BUS<m>:LIN:FRAMe<n>:SDEXport? 1502
BUS<m>:LIN:FRAMe<n>:STARt? 1501
BUS<m>:LIN:FRAMe<n>:STATus? 1500
BUS<m>:LIN:FRAMe<n>:STOP? 1501
BUS<m>:LIN:FRAMe<n>:SYMBol? 1502
BUS<m>:LIN:FRAMe<n>:SYSTate? 1504
BUS<m>:LIN:FRAMe<n>:VERSion? 1502
BUS<m>:LIN:POLarity 1493
BUS<m>:LIN:STANdard 1493
BUS<m>:LIN:TECHnology 1492
BUS<m>:MDIO:CLOCk:SOURce 1738
BUS<m>:MDIO:CLOCk:THReshold:HIGH 1739
BUS<m>:MDIO:CLOCk:THReshold:LOW 1739
BUS<m>:MDIO:COUPling 1740
BUS<m>:MDIO:DATA:SOURce 1738
BUS<m>:MDIO:DATA:THReshold:HIGH 1739
BUS<m>:MDIO:DATA:THReshold:LOW 1740
BUS<m>:MDIO:PRESet 1740
BUS<m>:MDIO:WCOunt? 1743
BUS<m>:MDIO:WORD<n>:DATA? 1743
BUS<m>:MDIO:WORD<n>:PHYS? 1743
BUS<m>:MDIO:WORD<n>:REGI? 1744
BUS<m>:MDIO:WORD<n>:ST? 1744
BUS<m>:MDIO:WORD<n>:STARt? 1744
BUS<m>:MDIO:WORD<n>:STATe? 1745
BUS<m>:MDIO:WORD<n>:STOP? 1745
BUS<m>:MDIO:WORD<n>:SYMBol? 1746
BUS<m>:MDIO:WORD<n>:TYPE? 1746
BUS<m>:MILStd:MAXResponse:BITS 1569
BUS<m>:MILStd:MAXResponse:SELect 1569
BUS<m>:MILStd:MINGap:BITS 1570
BUS<m>:MILStd:MINGap:SELect 1569
BUS<m>:MILStd:POLarity 1570
BUS<m>:MILStd:PRESet 1570
BUS<m>:MILStd:SOURce 1569
BUS<m>:MILStd:THReshold:HIGH 1570
BUS<m>:MILStd:THReshold:LOW 1571

BUS<m>:MILStd:WCOunt?....1581
BUS<m>:MILStd:WORD<n>:DATA?....1582
BUS<m>:MILStd:WORD<n>:INFO?....1582
BUS<m>:MILStd:WORD<n>:RTADdress?....1582
BUS<m>:MILStd:WORD<n>:STARt?....1583
BUS<m>:MILStd:WORD<n>:STATus?....1583
BUS<m>:MILStd:WORD<n>:STOP?....1584
BUS<m>:MILStd:WORD<n>:SYMBol?....1584
BUS<m>:MILStd:WORD<n>:TYPE?....1584
BUS<m>:NEWList....1384
BUS<m>:PARallel:BIT<n>:DESKew....1869
BUS<m>:PARallel:BIT<n>:LABel....1870
BUS<m>:PARallel:BIT<n>[:STATe]....1867
BUS<m>:PARallel:CLEar....1872
BUS<m>:PARallel:CLOCk....1872
BUS<m>:PARallel:CLON....1872
BUS<m>:PARallel:CLSLope....1872
BUS<m>:PARallel:DATA:FORMat....1870
BUS<m>:PARallel:DATA:FORMat....1885
BUS<m>:PARallel:DATA:HEADer?....1885
BUS<m>:PARallel:DATA[:VALues]?....1885
BUS<m>:PARallel:DECTable<n>:COUNt?....1873
BUS<m>:PARallel:DECTable<n>:DATA?....1873
BUS<m>:PARallel:DECTable<n>:SHOW....1873
BUS<m>:PARallel:DESoffset....1870
BUS<m>:PARallel:DISPlay:BTYP....1871
BUS<m>:PARallel:DISPlay:SHBU....1871
BUS<m>:PARallel:DISPlay:SHDI....1871
BUS<m>:PARallel:HYSTeresis<n>....1869
BUS<m>:PARallel:STATe....1867
BUS<m>:PARallel:TECHnology....1868
BUS<m>:PARallel:THCoupling....1868
BUS<m>:PARallel:THReshold<n>....1867
BUS<m>:RESDetail....1384
BUS<m>:RESult....1383
BUS<m>:SENT:CLKPeriod....1665
BUS<m>:SENT:CLKTolerance....1666
BUS<m>:SENT:CRCMethod....1667
BUS<m>:SENT:CRCVersion....1666
BUS<m>:SENT:DATA:SOURce....1664
BUS<m>:SENT:DATA:THReshold....1665
BUS<m>:SENT:DNIBbles....1666
BUS<m>:SENT:FCOunt?....1675
BUS<m>:SENT:FRAMe<n>:CSValue?....1676
BUS<m>:SENT:FRAMe<n>:DATA?....1676
BUS<m>:SENT:FRAMe<n>:IDTYpe?....1676
BUS<m>:SENT:FRAMe<n>:IDValue?....1677
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:STATe?....1677
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:VALue?....1678
BUS<m>:SENT:FRAMe<n>:PAPTicks?....1678

BUS<m>:SENT:FRAMe<n>:SCOM? 1678
BUS<m>:SENT:FRAMe<n>:SDATa? 1679
BUS<m>:SENT:FRAMe<n>:SDEXport? 1679
BUS<m>:SENT:FRAMe<n>:STARt? 1675
BUS<m>:SENT:FRAMe<n>:STATus? 1675
BUS<m>:SENT:FRAMe<n>:STOP? 1675
BUS<m>:SENT:FRAMe<n>:SYMBol? 1679
BUS<m>:SENT:FRAMe<n>:SYNCduration? 1680
BUS<m>:SENT:FRAMe<n>:TYPE? 1680
BUS<m>:SENT:PPFLength 1667
BUS<m>:SENT:PPULse 1667
BUS<m>:SENT:RDSL 1681
BUS<m>:SENT:SFORmat 1666
BUS<m>:SENT:TECHnology 1665
BUS<m>:SETReflevels 1383
BUS<m>:SPI:BORDer 1420
BUS<m>:SPI:FCOunt? 1428
BUS<m>:SPI:FRAMe<n>:BITRate? 1429
BUS<m>:SPI:FRAMe<n>:COUNt? 1428
BUS<m>:SPI:FRAMe<n>:DATA? 1428
BUS<m>:SPI:FRAMe<n>:STARt? 1430
BUS<m>:SPI:FRAMe<n>:STATus? 1429
BUS<m>:SPI:FRAMe<n>:STOP? 1430
BUS<m>:SPI:FRAMe<n>:WCOunt? 1430
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMISo? 1432
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMOSi? 1433
BUS<m>:SPI:FRAMe<n>:WORD<o>:MISO? 1431
BUS<m>:SPI:FRAMe<n>:WORD<o>:MOSI? 1432
BUS<m>:SPI:FRAMe<n>:WORD<o>:STARt? 1431
BUS<m>:SPI:FRAMe<n>:WORD<o>:STOP? 1431
BUS<m>:SPI:FRCondition 1424
BUS<m>:SPI:MISO:POLarity 1422
BUS<m>:SPI:MISO:SOURce 1422
BUS<m>:SPI:MISO:THReshold 1424
BUS<m>:SPI:MOSI:POLarity 1423
BUS<m>:SPI:MOSI:SOURce 1423
BUS<m>:SPI:MOSI:THReshold 1424
BUS<m>:SPI:SCLK:SOURce 1421
BUS<m>:SPI:SCLK:THReshold 1424
BUS<m>:SPI:SSELect:POLarity 1422
BUS<m>:SPI:SSELect:SOURce 1421
BUS<m>:SPI:SSELect:THReshold 1424
BUS<m>:SPI:TECHnology 1423
BUS<m>:SPI:TIMeout 1425
BUS<m>:SPI:WSIZe 1421
BUS<m>:SWIRe:BPOSition 1818
BUS<m>:SWIRe:COUPling 1821
BUS<m>:SWIRe:DATA:HYSTeresis 1819
BUS<m>:SWIRe:DATA:SOURce 1819
BUS<m>:SWIRe:DATA:THReshold 1819

BUS<m>:SWIRe:MGAP....1819
BUS<m>:SWIRe:PRESet....1821
BUS<m>:SWIRe:RESults:FCOunt....1825
BUS<m>:SWIRe:RESults:FRAMe<n>:DATA?....1825
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:FVAL?....1827
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STARt?....1827
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STATus?....1828
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STOP?....1828
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:TYPE?....1829
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:VAL?....1829
BUS<m>:SWIRe:RESults:FRAMe<n>:STARt?....1825
BUS<m>:SWIRe:RESults:FRAMe<n>:STATe?....1826
BUS<m>:SWIRe:RESults:FRAMe<n>:STOP?....1826
BUS<m>:SWIRe:RESults:FRAMe<n>:TYPE?....1826
BUS<m>:SWIRe:STRBe:HYSTeresis....1820
BUS<m>:SWIRe:STRBe:SOURce....1820
BUS<m>:SWIRe:STRBe:THReshold....1820
BUS<m>:SWIRe:SYSLect....1820
BUS<m>:SYMBols....1385
BUS<m>:THReshold....1383
BUS<m>:TYPE....1382
BUS<m>:TYPE....1387
BUS<m>:UART:BAUDrate....1442
BUS<m>:UART:BITRate....1442
BUS<m>:UART:EWORd....1445
BUS<m>:UART:PACKets....1444
BUS<m>:UART:PARity....1442
BUS<m>:UART:POLarity....1443
BUS<m>:UART:RX:SOURce....1440
BUS<m>:UART:RX:THReshold....1441
BUS<m>:UART:SBIT....1443
BUS<m>:UART:SSIZe....1443
BUS<m>:UART:TECHnology....1441
BUS<m>:UART:TOUT....1444
BUS<m>:UART:TX:SOURce....1440
BUS<m>:UART:TX:THReshold....1441
BUS<m>:UART:WORD<n>:BITRate?....1448
BUS<m>:UART:WORD<n>:COUNt?....1448
BUS<m>:UART:WORD<n>:RXValue?....1447
BUS<m>:UART:WORD<n>:SOURce?....1448
BUS<m>:UART:WORD<n>:STARt?....1449
BUS<m>:UART:WORD<n>:STATe?....1449
BUS<m>:UART:WORD<n>:TXValue?....1447
BUS<m>:USB:DATA:SOURce....1754
BUS<m>:USB:DATA:THReshold....1756
BUS<m>:USB:DIFFerential:SOURce....1754
BUS<m>:USB:DIFFerential:THReshold....1756
BUS<m>:USB:DMINus:SOURce....1754
BUS<m>:USB:DMINus:THReshold....1755
BUS<m>:USB:DPLus:SOURce....1754

BUS<m>:USB:DPLus:THReshold ... 1755
BUS<m>:USB:PACKet<n>:ADDRess? ... 1769
BUS<m>:USB:PACKet<n>:CRC? ... 1770
BUS<m>:USB:PACKet<n>:DATA? ... 1770
BUS<m>:USB:PACKet<n>:ENDPoint? ... 1770
BUS<m>:USB:PACKet<n>:ET? ... 1771
BUS<m>:USB:PACKet<n>:FRAMe? ... 1771
BUS<m>:USB:PACKet<n>:PID? ... 1768
BUS<m>:USB:PACKet<n>:PORT? ... 1771
BUS<m>:USB:PACKet<n>:SC? ... 1772
BUS<m>:USB:PACKet<n>:SEU? ... 1772
BUS<m>:USB:PACKet<n>:STARt? ... 1772
BUS<m>:USB:PACKet<n>:STATus? ... 1773
BUS<m>:USB:PACKet<n>:STOP? ... 1773
BUS<m>:USB:PCOunt? ... 1773
BUS<m>:USB:STRobe:SOURce ... 1755
BUS<m>:USB:STRobe:THReshold ... 1756
BUS<m>:USB:TECHnology ... 1753
BUS<m>:USBPd:DETail ... 1794
BUS<m>:USBPd:HYSTeresis ... 1794
BUS<m>:USBPd:RESult:FCOunt? ... 1802
BUS<m>:USBPd:RESult:FRAMe<n>:DATA? ... 1803
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL? ... 1803
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt? ... 1804
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus? ... 1803
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP? ... 1804
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE? ... 1805
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL? ... 1805
BUS<m>:USBPd:RESult:FRAMe<n>:INFO? ... 1805
BUS<m>:USBPd:RESult:FRAMe<n>:STARt? ... 1806
BUS<m>:USBPd:RESult:FRAMe<n>:STATe? ... 1806
BUS<m>:USBPd:RESult:FRAMe<n>:STOP? ... 1806
BUS<m>:USBPd:RESult:FRAMe<n>:TYPE? ... 1807
BUS<m>:USBPd:SOURce ... 1794
BUS<m>:USBPd:THRBottom ... 1795
BUS<m>:USBPd:THReshold ... 1795
BUS<m>:USBPd:THRMid ... 1795
BUS<m>:USBPd:THRTop ... 1796
BUS<m>:ZCOupling ... 1385
BUS<m>[:STATe] ... 1382
CALCulate:MATH<m>:ARIThmetics ... 1190
CALCulate:MATH<m>:DATA:HEADer? ... 1193
CALCulate:MATH<m>:DATA:HEADer? ... 1291
CALCulate:MATH<m>:DATA:STYPe? ... 1192
CALCulate:MATH<m>:DATA:STYPe? ... 1291
CALCulate:MATH<m>:DATA[:VALues]? ... 1193
CALCulate:MATH<m>:DATA[:VALues]? ... 1292
CALCulate:MATH<m>:ENVSelection ... 1190
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted? ... 1281
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO ... 1281

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio........1281
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue]........1282
CALCulate:MATH<m>:FFT:CFRequency........1280
CALCulate:MATH<m>:FFT:COUPled:WITH<1..8>........1290
CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics........1283
CALCulate:MATH<m>:FFT:FRAMe:COVerage?........1284
CALCulate:MATH<m>:FFT:FRAMe:MAXCount........1284
CALCulate:MATH<m>:FFT:FRAMe:OFACtor........1284
CALCulate:MATH<m>:FFT:FULLspan........1280
CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt........1286
CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP........1286
CALCulate:MATH<m>:FFT:GATE:COUPling........1285
CALCulate:MATH<m>:FFT:GATE:MODE........1286
CALCulate:MATH<m>:FFT:GATE:RELative:STARt........1286
CALCulate:MATH<m>:FFT:GATE:RELative:STOP........1287
CALCulate:MATH<m>:FFT:GATE:ZCOupling........1287
CALCulate:MATH<m>:FFT:GATE[:STATe]........1287
CALCulate:MATH<m>:FFT:LOGScale........1279
CALCulate:MATH<m>:FFT:MAGNitude:LEVel........1288
CALCulate:MATH<m>:FFT:MAGNitude:RANGe........1288
CALCulate:MATH<m>:FFT:MAGNitude:SCALe........1288
CALCulate:MATH<m>:FFT:PHASe:SCALe........1289
CALCulate:MATH<m>:FFT:PHASe:SUPPression........1289
CALCulate:MATH<m>:FFT:PHASe:THReshold........1290
CALCulate:MATH<m>:FFT:PHASe:UNWRap........1290
CALCulate:MATH<m>:FFT:SPAN........1280
CALCulate:MATH<m>:FFT:SPECtrogram:CMODe........1292
CALCulate:MATH<m>:FFT:SPECtrogram:STATe........1293
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition........1293
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe........1294
CALCulate:MATH<m>:FFT:STARt........1279
CALCulate:MATH<m>:FFT:STOP........1280
CALCulate:MATH<m>:FFT:USEColtab........1293
CALCulate:MATH<m>:FFT:WINDow:TYPE........1282
CALCulate:MATH<m>:STATe........1190
CALCulate:MATH<m>:VERTical:OFFSet........1191
CALCulate:MATH<m>:VERTical:RANGe........1192
CALCulate:MATH<m>:VERTical:SCALe........1192
CALCulate:MATH<m>[:EXPRession][:DEFine]........1189
CALibration:DATE?........1946
CALibration:RESult?........1947
CALibration:TIME?........1947
CHANnel<m>:BANDwidth........1083
CHANnel<m>:COUPling........1080
CHANnel<m>:CPLing........1083
CHANnel<m>:DIGFilter:STATe........1123
CHANnel<m>:EATScale........1089
CHANnel<m>:EATTenuation........1089
CHANnel<m>:EXPortstate........1364
CHANnel<m>:GND........1080

CHANnel<m>:IMPedance....1083
CHANnel<m>:INVert....1082
CHANnel<m>:OFFSet....1082
CHANnel<m>:OVERload....1084
CHANnel<m>:POSition....1081
CHANnel<m>:RANGe....1081
CHANnel<m>:SCALe....1080
CHANnel<m>:SKEW:MANual....1124
CHANnel<m>:SKEW:TIME....1124
CHANnel<m>:STATe....1079
CHANnel<m>[:WAVeform<n>]:ARIThmetics....1077
CHANnel<m>[:WAVeform<n>]:DATA:HEADer?....1084
CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?....1085
CHANnel<m>[:WAVeform<n>]:HISTory:CURRent....1195
CHANnel<m>[:WAVeform<n>]:HISTory:PLAY....1196
CHANnel<m>[:WAVeform<n>]:HISTory:REPLay....1196
CHANnel<m>[:WAVeform<n>]:HISTory:STARt....1195
CHANnel<m>[:WAVeform<n>]:HISTory:STOP....1195
CHANnel<m>[:WAVeform<n>]:HISTory:TPACq....1196
CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?....1197
CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?....1197
CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?....1197
CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?....1198
CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]....1194
CHANnel<m>[:WAVeform<n>]:TYPE....1077
CHANnel<m>[:WAVeform<n>][:STATe]....1076
CURSor<m>:AOFF....1202
CURSor<m>:FFT:SETCenter....1210
CURSor<m>:FFT:TOCenter....1210
CURSor<m>:FUNCtion....1202
CURSor<m>:LABel....1208
CURSor<m>:MAXimum:LEFT....1211
CURSor<m>:MAXimum:NEXT....1211
CURSor<m>:MAXimum:RIGHt....1211
CURSor<m>:MAXimum[:PEAK]....1210
CURSor<m>:PEXCursion....1212
CURSor<m>:PEXCursion....1341
CURSor<m>:SIAD....1208
CURSor<m>:SOURce....1203
CURSor<m>:SSOurce....1203
CURSor<m>:STATe....1201
CURSor<m>:STYLe....1207
CURSor<m>:THReshold....1211
CURSor<m>:THReshold....1342
CURSor<m>:TRACking[:STATe]....1202
CURSor<m>:USSource....1204
CURSor<m>:X1ENvelope....1206
CURSor<m>:X1Position....1204
CURSor<m>:X2ENvelope....1207
CURSor<m>:X2Position....1204

CURSor<m>:XCOupling....1205
CURSor<m>:XDELta:INVerse?....1209
CURSor<m>:XDELta[:VALue]?....1209
CURSor<m>:Y1Position....1205
CURSor<m>:Y2Position....1205
CURSor<m>:YCOupling....1206
CURSor<m>:YDELta:SLOPe....1210
CURSor<m>:YDELta[:VALue]?....1209
DIAGnostic:SERVice:CHANnelcount?....1050
DIAGnostic:SERVice:COMPutername....1050
DIAGnostic:SERVice:FWVersion?....1050
DIAGnostic:SERVice:PARTnumber....1050
DIAGnostic:SERVice:PWD....1947
DIAGnostic:SERVice:SERialnumber?....1050
DIAGnostic:SERVice:STST:EXECute....1947
DIAGnostic:SERVice:STST:STATe?....1947
DIAGnostic:SERVice:WFAModel?....1946
DIAGnostic:SERVice:WFASeries?....1946
DIAGnostic:SERVice:WFAType?....1946
DIFFerential<m>:AOUTput....1121
DIFFerential<m>:BOUTput....1121
DIFFerential<m>:COMMon:OFFSet....1122
DIFFerential<m>:COMMon:POSition....1122
DIFFerential<m>:COMMon:SCALe....1121
DIFFerential<m>:COUPling....1122
DIFFerential<m>:DIFFerential:OFFSet....1122
DIFFerential<m>:DIFFerential:POSition....1122
DIFFerential<m>:DIFFerential:SCALe....1121
DIFFerential<m>:NSIGnal[:SELect]....1121
DIFFerential<m>:PSIGnal[:SELect]....1121
DIFFerential<m>:STATe....1120
DIGital<m>:DATA:HEADer?....1884
DIGital<m>:DATA[:VALues]?....1884
DIGital<m>:DESKew....1866
DIGital<m>:DISPlay....1864
DIGital<m>:HYSTeresis....1865
DIGital<m>:LABel....1865
DIGital<m>:TECHnology....1864
DIGital<m>:THCoupling....1865
DIGital<m>:THReshold....1864
DISPlay:CLR....1068
DISPlay:COLor:PALette:ADD....1055
DISPlay:COLor:PALette:COUNt?....1056
DISPlay:COLor:PALette:POINt:ADD....1056
DISPlay:COLor:PALette:POINt:COUNt?....1057
DISPlay:COLor:PALette:POINt:INSert....1056
DISPlay:COLor:PALette:POINt:REMove....1056
DISPlay:COLor:PALette:POINt[:VALue]....1056
DISPlay:COLor:PALette:REMove....1056
DISPlay:COLor:SIGNal<m>:ASSign....1055

DISPlay:COLor:SIGNal<m>:COLor....1054
DISPlay:COLor:SIGNal<m>:USE....1055
DISPlay:DIAGram:CROSshair....1061
DISPlay:DIAGram:FINegrid....1061
DISPlay:DIAGram:GRID....1060
DISPlay:DIAGram:LABels....1061
DISPlay:DIAGram:REName....1060
DISPlay:DIAGram:STYLe....1054
DISPlay:DIAGram:TITLe....1061
DISPlay:DIAGram:XFIXed....1062
DISPlay:DIAGram:YFIXed....1061
DISPlay:GATE:TRANsparency....1062
DISPlay:INTensity....1054
DISPlay:PERSistence:INFinite....1053
DISPlay:PERSistence:RESet....1054
DISPlay:PERSistence:TIME....1053
DISPlay:PERSistence[:STATe]....1053
DISPlay:RESultboxes:CUPosition....1208
DISPlay:RESultboxes:DEFaultpos....1062
DISPlay:RESultboxes:DEPosition....1385
DISPlay:RESultboxes:MEPosition....1251
DISPlay:SIGBar:COLor:BORDer....1064
DISPlay:SIGBar:COLor:FILL....1064
DISPlay:SIGBar:HIDE:HEAD....1063
DISPlay:SIGBar:HIDE:TIME....1063
DISPlay:SIGBar:HIDE:TRANsparency....1063
DISPlay:SIGBar:HIDE[:AUTO]....1063
DISPlay:SIGBar:POSition....1063
DISPlay:SIGBar[:STATe]....1062
DISPlay:SIGNal:LABel:ADD....1065
DISPlay:SIGNal:LABel:FONTsize....1068
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition....1067
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition....1068
DISPlay:SIGNal:LABel:POSMode....1066
DISPlay:SIGNal:LABel:REMove....1066
DISPlay:SIGNal:LABel:TEXT....1066
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition....1067
DISPlay:SIGNal:LABel:VERTical:RELative:POSition....1068
DISPlay:TRIGger:LINes....1128
EXECutable:NAME....1069
EXECutable:PARameter....1069
EXECutable:WDIRectory....1069
EXPort:HISTogram:DATA?....1370
EXPort:HISTogram:INCidence....1370
EXPort:HISTogram:NAME....1370
EXPort:HISTogram:SAVE....1370
EXPort:HISTogram:SELect....1369
EXPort:MEASurement:DATA?....1373
EXPort:MEASurement:NAME....1373
EXPort:MEASurement:SAVE....1373

EXPort:MEASurement:SELect....1372
EXPort:MEASurement:TYPE....1372
EXPort:RESult:DECSymbol....1372
EXPort:RESult:DELimiter....1372
EXPort:RESult:NAME....1371
EXPort:RESult:NUMeric....1371
EXPort:RESult:SAVE....1371
EXPort:RESult:SELect....1371
EXPort:WAVeform:CURSorset....1366
EXPort:WAVeform:DISPlayoff....1369
EXPort:WAVeform:DLOGging....1367
EXPort:WAVeform:FASTexport....1369
EXPort:WAVeform:INCXvalues....1368
EXPort:WAVeform:MEAS....1366
EXPort:WAVeform:MULTichannel....1364
EXPort:WAVeform:NAME....1364
EXPort:WAVeform:RAW....1368
EXPort:WAVeform:SAVE....1365
EXPort:WAVeform:SCOPe....1365
EXPort:WAVeform:SOURce....1363
EXPort:WAVeform:STARt....1365
EXPort:WAVeform:STOP....1366
EXPort:WAVeform:TIMestamps....1368
EXPort:WAVeform:ZOOM....1366
FORMat:BORDer....1046
FORMat:BORDer....1126
FORMat:BPATtern....1046
FORMat[:DATA]....1045
GENerator:ALIGnment:DC:ABORt....1905
GENerator:ALIGnment:DC:RESult:DATE?....1905
GENerator:ALIGnment:DC:RESult:TIME?....1905
GENerator:ALIGnment:DC:RESult[:STATe]?....1906
GENerator:ALIGnment:DC[:STARt]....1906
GENerator:SYNC[:COMBination]....1905
GPIB:ADDRess....1047
GPIB:TERMinator....1047
HCOPy:CMAP<m>:DEFault....1377
HCOPy:DESTination<1..2>....1374
HCOPy:DEVice<m>:COLor....1376
HCOPy:DEVice<m>:INVerse....1376
HCOPy:DEVice<m>:LANGuage....1375
HCOPy:IMMediate<m>:NEXT....1378
HCOPy:IMMediate<m>[:DUM]....1378
HCOPy:ISBA....1377
HCOPy:PAGE:ORIentation<1..2>....1376
HCOPy:SSD....1377
HCOPy:WBKG....1376
HDEFinition:BWIDth....1125
HDEFinition:RESolution?....1126
HDEFinition:STATe....1125

LAYout:ADD....1057
LAYout:HISTogram:ADD....1241
LAYout:HISTogram:HORZ:ABSolute:STARt....1243
LAYout:HISTogram:HORZ:ABSolute:STOP....1243
LAYout:HISTogram:HORZ:MODE....1242
LAYout:HISTogram:HORZ:RELative:STARt....1243
LAYout:HISTogram:HORZ:RELative:STOP....1243
LAYout:HISTogram:MODE....1242
LAYout:HISTogram:REMove....1245
LAYout:HISTogram:RESet....1245
LAYout:HISTogram:SOURce....1241
LAYout:HISTogram:VERTical:ABSolute:STARt....1244
LAYout:HISTogram:VERTical:ABSolute:STOP....1244
LAYout:HISTogram:VERTical:MODE....1244
LAYout:HISTogram:VERTical:RELative:STARt....1245
LAYout:HISTogram:VERTical:RELative:STOP....1245
LAYout:REMove....1058
LAYout:SHOW....1059
LAYout:SIGNal:ASSign....1059
LAYout:SIGNal:AXIS....1199
LAYout:SIGNal:UNASsign....1059
LAYout:ZOOM:ADD....1174
LAYout:ZOOM:ADDCoupled....1175
LAYout:ZOOM:HORZ:ABSolute:POSition....1176
LAYout:ZOOM:HORZ:ABSolute:SPAN....1176
LAYout:ZOOM:HORZ:ABSolute:STARt....1177
LAYout:ZOOM:HORZ:ABSolute:STOP....1177
LAYout:ZOOM:HORZ:MODE....1176
LAYout:ZOOM:HORZ:RELative:POSition....1178
LAYout:ZOOM:HORZ:RELative:SPAN....1178
LAYout:ZOOM:HORZ:RELative:STARt....1178
LAYout:ZOOM:HORZ:RELative:STOP....1179
LAYout:ZOOM:ONEDiagram....1175
LAYout:ZOOM:POSCoupling....1175
LAYout:ZOOM:REMove....1182
LAYout:ZOOM:VERTical:ABSolute:POSition....1179
LAYout:ZOOM:VERTical:ABSolute:SPAN....1180
LAYout:ZOOM:VERTical:ABSolute:STARt....1180
LAYout:ZOOM:VERTical:ABSolute:STOP....1181
LAYout:ZOOM:VERTical:MODE....1179
LAYout:ZOOM:VERTical:RELative:POSition....1181
LAYout:ZOOM:VERTical:RELative:SPAN....1181
LAYout:ZOOM:VERTical:RELative:STARt....1182
LAYout:ZOOM:VERTical:RELative:STOP....1182
MEASurement<m>:AMPTime:ALEVel....1223
MEASurement<m>:AMPTime:CLCK<n>:LSELect....1227
MEASurement<m>:AMPTime:CSLope....1226
MEASurement<m>:AMPTime:DATA<n>:LSELect....1227
MEASurement<m>:AMPTime:DELay<n>:DIRection....1224
MEASurement<m>:AMPTime:DELay<n>:ECOunt....1225

MEASurement<m>:AMPTime:DELay<n>:SLOPe....1225
MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect....1228
MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe....1227
MEASurement<m>:AMPTime:ESLope....1226
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit....1228
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin....1229
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit....1228
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin....1229
MEASurement<m>:AMPTime:LCHeck<n>:VALid....1228
MEASurement<m>:AMPTime:PFSLope....1223
MEASurement<m>:AMPTime:PSLope....1224
MEASurement<m>:AMPTime:PTCount....1226
MEASurement<m>:ARES?....1217
MEASurement<m>:ARNames....1218
MEASurement<m>:CATegory....1215
MEASurement<m>:CLEar....1252
MEASurement<m>:DETThreshold....1223
MEASurement<m>:DISPlay:HISTogram....1251
MEASurement<m>:DISPlay:LEVels....1250
MEASurement<m>:DISPlay:RESults....1250
MEASurement<m>:DISPlay:STYLe....1251
MEASurement<m>:ENVSelect....1222
MEASurement<m>:EYEJitter:AUToscale....1231
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit....1231
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin....1232
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit....1231
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin....1232
MEASurement<m>:EYEJitter:LCHeck<n>:VALid....1231
MEASurement<m>:FSRC....1214
MEASurement<m>:GATE:ABSolute:STARt....1266
MEASurement<m>:GATE:ABSolute:STOP....1266
MEASurement<m>:GATE:CCOupling....1267
MEASurement<m>:GATE:CURSor....1267
MEASurement<m>:GATE:GCOupling....1268
MEASurement<m>:GATE:MODE....1266
MEASurement<m>:GATE:RELative:STARt....1266
MEASurement<m>:GATE:RELative:STOP....1266
MEASurement<m>:GATE:ZCOupling....1267
MEASurement<m>:GATE:ZDIagram....1268
MEASurement<m>:GATE[:STATe]....1265
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit....1249
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin....1249
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit....1249
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin....1249
MEASurement<m>:HISTogram:LCHeck<n>:VALid....1248
MEASurement<m>:HISTogram:PROBability:LIMit....1248
MEASurement<m>:HISTogram:PROBability:TYPE....1247
MEASurement<m>:HISTogram:SELect....1247
MEASurement<m>:LCHeck....1268
MEASurement<m>:LTMeas:COUNt....1257

MEASurement<m>:LTMeas:TIME....1258
MEASurement<m>:LTMeas[:STATe]....1257
MEASurement<m>:MAIN....1215
MEASurement<m>:MNOMeas....1253
MEASurement<m>:MULTiple....1253
MEASurement<m>:ONViolation:ACQStop....1269
MEASurement<m>:ONViolation:BEEP....1269
MEASurement<m>:ONViolation:PRINt....1270
MEASurement<m>:ONViolation:REPort....1270
MEASurement<m>:ONViolation:RUNexec....1271
MEASurement<m>:ONViolation:TRIGgerout....1271
MEASurement<m>:ONViolation:WFMSave....1270
MEASurement<m>:PROTocol:F2FRame:FLDFrom....1259
MEASurement<m>:PROTocol:F2FRame:FLDTo....1259
MEASurement<m>:PROTocol:F2FRame:FRMFrom....1260
MEASurement<m>:PROTocol:F2FRame:FRMTo....1260
MEASurement<m>:PROTocol:F2FRame:VALFrom....1260
MEASurement<m>:PROTocol:F2FRame:VALTo....1260
MEASurement<m>:PROTocol:FLDValue:FLD....1261
MEASurement<m>:PROTocol:FLDValue:FRM....1261
MEASurement<m>:PROTocol:FLDValue:TRCK....1261
MEASurement<m>:PROTocol:FLDValue:VAL....1261
MEASurement<m>:PROTocol:MBITrate:FLD....1262
MEASurement<m>:PROTocol:MBITrate:FRM....1262
MEASurement<m>:PROTocol:MBITrate:VAL....1262
MEASurement<m>:PROTocol:SBITrate:FLD....1262
MEASurement<m>:PROTocol:SBITrate:FRM....1262
MEASurement<m>:PROTocol:SBITrate:VAL....1263
MEASurement<m>:PROTocol:T2FRame:DIRection....1263
MEASurement<m>:PROTocol:T2FRame:FLD....1263
MEASurement<m>:PROTocol:T2FRame:FRM....1263
MEASurement<m>:PROTocol:T2FRame:VALue....1264
MEASurement<m>:REFLevel:RESult:LOWer?....1277
MEASurement<m>:REFLevel:RESult:MIDDle?....1277
MEASurement<m>:REFLevel:RESult:SIGHigh?....1277
MEASurement<m>:REFLevel:RESult:SIGLow?....1277
MEASurement<m>:REFLevel:RESult:UPPer?....1277
MEASurement<m>:RESult:AVG?....1218
MEASurement<m>:RESult:COUNt?....1219
MEASurement<m>:RESult:EVTCount?....1218
MEASurement<m>:RESult:INVerse....1237
MEASurement<m>:RESult:LABorder....1237
MEASurement<m>:RESult:MAXCount....1236
MEASurement<m>:RESult:NPEak?....1218
MEASurement<m>:RESult:PPEak?....1218
MEASurement<m>:RESult:RELiability?....1218
MEASurement<m>:RESult:RMS?....1218
MEASurement<m>:RESult:SHFRequency....1238
MEASurement<m>:RESult:SHLabels....1238
MEASurement<m>:RESult:STARt?....1219

MEASurement<m>:RESult:STDDev?........1218
MEASurement<m>:RESult:STOP?........1219
MEASurement<m>:RESult:WFMCount?........1218
MEASurement<m>:RESult[:ACTual]?........1218
MEASurement<m>:SOURce........1213
MEASurement<m>:SPECtrum:ATHReshold........1235
MEASurement<m>:SPECtrum:CPOWer:BANDwidth........1233
MEASurement<m>:SPECtrum:CPOWer:CFRequency........1234
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit........1239
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin........1240
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit........1239
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin........1240
MEASurement<m>:SPECtrum:LCHeck<n>:VALid........1239
MEASurement<m>:SPECtrum:NDBDown........1234
MEASurement<m>:SPECtrum:OBANdwidth........1234
MEASurement<m>:SPECtrum:PEXCursion........1235
MEASurement<m>:SPECtrum:RESult<n>:COUNt........1235
MEASurement<m>:SPECtrum:RESult<n>:MODE........1236
MEASurement<m>:SSRC........1215
MEASurement<m>:STATistics:HBINs........1254
MEASurement<m>:STATistics:HISTogram........1253
MEASurement<m>:STATistics:MODE........1254
MEASurement<m>:STATistics:RCOunt........1254
MEASurement<m>:STATistics:RESet........1255
MEASurement<m>:STATistics:RMEascount........1255
MEASurement<m>:STATistics:RTIMe........1255
MEASurement<m>:STATistics[:ENABle]........1252
MEASurement<m>:TRACk:DATA:HEADer?........1264
MEASurement<m>:TRACk:DATA:HEADer?........1568
MEASurement<m>:TRACk:DATA:STYPe?........1265
MEASurement<m>:TRACk:DATA:STYPe?........1568
MEASurement<m>:TRACk:DATA[:VALues]?........1265
MEASurement<m>:TRACk:DATA[:VALues]?........1568
MEASurement<m>:TRACk[:STATe]........1264
MEASurement<m>:TRACk[:STATe]........1567
MEASurement<m>:VERTical:AUTO........1256
MEASurement<m>:VERTical:CONT........1256
MEASurement<m>:VERTical:OFFSet........1256
MEASurement<m>:VERTical:SCALe........1257
MEASurement<m>[:ENABle]........1213
MMEMory:ATTRibute........1358
MMEMory:AUTonaming:DATE........1361
MMEMory:AUTonaming:DEFaultpath........1361
MMEMory:AUTonaming:INDex........1361
MMEMory:AUTonaming:PREFix........1361
MMEMory:AUTonaming:RESall........1362
MMEMory:AUTonaming:RESPath........1362
MMEMory:AUTonaming:TEXT........1361
MMEMory:AUTonaming:TIME........1361
MMEMory:AUTonaming:USERtext........1361

MMEMory:CATalog:LENGth?.......1356
MMEMory:CATalog?.......1355
MMEMory:CDIRectory.......1355
MMEMory:COPY.......1356
MMEMory:DATA.......1357
MMEMory:DCATalog:LENGth?.......1354
MMEMory:DCATalog?.......1354
MMEMory:DELete.......1357
MMEMory:DRIVes?.......1353
MMEMory:LOAD:STATe.......1360
MMEMory:MDIRectory.......1355
MMEMory:MOVE.......1357
MMEMory:MSIS.......1354
MMEMory:NAME.......1375
MMEMory:RCL.......1359
MMEMory:RDIRectory.......1355
MMEMory:SAV.......1358
MMEMory:STORe:STATe.......1359
MTESt:ADD.......1295
MTESt:COLor:BORDer.......1311
MTESt:COLor:INTerior.......1311
MTESt:COLor:MATCh.......1310
MTESt:COLor:UNMatch.......1310
MTESt:CONDition.......1296
MTESt:CTYPe.......1297
MTESt:FILE:DELete.......1298
MTESt:FILE:NAME.......1298
MTESt:FILE:OPEN.......1298
MTESt:FILE:SAVE.......1298
MTESt:HIGHlight:INFinite.......1309
MTESt:HIGHlight:STATe.......1310
MTESt:HIGHlight:TIME.......1310
MTESt:LABel.......1309
MTESt:ONViolation:BEEP.......1307
MTESt:ONViolation:PRINt.......1307
MTESt:ONViolation:REPort.......1308
MTESt:ONViolation:RUNexec.......1308
MTESt:ONViolation:SAVewaveform.......1308
MTESt:ONViolation:STOP.......1307
MTESt:ONViolation:TRIGgerout.......1308
MTESt:REFWfm.......1304
MTESt:REMove.......1295
MTESt:REName.......1309
MTESt:RESult:COUNt:FAILures?.......1313
MTESt:RESult:COUNt:FWAVeforms?.......1313
MTESt:RESult:COUNt:REMaining?.......1313
MTESt:RESult:COUNt:WAVeforms?.......1312
MTESt:RESult:FRATe?.......1314
MTESt:RESult:STATe?.......1312
MTESt:RESult[:RESult]?.......1312

MTESt:RST....1296
MTESt:SBITnumber?....1952
MTESt:SEGMent:ADD....1299
MTESt:SEGMent:CLEar....1300
MTESt:SEGMent:COUNt?....1300
MTESt:SEGMent:INSert....1300
MTESt:SEGMent:POINt:ADD....1301
MTESt:SEGMent:POINt:COUNt?....1302
MTESt:SEGMent:POINt:INSert....1301
MTESt:SEGMent:POINt:REMove....1301
MTESt:SEGMent:POINt:X....1302
MTESt:SEGMent:POINt:Y....1302
MTESt:SEGMent:REGion....1300
MTESt:SEGMent:REMove....1300
MTESt:SEGMent:RESCale:RECalculate....1303
MTESt:SEGMent:RESCale:XFACtor....1303
MTESt:SEGMent:RESCale:XOFFset....1303
MTESt:SEGMent:RESCale:YFACtor....1303
MTESt:SEGMent:RESCale:YOFFset....1304
MTESt:SEGMent:STATe....1299
MTESt:SOURce....1296
MTESt:TOLerance....1297
MTESt:WFMLupdate....1305
MTESt:WFMRescale:XWIDth....1305
MTESt:WFMRescale:YPOSition....1306
MTESt:WFMRescale:YSTRetch....1306
MTESt:WFMRescale:YWIDth....1305
MTESt[:STATe]....1295
PGENerator:BITRate....1902
PGENerator:ENABle....1902
PGENerator:FILE:OPEN....1902
PGENerator:FILE[:NAME]....1903
PGENerator:HLEVel....1903
PGENerator:PRESet....1903
PGENerator:RUNMode....1903
POWer:DESKew:CURRent....1907
POWer:DESKew:EXECute....1908
POWer:DESKew:RESet....1908
POWer:DESKew:TIME?....1908
POWer:DESKew:UDPReset....1908
POWer:DONRes:AUTO....1922
POWer:DONRes:AVG....1922
POWer:DONRes:EXECute....1923
POWer:DONRes:GATE<m>:STARt....1923
POWer:DONRes:GATE<m>:STOP....1923
POWer:DONRes:REPort:ADD....1923
POWer:DONRes:RESult:RESistance?....1923
POWer:EFFiciency:AUTO....1933
POWer:EFFiciency:EXECute....1934
POWer:EFFiciency:REPort:ADD....1934

POWer:EFFiciency:RESult<m>:ACTual?....1934
POWer:EFFiciency:RESult<m>:AVG?....1934
POWer:EFFiciency:RESult<m>:EVTCount?....1934
POWer:EFFiciency:RESult<m>:NPEak?....1934
POWer:EFFiciency:RESult<m>:PPEak?....1934
POWer:EFFiciency:RESult<m>:RMS?....1934
POWer:EFFiciency:RESult<m>:STDDev?....1934
POWer:EFFiciency:RESult<m>:WFMCount?....1934
POWer:ENABle....1907
POWer:HARMonics:AUTO....1918
POWer:HARMonics:DOFR....1918
POWer:HARMonics:ENFR....1918
POWer:HARMonics:EVAL....1919
POWer:HARMonics:EXECute....1919
POWer:HARMonics:MIFR....1919
POWer:HARMonics:REPort:ADD....1919
POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?....1919
POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?....1919
POWer:HARMonics:RESult<m>:STDinuse?....1919
POWer:HARMonics:RESult<m>:STDValue<n>:VALue?....1919
POWer:HARMonics:RESult<m>:VALue<n>:VALue?....1919
POWer:HARMonics:STAN....1920
POWer:HARMonics:VOLT....1920
POWer:INRush:ADD....1916
POWer:INRush:COUNt?....1916
POWer:INRush:EXECute....1917
POWer:INRush:GATE<m>:STARt....1917
POWer:INRush:GATE<m>:STOP....1917
POWer:INRush:GATE<m>:VALue....1917
POWer:INRush:INSert....1916
POWer:INRush:MAXCurrent....1917
POWer:INRush:REMove....1916
POWer:INRush:REPort:ADD....1918
POWer:INRush:TRIGger....1917
POWer:MODulation:AUTO....1921
POWer:MODulation:DHIStogram....1921
POWer:MODulation:EXECute....1921
POWer:MODulation:REPort:ADD....1921
POWer:MODulation:RESult:ACTual?....1921
POWer:MODulation:RESult:AVG?....1921
POWer:MODulation:RESult:EVTCount?....1921
POWer:MODulation:RESult:NPEak?....1921
POWer:MODulation:RESult:PPEak?....1921
POWer:MODulation:RESult:RMS?....1921
POWer:MODulation:RESult:STDDev?....1921
POWer:MODulation:RESult:WFMCount?....1921
POWer:MODulation:SOURce....1922
POWer:MODulation:TYPE....1922
POWer:ONOFf:ATOFf....1929
POWer:ONOFf:ATON....1929

POWer:ONOFf:DSOFf....1929
POWer:ONOFf:DSON....1929
POWer:ONOFf:DTOFf....1929
POWer:ONOFf:DTON....1929
POWer:ONOFf:EXECute....1929
POWer:ONOFf:INPut....1929
POWer:ONOFf:REPort:ADD....1929
POWer:ONOFf:RESult:TOFF?....1930
POWer:ONOFf:RESult:TON?....1930
POWer:ONOFf:TIME....1930
POWer:ONOFf:TYPE....1930
POWer:QUALity:AUTO....1914
POWer:QUALity:EXECute....1914
POWer:QUALity:FCUS....1915
POWer:QUALity:FREQ....1914
POWer:QUALity:REPort:ADD....1915
POWer:QUALity:RESult:CURRent:CRESTfactor?....1915
POWer:QUALity:RESult:CURRent:FREQuency?....1915
POWer:QUALity:RESult:CURRent:PEAK?....1915
POWer:QUALity:RESult:CURRent:RMS?....1915
POWer:QUALity:RESult:POWer:APParent?....1915
POWer:QUALity:RESult:POWer:PFACtor?....1915
POWer:QUALity:RESult:POWer:PHASe?....1915
POWer:QUALity:RESult:POWer:REACtive?....1915
POWer:QUALity:RESult:POWer:REALpower?....1915
POWer:QUALity:RESult:VOLTage:CRESTfactor?....1915
POWer:QUALity:RESult:VOLTage:FREQuency?....1915
POWer:QUALity:RESult:VOLTage:PEAK?....1915
POWer:QUALity:RESult:VOLTage:RMS?....1915
POWer:REPort:CONTent:HSETup....1909
POWer:REPort:CONTent:MSETup....1909
POWer:REPort:CONTent:MSIGnal....1909
POWer:REPort:CONTent:RESU....1909
POWer:REPort:CONTent:SETTings....1909
POWer:REPort:CONTent:TITLe....1909
POWer:REPort:CONTent:TSETup....1909
POWer:REPort:CONTent:VSETup....1909
POWer:REPort:DESCription....1910
POWer:REPort:DUT....1910
POWer:REPort:FILE:DELete....1911
POWer:REPort:FILE:NAME....1911
POWer:REPort:FILE:NEW....1911
POWer:REPort:FILE:SAVE....1911
POWer:REPort:FONT:COLO....1910
POWer:REPort:FONT:FAMI....1910
POWer:REPort:FONT:SIZE....1910
POWer:REPort:INVert....1912
POWer:REPort:LOGO....1910
POWer:REPort:PAPersize....1910
POWer:REPort:SITE....1910

POWer:REPort:TEMPerature....1910
POWer:REPort:TEST:ADD....1911
POWer:REPort:TEST:COMMent....1913
POWer:REPort:TEST:COUNt....1913
POWer:REPort:TEST:DIRectory....1913
POWer:REPort:TEST:DSEA....1912
POWer:REPort:TEST:INSert....1911
POWer:REPort:TEST:ISE....1912
POWer:REPort:TEST:LSENd?....1914
POWer:REPort:TEST:REMove....1911
POWer:REPort:TEST:RSE....1912
POWer:REPort:TEST:SEA....1912
POWer:REPort:USER....1910
POWer:RIPPle:AUToscale....1936
POWer:RIPPle:CURRent....1936
POWer:RIPPle:EXECute....1936
POWer:RIPPle:FREQuency....1936
POWer:RIPPle:REPort:ADD....1937
POWer:RIPPle:RESult:FREQuency:AVG?....1937
POWer:RIPPle:RESult:FREQuency:EVTCount?....1937
POWer:RIPPle:RESult:FREQuency:NPEak?....1937
POWer:RIPPle:RESult:FREQuency:PPEak?....1937
POWer:RIPPle:RESult:FREQuency:RMS?....1937
POWer:RIPPle:RESult:FREQuency:STDDev?....1937
POWer:RIPPle:RESult:FREQuency:WFMCount?....1937
POWer:RIPPle:RESult:FREQuency[:ACTual]?....1937
POWer:RIPPle:RESult:MAXimum:AVG?....1937
POWer:RIPPle:RESult:MAXimum:EVTCount?....1937
POWer:RIPPle:RESult:MAXimum:NPEak?....1937
POWer:RIPPle:RESult:MAXimum:PPEak?....1937
POWer:RIPPle:RESult:MAXimum:RMS?....1937
POWer:RIPPle:RESult:MAXimum:STDDev?....1937
POWer:RIPPle:RESult:MAXimum:WFMCount?....1937
POWer:RIPPle:RESult:MAXimum[:ACTual]?....1937
POWer:RIPPle:RESult:MINimum:AVG?....1938
POWer:RIPPle:RESult:MINimum:EVTCount?....1938
POWer:RIPPle:RESult:MINimum:NPEak?....1938
POWer:RIPPle:RESult:MINimum:PPEak?....1938
POWer:RIPPle:RESult:MINimum:RMS?....1938
POWer:RIPPle:RESult:MINimum:STDDev?....1938
POWer:RIPPle:RESult:MINimum:WFMCount?....1938
POWer:RIPPle:RESult:MINimum[:ACTual]?....1938
POWer:RIPPle:RESult:NDCYcle:AVG?....1938
POWer:RIPPle:RESult:NDCYcle:EVTCount?....1938
POWer:RIPPle:RESult:NDCYcle:NPEak?....1938
POWer:RIPPle:RESult:NDCYcle:PPEak?....1938
POWer:RIPPle:RESult:NDCYcle:RMS?....1938
POWer:RIPPle:RESult:NDCYcle:STDDev?....1939
POWer:RIPPle:RESult:NDCYcle:WFMCount?....1939
POWer:RIPPle:RESult:NDCYcle[:ACTual]?....1938

POWer:RIPPle:RESult:PDCYcle:AVG?....1939
POWer:RIPPle:RESult:PDCYcle:EVTCount?....1939
POWer:RIPPle:RESult:PDCYcle:NPEak?....1939
POWer:RIPPle:RESult:PDCYcle:PPEak?....1939
POWer:RIPPle:RESult:PDCYcle:RMS?....1939
POWer:RIPPle:RESult:PDCYcle:STDDev?....1939
POWer:RIPPle:RESult:PDCYcle:WFMCount?....1939
POWer:RIPPle:RESult:PDCYcle[:ACTual]?....1939
POWer:RIPPle:RESult:PDEL:AVG?....1940
POWer:RIPPle:RESult:PDEL:EVTCount?....1940
POWer:RIPPle:RESult:PDEL:NPEak?....1940
POWer:RIPPle:RESult:PDEL:PPEak?....1940
POWer:RIPPle:RESult:PDEL:RMS?....1940
POWer:RIPPle:RESult:PDEL:STDDev?....1940
POWer:RIPPle:RESult:PDEL:WFMCount?....1940
POWer:RIPPle:RESult:PDEL[:ACTual]?....1940
POWer:RIPPle:RESult:PERiod:AVG?....1940
POWer:RIPPle:RESult:PERiod:EVTCount?....1940
POWer:RIPPle:RESult:PERiod:NPEak?....1940
POWer:RIPPle:RESult:PERiod:PPEak?....1940
POWer:RIPPle:RESult:PERiod:RMS?....1940
POWer:RIPPle:RESult:PERiod:STDDev?....1940
POWer:RIPPle:RESult:PERiod:WFMCount?....1940
POWer:RIPPle:RESult:PERiod[:ACTual]?....1940
POWer:RIPPle:RESult:STDDev:AVG?....1941
POWer:RIPPle:RESult:STDDev:EVTCount?....1941
POWer:RIPPle:RESult:STDDev:NPEak?....1941
POWer:RIPPle:RESult:STDDev:PPEak?....1941
POWer:RIPPle:RESult:STDDev:RMS?....1941
POWer:RIPPle:RESult:STDDev:STDDev?....1941
POWer:RIPPle:RESult:STDDev:WFMCount?....1941
POWer:RIPPle:RESult:STDDev[:ACTual]?....1941
POWer:SLEWrate:AUTO....1924
POWer:SLEWrate:AVGDeriv....1924
POWer:SLEWrate:EXECute....1924
POWer:SLEWrate:GATE:STARt....1925
POWer:SLEWrate:GATE:STOP....1925
POWer:SLEWrate:REPort:ADD....1925
POWer:SLEWrate:RESult:ACTual?....1925
POWer:SLEWrate:RESult:AVG?....1925
POWer:SLEWrate:RESult:EVTCount?....1925
POWer:SLEWrate:RESult:NPEak?....1925
POWer:SLEWrate:RESult:PPEak?....1925
POWer:SLEWrate:RESult:RMS?....1925
POWer:SLEWrate:RESult:STDDev?....1926
POWer:SLEWrate:RESult:WFMCount?....1926
POWer:SLEWrate:SOURce....1924
POWer:SOA:EXECute....1926
POWer:SOA:LINear:ADD....1926
POWer:SOA:LINear:COUNt?....1927

POWer:SOA:LINear:INSert....1927
POWer:SOA:LINear:POINt<m>:CURRent....1927
POWer:SOA:LINear:POINt<m>:VOLTage....1927
POWer:SOA:LINear:REMove....1927
POWer:SOA:LOGarithmic:ADD....1926
POWer:SOA:LOGarithmic:COUNt?....1927
POWer:SOA:LOGarithmic:INSert....1927
POWer:SOA:LOGarithmic:POINt<m>:CURRent....1927
POWer:SOA:LOGarithmic:POINt<m>:VOLTage....1927
POWer:SOA:LOGarithmic:REMove....1927
POWer:SOA:MASK....1928
POWer:SOA:REPort:ADD....1928
POWer:SOA:SCALe....1928
POWer:SOA:SWITch....1928
POWer:SOURce:CURRent<1..2>....1907
POWer:SOURce:VOLTage<1..4>....1907
POWer:SPECtrum:AUToscale....1944
POWer:SPECtrum:EXECute....1944
POWer:SPECtrum:FREQuency....1944
POWer:SPECtrum:RCOunt?....1945
POWer:SPECtrum:REPort:ADD....1945
POWer:SPECtrum:RESult<m>:FREQuency?....1945
POWer:SPECtrum:RESult<m>:LEVel?....1945
POWer:SWITching:AUTO....1931
POWer:SWITching:COND....1932
POWer:SWITching:EXECute....1931
POWer:SWITching:GATE:COND:STARt....1932
POWer:SWITching:GATE:COND:STOP....1932
POWer:SWITching:GATE:NCON:STARt....1932
POWer:SWITching:GATE:TOFF:STARt....1932
POWer:SWITching:GATE:TOFF:STOP....1932
POWer:SWITching:GATE:TON:STARt....1932
POWer:SWITching:GATE:TON:STOP....1932
POWer:SWITching:NCON....1932
POWer:SWITching:REPort:ADD....1931
POWer:SWITching:RESult:ENERgy:ACTual?....1932
POWer:SWITching:RESult:ENERgy:AVG?....1932
POWer:SWITching:RESult:ENERgy:EVTCount?....1932
POWer:SWITching:RESult:ENERgy:NPEak?....1932
POWer:SWITching:RESult:ENERgy:PPEak?....1932
POWer:SWITching:RESult:ENERgy:RMS?....1932
POWer:SWITching:RESult:ENERgy:STDDev?....1932
POWer:SWITching:RESult:ENERgy:WFMCount?....1932
POWer:SWITching:RESult:POWer:ACTual?....1933
POWer:SWITching:RESult:POWer:AVG?....1933
POWer:SWITching:RESult:POWer:EVTCount?....1933
POWer:SWITching:RESult:POWer:NPEak?....1933
POWer:SWITching:RESult:POWer:PPEak?....1933
POWer:SWITching:RESult:POWer:RMS?....1933
POWer:SWITching:RESult:POWer:STDDev?....1933

POWer:SWITching:RESult:POWer:WFMCount?....1933
POWer:SWITching:SWIFrequency....1931
POWer:SWITching:SWIT....1932
POWer:SWITching:TOFF....1932
POWer:SWITching:TON....1932
POWer:SWITching:TOTal....1932
POWer:TRANsient:AUToscale....1942
POWer:TRANsient:EXECute....1942
POWer:TRANsient:FREQuency....1942
POWer:TRANsient:HYSTeresis....1942
POWer:TRANsient:INPut....1942
POWer:TRANsient:REPort:ADD....1943
POWer:TRANsient:RESult[:ACTual]?....1943
POWer:TRANsient:SIGHigh....1943
POWer:TRANsient:SIGLow....1943
POWer:TRANsient:TRGChannel....1943
POWer:TRANsient:TRGLevel....1944
POWer:TRANsient:TRGSlope....1944
PROBe<m>:ID:PARTnumber?....1104
PROBe<m>:ID:PRDate?....1104
PROBe<m>:ID:SRNumber?....1105
PROBe<m>:ID:SWVersion?....1104
PROBe<m>:PMETer:RESults:COMMon?....1092
PROBe<m>:PMETer:RESults:DIFFerential?....1092
PROBe<m>:PMETer:RESults:NEGative?....1093
PROBe<m>:PMETer:RESults:POSitive?....1093
PROBe<m>:PMETer:RESults:SINGle?....1092
PROBe<m>:PMETer:VISibility....1091
PROBe<m>:SETup:ACCoupling....1096
PROBe<m>:SETup:ADAPter?....1106
PROBe<m>:SETup:ADVanced:AUDioverload....1097
PROBe<m>:SETup:ADVanced:FILTer....1097
PROBe<m>:SETup:ADVanced:PMToffset....1096
PROBe<m>:SETup:ADVanced:RANGe....1097
PROBe<m>:SETup:ATTenuation:DEFProbe....1101
PROBe<m>:SETup:ATTenuation:MANual....1094
PROBe<m>:SETup:ATTenuation:MODE....1094
PROBe<m>:SETup:ATTenuation:TDEFprobe....1106
PROBe<m>:SETup:ATTenuation:UNIT....1094
PROBe<m>:SETup:ATTenuation[:AUTO]?....1088
PROBe<m>:SETup:BANDwidth?....1087
PROBe<m>:SETup:CAPacitance?....1105
PROBe<m>:SETup:CMOFfset....1095
PROBe<m>:SETup:CMOFfset....1099
PROBe<m>:SETup:DEGauss....1103
PROBe<m>:SETup:DISPlaydiff....1091
PROBe<m>:SETup:DMOFfset....1099
PROBe<m>:SETup:GAIN:MANual....1095
PROBe<m>:SETup:IMPedance?....1105
PROBe<m>:SETup:MODE....1090

PROBe<m>:SETup:NAME?....1087
PROBe<m>:SETup:NOFFset....1099
PROBe<m>:SETup:OFFSet:AZERo....1088
PROBe<m>:SETup:OFFSet:STPRobe....1103
PROBe<m>:SETup:OFFSet:TOMean....1102
PROBe<m>:SETup:OFFSet:USEautozero....1088
PROBe<m>:SETup:OFFSet:ZADJust....1103
PROBe<m>:SETup:POFFset....1100
PROBe<m>:SETup:PRMode....1098
PROBe<m>:SETup:STATe?....1087
PROBe<m>:SETup:TERM:ADJust....1101
PROBe<m>:SETup:TERM:MEASure?....1101
PROBe<m>:SETup:TERM:MODE....1100
PROBe<m>:SETup:TERM:STATe....1100
PROBe<m>:SETup:TYPE?....1087
PROBe<m>:SETup:ZAXV....1096
PROBe<m>:SKEState....1125
REFCurve<m>:CLEar....1185
REFCurve<m>:DATA:HEADer?....1187
REFCurve<m>:DATA:STYPe?....1187
REFCurve<m>:DATA[:VALues]?....1187
REFCurve<m>:DELete....1184
REFCurve<m>:MULTichannel:IMPort....1188
REFCurve<m>:MULTichannel:NAME....1188
REFCurve<m>:MULTichannel:OPEN....1189
REFCurve<m>:NAME....1184
REFCurve<m>:OPEN....1184
REFCurve<m>:RESCale:HORizontal:OFFSet....1186
REFCurve<m>:RESCale:HORizontal:STATe....1186
REFCurve<m>:RESCale:VERTical:OFFSet....1185
REFCurve<m>:RESCale:VERTical:STATe....1185
REFCurve<m>:SAVE....1184
REFCurve<m>:SOURce....1183
REFCurve<m>:STATe....1183
REFCurve<m>:UPDate....1184
REFLevel<m>:ABSolute:BDIStance....1275
REFLevel<m>:ABSolute:HIGH....1274
REFLevel<m>:ABSolute:LOW....1274
REFLevel<m>:ABSolute:MLEVel....1275
REFLevel<m>:ABSolute:TDIStance....1275
REFLevel<m>:AUTO:COUNt....1273
REFLevel<m>:AUTO[:STATe]....1273
REFLevel<m>:LDETection....1272
REFLevel<m>:LMODe....1272
REFLevel<m>:RELative:LOWer....1277
REFLevel<m>:RELative:MIDDle....1276
REFLevel<m>:RELative:MODE....1272
REFLevel<m>:RELative:UPPer....1276
REPort:COMMent....1380
REPort:FILE:NAME....1381

REPort:FILE:SAVE.....1381
REPort:LANGuage.....1379
REPort:LOGO.....1380
REPort:LOGType.....1380
REPort:PAPersize.....1380
REPort:USER.....1380
RUN.....1070
RUNContinous.....1070
RUNSingle.....1070
SAVeset:CONFig:PREView.....1360
SAVeset:ONEFile:NAME.....1360
SAVeset:ONEFile:OPEN.....1360
SAVeset:ONEFile:SAVE.....1361
SEARch:ADD.....1315
SEARch:ALL.....1316
SEARch:CLEar.....1315
SEARch:GATE:ABSolute:STARt.....1343
SEARch:GATE:ABSolute:STOP.....1343
SEARch:GATE:MODE.....1342
SEARch:GATE:RELative:STARt.....1343
SEARch:GATE:RELative:STOP.....1344
SEARch:GATE:SHOW.....1343
SEARch:GATE:ZCOupling.....1344
SEARch:GATE:ZDIagram.....1344
SEARch:GATE[:STATe].....1342
SEARch:ONLine.....1316
SEARch:REMove.....1315
SEARch:RESDiagram:HORZ:ABSolute:POSition.....1347
SEARch:RESDiagram:HORZ:ABSolute:SPAN.....1347
SEARch:RESDiagram:HORZ:MODE.....1348
SEARch:RESDiagram:HORZ:RELative:POSition.....1348
SEARch:RESDiagram:HORZ:RELative:SPAN.....1348
SEARch:RESDiagram:SHOW.....1349
SEARch:RESDiagram:VERT:ABSolute:POSition.....1349
SEARch:RESDiagram:VERT:ABSolute:SPAN.....1349
SEARch:RESDiagram:VERT:MODE.....1349
SEARch:RESDiagram:VERT:RELative:POSition.....1350
SEARch:RESDiagram:VERT:RELative:SPAN.....1350
SEARch:RESult:ARINc:WCOunt?.....1609
SEARch:RESult:ARINc:WORD<m>:DATA?.....1608
SEARch:RESult:ARINc:WORD<m>:LABel?.....1608
SEARch:RESult:ARINc:WORD<m>:PATTern?.....1608
SEARch:RESult:ARINc:WORD<m>:SDI?.....1609
SEARch:RESult:ARINc:WORD<m>:SSM?.....1608
SEARch:RESult:ARINc:WORD<m>:STARt?.....1610
SEARch:RESult:ARINc:WORD<m>:STATe?.....1610
SEARch:RESult:ARINc:WORD<m>:STOP?.....1609
SEARch:RESult:ARINc:WORD<m>:SYMBol?.....1609
SEARch:RESult:CAN:FCOunt?.....1481
SEARch:RESult:CAN:FDATa:FRAMe<m>:SCValue?.....1484

SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard? 1484
SEARch:RESult:CAN:FRAMe<m>:ACKState? 1482
SEARch:RESult:CAN:FRAMe<m>:ACKValue? 1481
SEARch:RESult:CAN:FRAMe<m>:BSEPosition? 1481
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe? 1481
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue? 1482
SEARch:RESult:CAN:FRAMe<m>:CSSTate? 1482
SEARch:RESult:CAN:FRAMe<m>:CSValue? 1482
SEARch:RESult:CAN:FRAMe<m>:DATA? 1483
SEARch:RESult:CAN:FRAMe<m>:DLCState? 1482
SEARch:RESult:CAN:FRAMe<m>:DLCValue? 1483
SEARch:RESult:CAN:FRAMe<m>:FERCause? 1483
SEARch:RESult:CAN:FRAMe<m>:IDSTate? 1482
SEARch:RESult:CAN:FRAMe<m>:IDTYpe? 1484
SEARch:RESult:CAN:FRAMe<m>:IDValue? 1484
SEARch:RESult:CAN:FRAMe<m>:NDBYtes? 1476
SEARch:RESult:CAN:FRAMe<m>:SDATa? 1489
SEARch:RESult:CAN:FRAMe<m>:STARt? 1485
SEARch:RESult:CAN:FRAMe<m>:STATus? 1485
SEARch:RESult:CAN:FRAMe<m>:STOP? 1485
SEARch:RESult:CAN:FRAMe<m>:SYMBol? 1486
SEARch:RESult:CAN:FRAMe<m>:TYPE? 1486
SEARch:RESult:CMSB<m>:FCOunt? 1732
SEARch:RESult:CMSB<m>:FRAMe<n>:CCOunt? 1732
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:NAME? 1732
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STARt? 1733
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STATe? 1733
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STOP? 1734
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:VALue? 1734
SEARch:RESult:CMSB<m>:FRAMe<n>:CONE? 1735
SEARch:RESult:CMSB<m>:FRAMe<n>:CTHRee? 1735
SEARch:RESult:CMSB<m>:FRAMe<n>:CTWO? 1735
SEARch:RESult:CMSB<m>:FRAMe<n>:STARt? 1736
SEARch:RESult:CMSB<m>:FRAMe<n>:STATe? 1736
SEARch:RESult:CMSB<m>:FRAMe<n>:STOP? 1737
SEARch:RESult:CMSB<m>:FRAMe<n>:TYPE? 1737
SEARch:RESult:CXPI:FCOunt? 1860
SEARch:RESult:CXPI:FRAMe<m>:DATA? 1860
SEARch:RESult:CXPI:FRAMe<m>:DLCV? 1861
SEARch:RESult:CXPI:FRAMe<m>:STARt? 1861
SEARch:RESult:CXPI:FRAMe<m>:STATe? 1861
SEARch:RESult:CXPI:FRAMe<m>:STOP? 1861
SEARch:RESult:CXPI:FRAMe<m>:TYPE? 1862
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus? 1862
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE? 1862
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue? 1863
SEARch:RESult:ETHernet:WCOunt? 1630
SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue? 1634
SEARch:RESult:ETHernet:WORD<m>:CRC? 1633
SEARch:RESult:ETHernet:WORD<m>:DATA? 1632

SEARch:RESult:ETHernet:WORD<m>:DESTaddress?....1631
SEARch:RESult:ETHernet:WORD<m>:DSYMbol?....1633
SEARch:RESult:ETHernet:WORD<m>:FTYPe?....1632
SEARch:RESult:ETHernet:WORD<m>:SRCaddress?....1631
SEARch:RESult:ETHernet:WORD<m>:SSYMbol?....1633
SEARch:RESult:ETHernet:WORD<m>:STARt?....1630
SEARch:RESult:ETHernet:WORD<m>:STATe?....1630
SEARch:RESult:ETHernet:WORD<m>:STOP?....1631
SEARch:RESult:ETHernet:WORD<m>:TYPE?....1632
SEARch:RESult:FLXRay:FCOunt?....1549
SEARch:RESult:FLXRay:FRAMe<m>:ADID?....1549
SEARch:RESult:FLXRay:FRAMe<m>:CSSTate?....1549
SEARch:RESult:FLXRay:FRAMe<m>:CSValue?....1550
SEARch:RESult:FLXRay:FRAMe<m>:CYCount?....1550
SEARch:RESult:FLXRay:FRAMe<m>:DATA?....1550
SEARch:RESult:FLXRay:FRAMe<m>:FCSTate?....1550
SEARch:RESult:FLXRay:FRAMe<m>:FCValue?....1551
SEARch:RESult:FLXRay:FRAMe<m>:FLAGs?....1551
SEARch:RESult:FLXRay:FRAMe<m>:PAYLength?....1551
SEARch:RESult:FLXRay:FRAMe<m>:STARt?....1552
SEARch:RESult:FLXRay:FRAMe<m>:STATus?....1552
SEARch:RESult:FLXRay:FRAMe<m>:STOP?....1553
SEARch:RESult:FLXRay:FRAMe<m>:SYMBol?....1553
SEARch:RESult:FLXRay:FRAMe<m>:TYPE?....1553
SEARch:RESult:HBTO:FCOunt?....1659
SEARch:RESult:HBTO:FRAMe<m>:CRC?....1662
SEARch:RESult:HBTO:FRAMe<m>:DATA?....1661
SEARch:RESult:HBTO:FRAMe<m>:DESTaddress?....1661
SEARch:RESult:HBTO:FRAMe<m>:DSYMbol?....1663
SEARch:RESult:HBTO:FRAMe<m>:FTYPe?....1662
SEARch:RESult:HBTO:FRAMe<m>:NUMWords?....1662
SEARch:RESult:HBTO:FRAMe<m>:SRCaddress?....1661
SEARch:RESult:HBTO:FRAMe<m>:SSYMbol?....1663
SEARch:RESult:HBTO:FRAMe<m>:STARt?....1660
SEARch:RESult:HBTO:FRAMe<m>:STATe?....1660
SEARch:RESult:HBTO:FRAMe<m>:STOP?....1660
SEARch:RESult:HBTO:FRAMe<m>:TYPE?....1659
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE?....1663
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue?....1664
SEARch:RESult:I2C:FCOunt?....1414
SEARch:RESult:I2C:FRAMe<m>:AACCess?....1415
SEARch:RESult:I2C:FRAMe<m>:ACCess?....1415
SEARch:RESult:I2C:FRAMe<m>:ACOMplete?....1416
SEARch:RESult:I2C:FRAMe<m>:ADBStart?....1416
SEARch:RESult:I2C:FRAMe<m>:ADDRess?....1416
SEARch:RESult:I2C:FRAMe<m>:ADEVice?....1416
SEARch:RESult:I2C:FRAMe<m>:AMODe?....1417
SEARch:RESult:I2C:FRAMe<m>:ASTart?....1417
SEARch:RESult:I2C:FRAMe<m>:BCOunt?....1418
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess?....1418

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart?....1419
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete?....1419
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt?....1419
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue?....1420
SEARch:RESult:I2C:FRAMe<m>:DATA?....1417
SEARch:RESult:I2C:FRAMe<m>:RWBStart?....1418
SEARch:RESult:I2C:FRAMe<m>:STARt?....1414
SEARch:RESult:I2C:FRAMe<m>:STATus?....1414
SEARch:RESult:I2C:FRAMe<m>:STOP?....1415
SEARch:RESult:I2C:FRAMe<m>:SYMBol?....1418
SEARch:RESult:LIMit....1350
SEARch:RESult:LIN:FCOunt?....1515
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe?....1519
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue?....1519
SEARch:RESult:LIN:FRAMe<m>:CSSTate?....1516
SEARch:RESult:LIN:FRAMe<m>:CSValue?....1516
SEARch:RESult:LIN:FRAMe<m>:DATA?....1516
SEARch:RESult:LIN:FRAMe<m>:IDPValue?....1517
SEARch:RESult:LIN:FRAMe<m>:IDSTate?....1517
SEARch:RESult:LIN:FRAMe<m>:IDValue?....1517
SEARch:RESult:LIN:FRAMe<m>:STARt?....1515
SEARch:RESult:LIN:FRAMe<m>:STATus?....1515
SEARch:RESult:LIN:FRAMe<m>:STOP?....1515
SEARch:RESult:LIN:FRAMe<m>:SYMBol?....1518
SEARch:RESult:LIN:FRAMe<m>:SYSTate?....1518
SEARch:RESult:LIN:FRAMe<m>:VERSion?....1518
SEARch:RESult:MDIO:WCOunt?....1752
SEARch:RESult:MDIO:WORD<m>:DATA?....1749
SEARch:RESult:MDIO:WORD<m>:PHYS?....1749
SEARch:RESult:MDIO:WORD<m>:REGI?....1750
SEARch:RESult:MDIO:WORD<m>:ST?....1750
SEARch:RESult:MDIO:WORD<m>:STARt?....1750
SEARch:RESult:MDIO:WORD<m>:STATe?....1751
SEARch:RESult:MDIO:WORD<m>:STOP?....1751
SEARch:RESult:MDIO:WORD<m>:SYMBol?....1752
SEARch:RESult:MDIO:WORD<m>:TYPE?....1752
SEARch:RESult:MILStd:WCOunt?....1592
SEARch:RESult:MILStd:WORD<m>:DATA?....1594
SEARch:RESult:MILStd:WORD<m>:INFO?....1594
SEARch:RESult:MILStd:WORD<m>:RTADdress?....1593
SEARch:RESult:MILStd:WORD<m>:STARt?....1592
SEARch:RESult:MILStd:WORD<m>:STATus?....1592
SEARch:RESult:MILStd:WORD<m>:STOP?....1593
SEARch:RESult:MILStd:WORD<m>:SYMBol?....1593
SEARch:RESult:MILStd:WORD<m>:TYPE?....1592
SEARch:RESult:SENT:FCOunt?....1689
SEARch:RESult:SENT:FRAMe<m>:CSValue?....1690
SEARch:RESult:SENT:FRAMe<m>:DATA?....1690
SEARch:RESult:SENT:FRAMe<m>:IDTYpe?....1690
SEARch:RESult:SENT:FRAMe<m>:IDValue?....1690

SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe?....1691
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue?....1691
SEARch:RESult:SENT:FRAMe<m>:PAPTicks?....1691
SEARch:RESult:SENT:FRAMe<m>:SCOM?....1692
SEARch:RESult:SENT:FRAMe<m>:SDATa?....1692
SEARch:RESult:SENT:FRAMe<m>:STARt?....1692
SEARch:RESult:SENT:FRAMe<m>:STATus?....1693
SEARch:RESult:SENT:FRAMe<m>:STOP?....1693
SEARch:RESult:SENT:FRAMe<m>:SYMBol?....1693
SEARch:RESult:SENT:FRAMe<m>:SYNCduration?....1693
SEARch:RESult:SENT:FRAMe<m>:TYPE?....1694
SEARch:RESult:SHOW....1351
SEARch:RESult:SORT:ASCending....1351
SEARch:RESult:SORT[:MODE]....1351
SEARch:RESult:SPI:FCOunt?....1436
SEARch:RESult:SPI:FRAMe<m>:COUNt?....1437
SEARch:RESult:SPI:FRAMe<m>:DATA?....1437
SEARch:RESult:SPI:FRAMe<m>:STARt?....1437
SEARch:RESult:SPI:FRAMe<m>:STATus?....1437
SEARch:RESult:SPI:FRAMe<m>:STOP?....1438
SEARch:RESult:SPI:FRAMe<m>:WCOunt?....1438
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MISO?....1438
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MOSI?....1439
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STARt?....1439
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STOP?....1439
SEARch:RESult:SWIRe:FCOunt?....1833
SEARch:RESult:SWIRe:FRAMe<m>:DATA?....1833
SEARch:RESult:SWIRe:FRAMe<m>:STARt?....1834
SEARch:RESult:SWIRe:FRAMe<m>:STATe?....1834
SEARch:RESult:SWIRe:FRAMe<m>:STOP?....1834
SEARch:RESult:SWIRe:FRAMe<m>:TYPE?....1835
SEARch:RESult:USB:PACKet<m>:ADDRess?....1789
SEARch:RESult:USB:PACKet<m>:CRC?....1789
SEARch:RESult:USB:PACKet<m>:DATA?....1790
SEARch:RESult:USB:PACKet<m>:ENDPoint?....1790
SEARch:RESult:USB:PACKet<m>:ET?....1790
SEARch:RESult:USB:PACKet<m>:FRAMe?....1790
SEARch:RESult:USB:PACKet<m>:PID?....1791
SEARch:RESult:USB:PACKet<m>:PORT?....1791
SEARch:RESult:USB:PACKet<m>:SC?....1792
SEARch:RESult:USB:PACKet<m>:SEU?....1792
SEARch:RESult:USB:PACKet<m>:STARt?....1792
SEARch:RESult:USB:PACKet<m>:STATus?....1793
SEARch:RESult:USB:PACKet<m>:STOP?....1793
SEARch:RESult:USB:PCOunt?....1793
SEARch:RESult:USBPd:FCOunt?....1813
SEARch:RESult:USBPd:FRAMe<m>:DATA?....1813
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL?....1814
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt?....1815
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus?....1814

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP?....1815
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE?....1815
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL?....1816
SEARch:RESult:USBPd:FRAMe<m>:INFO?....1816
SEARch:RESult:USBPd:FRAMe<m>:STARt?....1816
SEARch:RESult:USBPd:FRAMe<m>:STATe?....1817
SEARch:RESult:USBPd:FRAMe<m>:STOP?....1817
SEARch:RESult:USBPd:FRAMe<m>:TYPE?....1817
SEARch:RESult[:ALL]?....1352
SEARch:SOURce....1315
SEARch:TRIGger:ARINc:DATA:CONDition....1605
SEARch:TRIGger:ARINc:DATA:MAX....1606
SEARch:TRIGger:ARINc:DATA:MIN....1605
SEARch:TRIGger:ARINc:ERRor:CODing....1606
SEARch:TRIGger:ARINc:ERRor:PARity....1607
SEARch:TRIGger:ARINc:ERRor:TIMing....1607
SEARch:TRIGger:ARINc:LABel:CONDition....1605
SEARch:TRIGger:ARINc:LABel:MAX....1606
SEARch:TRIGger:ARINc:LABel:MIN....1605
SEARch:TRIGger:ARINc:SDI....1606
SEARch:TRIGger:ARINc:SSM....1606
SEARch:TRIGger:ARINc:TYPE....1605
SEARch:TRIGger:CAN:ACKerror....1478
SEARch:TRIGger:CAN:BITSterror....1478
SEARch:TRIGger:CAN:CRCerror....1478
SEARch:TRIGger:CAN:DCONdition....1475
SEARch:TRIGger:CAN:DLC....1476
SEARch:TRIGger:CAN:DLCCondition....1476
SEARch:TRIGger:CAN:DMAX....1476
SEARch:TRIGger:CAN:DMIN....1475
SEARch:TRIGger:CAN:FDATa:BRS....1479
SEARch:TRIGger:CAN:FDATa:DPOPerator....1477
SEARch:TRIGger:CAN:FDATa:DPOSition....1477
SEARch:TRIGger:CAN:FDATa:DPTO....1478
SEARch:TRIGger:CAN:FDATa:ESI....1480
SEARch:TRIGger:CAN:FDATa:SCERror....1479
SEARch:TRIGger:CAN:FDATa:STANdard....1473
SEARch:TRIGger:CAN:FDATa[:FDF]....1479
SEARch:TRIGger:CAN:FORMerror....1479
SEARch:TRIGger:CAN:FTYPe....1473
SEARch:TRIGger:CAN:ICONdition....1474
SEARch:TRIGger:CAN:IMAX....1474
SEARch:TRIGger:CAN:IMIN....1475
SEARch:TRIGger:CAN:ITYPe....1474
SEARch:TRIGger:CAN:SERRor....1473
SEARch:TRIGger:CAN:SFIDentifier....1472
SEARch:TRIGger:CAN:SFTYpe....1472
SEARch:TRIGger:CAN:SIDData....1473
SEARch:TRIGger:CAN:SSYMbolic....1489
SEARch:TRIGger:CAN:SYMBolic:DMAX....1490

SEARch:TRIGger:CAN:SYMBolic:DMIN....1490
SEARch:TRIGger:CAN:SYMBolic:MSGValue....1489
SEARch:TRIGger:CAN:SYMBolic:SGEValue....1491
SEARch:TRIGger:CAN:SYMBolic:SIGValue....1490
SEARch:TRIGger:CAN:SYMBolic:SSIGnals....1490
SEARch:TRIGger:CAN[:SSOFrame]....1472
SEARch:TRIGger:CMSB:BIT....1728
SEARch:TRIGger:CMSB:DMAX....1728
SEARch:TRIGger:CMSB:DMIN....1729
SEARch:TRIGger:CMSB:DOPerator....1729
SEARch:TRIGger:CMSB:ERENable....1727
SEARch:TRIGger:CMSB:ERRor<m>:ENABle....1727
SEARch:TRIGger:CMSB:FIENable....1727
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle....1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT....1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX....1728
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN....1729
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator....1729
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle....1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX....1730
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN....1731
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator....1731
SEARch:TRIGger:CMSB:FRENable....1728
SEARch:TRIGger:CMSB:IMAX....1730
SEARch:TRIGger:CMSB:IMIN....1731
SEARch:TRIGger:CMSB:IOPerator....1731
SEARch:TRIGger:CXPI:CT....1852
SEARch:TRIGger:CXPI:DATA:DCONdition....1852
SEARch:TRIGger:CXPI:DATA:DMAX....1852
SEARch:TRIGger:CXPI:DATA:DMIN....1853
SEARch:TRIGger:CXPI:DATA:ICONdition....1853
SEARch:TRIGger:CXPI:DATA:IMAX....1853
SEARch:TRIGger:CXPI:DATA:IMIN....1854
SEARch:TRIGger:CXPI:DEXTension:CONDition....1854
SEARch:TRIGger:CXPI:DEXTension:MAX....1854
SEARch:TRIGger:CXPI:DEXTension:MIN....1855
SEARch:TRIGger:CXPI:DLC:CONDition....1855
SEARch:TRIGger:CXPI:DLC:MAX....1855
SEARch:TRIGger:CXPI:DLC:MIN....1855
SEARch:TRIGger:CXPI:ERRor:CRC....1856
SEARch:TRIGger:CXPI:ERRor:DLC....1856
SEARch:TRIGger:CXPI:ERRor:IBS....1856
SEARch:TRIGger:CXPI:ERRor:IFS....1856
SEARch:TRIGger:CXPI:ERRor:LENGth....1857
SEARch:TRIGger:CXPI:ERRor:PARity....1857
SEARch:TRIGger:CXPI:ERRor:UART....1857
SEARch:TRIGger:CXPI:FID:CONDition....1857
SEARch:TRIGger:CXPI:FID:MAX....1858
SEARch:TRIGger:CXPI:FID:MIN....1858
SEARch:TRIGger:CXPI:NM....1858

SEARch:TRIGger:CXPI:TYPE....1859
SEARch:TRIGger:DATatoclock:ACOPy....1318
SEARch:TRIGger:DATatoclock:CEDGe....1332
SEARch:TRIGger:DATatoclock:CLEVel....1333
SEARch:TRIGger:DATatoclock:CSOurce....1333
SEARch:TRIGger:DATatoclock:HTIMe....1333
SEARch:TRIGger:DATatoclock:STIMe....1334
SEARch:TRIGger:DATatoclock[:STATe]....1317
SEARch:TRIGger:EDGE:ACOPy....1318
SEARch:TRIGger:EDGE:BCOPy....1318
SEARch:TRIGger:EDGE:SLOPe....1319
SEARch:TRIGger:EDGE[:STATe]....1317
SEARch:TRIGger:ETHernet:ERRor:LENGth....1629
SEARch:TRIGger:ETHernet:ERRor:PREamble....1629
SEARch:TRIGger:ETHernet:ERRor:SELect....1629
SEARch:TRIGger:ETHernet:FRAMe:CCONdition....1628
SEARch:TRIGger:ETHernet:FRAMe:CMAX....1628
SEARch:TRIGger:ETHernet:FRAMe:CMIN....1628
SEARch:TRIGger:ETHernet:FRAMe:DCONdition....1625
SEARch:TRIGger:ETHernet:FRAMe:DMAX....1625
SEARch:TRIGger:ETHernet:FRAMe:DMIN....1625
SEARch:TRIGger:ETHernet:FRAMe:SCONdition....1626
SEARch:TRIGger:ETHernet:FRAMe:SELect....1624
SEARch:TRIGger:ETHernet:FRAMe:SMAX....1626
SEARch:TRIGger:ETHernet:FRAMe:SMIN....1626
SEARch:TRIGger:ETHernet:FRAMe:TCONdition....1627
SEARch:TRIGger:ETHernet:FRAMe:TMAX....1627
SEARch:TRIGger:ETHernet:FRAMe:TMIN....1627
SEARch:TRIGger:FLXRay:BSSerror....1547
SEARch:TRIGger:FLXRay:CENable....1540
SEARch:TRIGger:FLXRay:CMAX....1540
SEARch:TRIGger:FLXRay:CMIN....1541
SEARch:TRIGger:FLXRay:CSTep....1541
SEARch:TRIGger:FLXRay:DCONdition....1541
SEARch:TRIGger:FLXRay:DMAX....1542
SEARch:TRIGger:FLXRay:DMIN....1542
SEARch:TRIGger:FLXRay:DPOPerator....1542
SEARch:TRIGger:FLXRay:DPOSition....1543
SEARch:TRIGger:FLXRay:DPTO....1543
SEARch:TRIGger:FLXRay:FCONdition....1543
SEARch:TRIGger:FLXRay:FESerror....1547
SEARch:TRIGger:FLXRay:FMAX....1544
SEARch:TRIGger:FLXRay:FMIN....1544
SEARch:TRIGger:FLXRay:FSSerror....1547
SEARch:TRIGger:FLXRay:HCRCerror....1548
SEARch:TRIGger:FLXRay:NUFRame....1544
SEARch:TRIGger:FLXRay:PCONdition....1545
SEARch:TRIGger:FLXRay:PCRCerror....1548
SEARch:TRIGger:FLXRay:PLPReamble....1545
SEARch:TRIGger:FLXRay:PMAX....1545

SEARch:TRIGger:FLXRay:PMIN....1546
SEARch:TRIGger:FLXRay:SERRor....1539
SEARch:TRIGger:FLXRay:SIDData....1539
SEARch:TRIGger:FLXRay:SSYMbol....1540
SEARch:TRIGger:FLXRay:STFRame....1546
SEARch:TRIGger:FLXRay:SYFRame....1546
SEARch:TRIGger:FLXRay:SYMBol....1547
SEARch:TRIGger:FLXRay[:SSOFrame]....1539
SEARch:TRIGger:GLITch:ACOPy....1318
SEARch:TRIGger:GLITch:POLarity....1319
SEARch:TRIGger:GLITch:RANGe....1320
SEARch:TRIGger:GLITch:WIDTh....1320
SEARch:TRIGger:GLITch[:STATe]....1317
SEARch:TRIGger:HBTO:CRC:CONDition....1655
SEARch:TRIGger:HBTO:CRC:MAX....1656
SEARch:TRIGger:HBTO:CRC:MIN....1655
SEARch:TRIGger:HBTO:DADDress:CONDition....1652
SEARch:TRIGger:HBTO:DADDress:MAX....1653
SEARch:TRIGger:HBTO:DADDress:MIN....1652
SEARch:TRIGger:HBTO:DATA:DCONdition....1656
SEARch:TRIGger:HBTO:DATA:DMAX....1657
SEARch:TRIGger:HBTO:DATA:DMIN....1656
SEARch:TRIGger:HBTO:DATA:ICONdition....1657
SEARch:TRIGger:HBTO:DATA:IMAX....1658
SEARch:TRIGger:HBTO:DATA:IMIN....1657
SEARch:TRIGger:HBTO:ERRor:CRC....1658
SEARch:TRIGger:HBTO:ERRor:PREamble....1658
SEARch:TRIGger:HBTO:ERRor:SFD....1658
SEARch:TRIGger:HBTO:LENGth:CONDition....1654
SEARch:TRIGger:HBTO:LENGth:MAX....1655
SEARch:TRIGger:HBTO:LENGth:MIN....1654
SEARch:TRIGger:HBTO:SADDress:CONDition....1653
SEARch:TRIGger:HBTO:SADDress:MAX....1654
SEARch:TRIGger:HBTO:SADDress:MIN....1653
SEARch:TRIGger:HBTO:TYPE....1651
SEARch:TRIGger:I2C:ACCess....1409
SEARch:TRIGger:I2C:ACONdition....1407
SEARch:TRIGger:I2C:ADData....1407
SEARch:TRIGger:I2C:ADDO<m>:ADRType....1409
SEARch:TRIGger:I2C:ADDO<m>:ENABle....1409
SEARch:TRIGger:I2C:ADDO<m>:RWBit....1410
SEARch:TRIGger:I2C:ADDO<m>[:VALue]....1410
SEARch:TRIGger:I2C:ADDRess....1408
SEARch:TRIGger:I2C:ADDTo....1408
SEARch:TRIGger:I2C:ADNack....1412
SEARch:TRIGger:I2C:ADOR....1407
SEARch:TRIGger:I2C:AMODe....1408
SEARch:TRIGger:I2C:DCONdition....1412
SEARch:TRIGger:I2C:DMAX....1412
SEARch:TRIGger:I2C:DMIN....1412

SEARch:TRIGger:I2C:DPOPerator....1411
SEARch:TRIGger:I2C:DPOSition....1411
SEARch:TRIGger:I2C:DPTO....1411
SEARch:TRIGger:I2C:DRNack....1413
SEARch:TRIGger:I2C:DWNack....1413
SEARch:TRIGger:I2C:NACKnowledge....1406
SEARch:TRIGger:I2C:RCONdition....1405
SEARch:TRIGger:I2C:SADDress....1406
SEARch:TRIGger:I2C:SCONdition....1405
SEARch:TRIGger:I2C:STCNdition....1406
SEARch:TRIGger:INTerval:ACOPy....1318
SEARch:TRIGger:INTerval:DELTa....1321
SEARch:TRIGger:INTerval:RANGe....1321
SEARch:TRIGger:INTerval:SLOPe....1320
SEARch:TRIGger:INTerval:WIDTh....1322
SEARch:TRIGger:INTerval[:STATe]....1317
SEARch:TRIGger:LEVel:NOISe:ABSolute....1345
SEARch:TRIGger:LEVel:NOISe:MODE....1345
SEARch:TRIGger:LEVel:NOISe:RELative....1346
SEARch:TRIGger:LEVel:NOISe[:STATe]....1346
SEARch:TRIGger:LEVel:RUNT:LOWer....1324
SEARch:TRIGger:LEVel:RUNT:UPPer....1324
SEARch:TRIGger:LEVel:TRANsition:LOWer....1326
SEARch:TRIGger:LEVel:TRANsition:UPPer....1326
SEARch:TRIGger:LEVel:WINDow:LOWer....1332
SEARch:TRIGger:LEVel:WINDow:UPPer....1332
SEARch:TRIGger:LEVel[:VALue]....1318
SEARch:TRIGger:LIN:BORDer....1511
SEARch:TRIGger:LIN:CHKSerror....1513
SEARch:TRIGger:LIN:CRCDatalen....1514
SEARch:TRIGger:LIN:DCONdition....1510
SEARch:TRIGger:LIN:DLECondition....1512
SEARch:TRIGger:LIN:DLENgth....1512
SEARch:TRIGger:LIN:DMAX....1511
SEARch:TRIGger:LIN:DMIN....1511
SEARch:TRIGger:LIN:ERRPattern....1513
SEARch:TRIGger:LIN:ICONdition....1509
SEARch:TRIGger:LIN:IDENtifieror....1508
SEARch:TRIGger:LIN:IDOR<m>:ENABle....1510
SEARch:TRIGger:LIN:IDOR<m>[:VALue]....1510
SEARch:TRIGger:LIN:IMAX....1509
SEARch:TRIGger:LIN:IMIN....1509
SEARch:TRIGger:LIN:IPERror....1512
SEARch:TRIGger:LIN:SERRor....1508
SEARch:TRIGger:LIN:SFIDentifier....1508
SEARch:TRIGger:LIN:SIDData....1508
SEARch:TRIGger:LIN:SSOFrame....1507
SEARch:TRIGger:LIN:STANdard....1514
SEARch:TRIGger:LIN:SYERror....1513
SEARch:TRIGger:LIN:WUFRame....1509

SEARch:TRIGger:MDIO:DATA....1747
SEARch:TRIGger:MDIO:FRAMetype....1747
SEARch:TRIGger:MDIO:PHYS....1748
SEARch:TRIGger:MDIO:REGI....1748
SEARch:TRIGger:MDIO:ST....1748
SEARch:TRIGger:MDIO:TYPE....1748
SEARch:TRIGger:MILStd:CDST:ICONdition....1587
SEARch:TRIGger:MILStd:CDST:IMAX....1588
SEARch:TRIGger:MILStd:CDST:IMIN....1588
SEARch:TRIGger:MILStd:CDST:RCONdition....1586
SEARch:TRIGger:MILStd:CDST:RMAX....1587
SEARch:TRIGger:MILStd:CDST:RMIN....1587
SEARch:TRIGger:MILStd:CMD:CCONdition....1587
SEARch:TRIGger:MILStd:CMD:CMAX....1588
SEARch:TRIGger:MILStd:CMD:CMIN....1588
SEARch:TRIGger:MILStd:CMD:RCONdition....1586
SEARch:TRIGger:MILStd:CMD:RMAX....1587
SEARch:TRIGger:MILStd:CMD:RMIN....1587
SEARch:TRIGger:MILStd:CMD:SCONdition....1588
SEARch:TRIGger:MILStd:CMD:SMAX....1589
SEARch:TRIGger:MILStd:CMD:SMIN....1588
SEARch:TRIGger:MILStd:CMD:TR....1590
SEARch:TRIGger:MILStd:DATA:DCONdition....1588
SEARch:TRIGger:MILStd:DATA:DMAX....1589
SEARch:TRIGger:MILStd:DATA:DMIN....1588
SEARch:TRIGger:MILStd:DATA:ICONdition....1589
SEARch:TRIGger:MILStd:DATA:IMAX....1589
SEARch:TRIGger:MILStd:DATA:IMIN....1589
SEARch:TRIGger:MILStd:DATA:RCONdition....1586
SEARch:TRIGger:MILStd:DATA:RMAX....1587
SEARch:TRIGger:MILStd:DATA:RMIN....1587
SEARch:TRIGger:MILStd:ERRor:MANChester....1590
SEARch:TRIGger:MILStd:ERRor:PARity....1590
SEARch:TRIGger:MILStd:ERRor:SYNC....1590
SEARch:TRIGger:MILStd:ERRor:TIMing....1590
SEARch:TRIGger:MILStd:STATus:BCReceived....1590
SEARch:TRIGger:MILStd:STATus:BUSY....1590
SEARch:TRIGger:MILStd:STATus:DBCaccept....1591
SEARch:TRIGger:MILStd:STATus:INSTrument....1591
SEARch:TRIGger:MILStd:STATus:MERRor....1591
SEARch:TRIGger:MILStd:STATus:SREQuest....1591
SEARch:TRIGger:MILStd:STATus:SUBSystem....1591
SEARch:TRIGger:MILStd:STATus:TERMinal....1591
SEARch:TRIGger:MILStd:TPSPecifier....1591
SEARch:TRIGger:MILStd:TYPE....1586
SEARch:TRIGger:PATTern:A:LOGic....1335
SEARch:TRIGger:PATTern:A[:ENABle]....1335
SEARch:TRIGger:PATTern:AB:LOGic....1336
SEARch:TRIGger:PATTern:ABCD:LOGic....1336
SEARch:TRIGger:PATTern:ACOPy....1318

SEARch:TRIGger:PATTern:B:LOGic........ 1335
SEARch:TRIGger:PATTern:B[:ENABle]........ 1335
SEARch:TRIGger:PATTern:C:LOGic........ 1335
SEARch:TRIGger:PATTern:C[:ENABle]........ 1335
SEARch:TRIGger:PATTern:CD:LOGic........ 1336
SEARch:TRIGger:PATTern:D:LOGic........ 1335
SEARch:TRIGger:PATTern:D[:ENABle]........ 1335
SEARch:TRIGger:PATTern:MODE........ 1336
SEARch:TRIGger:PATTern:TIMeout:MODE........ 1337
SEARch:TRIGger:PATTern:TIMeout[:TIME]........ 1337
SEARch:TRIGger:PATTern:WIDTh:DELTa........ 1338
SEARch:TRIGger:PATTern:WIDTh:RANGe........ 1337
SEARch:TRIGger:PATTern:WIDTh[:WIDTh]........ 1338
SEARch:TRIGger:PATTern[:STATe]........ 1317
SEARch:TRIGger:RUNT:ACOPy........ 1318
SEARch:TRIGger:RUNT:DELTa........ 1322
SEARch:TRIGger:RUNT:POLarity........ 1322
SEARch:TRIGger:RUNT:RANGe........ 1323
SEARch:TRIGger:RUNT:WIDTh........ 1323
SEARch:TRIGger:RUNT[:STATe]........ 1317
SEARch:TRIGger:SENT:CALibration........ 1681
SEARch:TRIGger:SENT:CRCerror........ 1688
SEARch:TRIGger:SENT:ERRor........ 1682
SEARch:TRIGger:SENT:FORMerror........ 1688
SEARch:TRIGger:SENT:IRFLength........ 1688
SEARch:TRIGger:SENT:PPERioderror........ 1688
SEARch:TRIGger:SENT:PULSeerror........ 1687
SEARch:TRIGger:SENT:SDCN........ 1686
SEARch:TRIGger:SENT:SDMN........ 1687
SEARch:TRIGger:SENT:SDMX........ 1687
SEARch:TRIGger:SENT:SICN........ 1685
SEARch:TRIGger:SENT:SIDType........ 1685
SEARch:TRIGger:SENT:SIMN........ 1686
SEARch:TRIGger:SENT:SIMX........ 1686
SEARch:TRIGger:SENT:SMSG........ 1682
SEARch:TRIGger:SENT:STATus........ 1683
SEARch:TRIGger:SENT:STYPe........ 1685
SEARch:TRIGger:SENT:TDCN........ 1684
SEARch:TRIGger:SENT:TDMN........ 1684
SEARch:TRIGger:SENT:TDMX........ 1684
SEARch:TRIGger:SENT:TRANsmission........ 1682
SEARch:TRIGger:SENT:TTYPe........ 1683
SEARch:TRIGger:SLEWrate:ACOPy........ 1318
SEARch:TRIGger:SLEWrate:DELTa........ 1324
SEARch:TRIGger:SLEWrate:RANGe........ 1325
SEARch:TRIGger:SLEWrate:SLOPe........ 1325
SEARch:TRIGger:SLEWrate:TIME........ 1326
SEARch:TRIGger:SLEWrate[:STATe]........ 1317
SEARch:TRIGger:SPI:DPOPerator........ 1434
SEARch:TRIGger:SPI:DPOSition........ 1435

SEARch:TRIGger:SPI:DPTO....1435
SEARch:TRIGger:SPI:FCONdition....1434
SEARch:TRIGger:SPI:MISopattern....1434
SEARch:TRIGger:SPI:MODE....1433
SEARch:TRIGger:SPI:MOSipattern....1434
SEARch:TRIGger:SPI:PALignment....1435
SEARch:TRIGger:STATe:A:LOGic....1340
SEARch:TRIGger:STATe:A[:ENABle]....1340
SEARch:TRIGger:STATe:AB:LOGic....1341
SEARch:TRIGger:STATe:ABCD:LOGic....1341
SEARch:TRIGger:STATe:ACOPy....1318
SEARch:TRIGger:STATe:B:LOGic....1340
SEARch:TRIGger:STATe:B[:ENABle]....1340
SEARch:TRIGger:STATe:C:LOGic....1340
SEARch:TRIGger:STATe:C[:ENABle]....1340
SEARch:TRIGger:STATe:CD:LOGic....1341
SEARch:TRIGger:STATe:CEDGe....1339
SEARch:TRIGger:STATe:CLEVel....1339
SEARch:TRIGger:STATe:CSOurce....1339
SEARch:TRIGger:STATe:D:LOGic....1340
SEARch:TRIGger:STATe:D[:ENABle]....1340
SEARch:TRIGger:STATe[:STATe]....1317
SEARch:TRIGger:SWIRe:CTYPe....1830
SEARch:TRIGger:SWIRe:DATA:CONDition....1830
SEARch:TRIGger:SWIRe:DATA:MAX....1831
SEARch:TRIGger:SWIRe:DATA:MIN....1831
SEARch:TRIGger:SWIRe:ERRor:ESC....1831
SEARch:TRIGger:SWIRe:ERRor:PARity....1831
SEARch:TRIGger:SWIRe:TIME:CONDition....1832
SEARch:TRIGger:SWIRe:TIME:MAX....1832
SEARch:TRIGger:SWIRe:TIME:MIN....1832
SEARch:TRIGger:SWIRe:TYPE....1833
SEARch:TRIGger:TIMeout:ACOPy....1318
SEARch:TRIGger:TIMeout:RANGe....1327
SEARch:TRIGger:TIMeout:TIME....1327
SEARch:TRIGger:TIMeout[:STATe]....1317
SEARch:TRIGger:USB:ACONdition....1775
SEARch:TRIGger:USB:AMAX....1775
SEARch:TRIGger:USB:AMIN....1775
SEARch:TRIGger:USB:BITSterror....1775
SEARch:TRIGger:USB:CRC5error....1776
SEARch:TRIGger:USB:CRC16error....1776
SEARch:TRIGger:USB:DATA....1776
SEARch:TRIGger:USB:DCONdition....1777
SEARch:TRIGger:USB:DPOPerator....1777
SEARch:TRIGger:USB:DPOSition....1777
SEARch:TRIGger:USB:ECONdition....1778
SEARch:TRIGger:USB:EMAX....1778
SEARch:TRIGger:USB:EMIN....1778
SEARch:TRIGger:USB:FCONdition....1779

SEARch:TRIGger:USB:FMAX....1779
SEARch:TRIGger:USB:FMIN....1779
SEARch:TRIGger:USB:GLITcherror....1779
SEARch:TRIGger:USB:HAND....1780
SEARch:TRIGger:USB:PATT....1780
SEARch:TRIGger:USB:PCONdition....1780
SEARch:TRIGger:USB:PIDerror....1781
SEARch:TRIGger:USB:PMAX....1781
SEARch:TRIGger:USB:PMIN....1781
SEARch:TRIGger:USB:SCONdition....1781
SEARch:TRIGger:USB:SDATa....1782
SEARch:TRIGger:USB:SERRor....1783
SEARch:TRIGger:USB:SHANdshake....1783
SEARch:TRIGger:USB:SMAX....1782
SEARch:TRIGger:USB:SMIN....1782
SEARch:TRIGger:USB:SPEC....1784
SEARch:TRIGger:USB:SSOP....1783
SEARch:TRIGger:USB:SSPE....1783
SEARch:TRIGger:USB:STCO....1784
SEARch:TRIGger:USB:STOKen....1785
SEARch:TRIGger:USB:TCONdition....1785
SEARch:TRIGger:USB:TMAX....1786
SEARch:TRIGger:USB:TMIN....1785
SEARch:TRIGger:USB:TOKen....1786
SEARch:TRIGger:USB:WADD....1786
SEARch:TRIGger:USB:WEND....1787
SEARch:TRIGger:USB:WETCheck....1787
SEARch:TRIGger:USB:WFRN....1787
SEARch:TRIGger:USB:WPAY....1787
SEARch:TRIGger:USB:WPID....1788
SEARch:TRIGger:USB:WPOR....1788
SEARch:TRIGger:USB:WSEU....1788
SEARch:TRIGger:USB:WSTC....1788
SEARch:TRIGger:USBPd:BIT....1809
SEARch:TRIGger:USBPd:DMAX....1810
SEARch:TRIGger:USBPd:DMIN....1810
SEARch:TRIGger:USBPd:DOPerator....1811
SEARch:TRIGger:USBPd:ERENable....1808
SEARch:TRIGger:USBPd:ERRor<m>:ENABle....1808
SEARch:TRIGger:USBPd:FIENable....1809
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle....1808
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT....1809
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX....1810
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN....1810
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator....1811
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle....1809
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX....1811
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN....1812
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator....1812
SEARch:TRIGger:USBPd:FRENable....1808

SEARch:TRIGger:USBPd:IMAX....1811
SEARch:TRIGger:USBPd:IMIN....1812
SEARch:TRIGger:USBPd:IOPerator....1812
SEARch:TRIGger:WIDTh:ACOPy....1318
SEARch:TRIGger:WIDTh:DELTa....1328
SEARch:TRIGger:WIDTh:POLarity....1328
SEARch:TRIGger:WIDTh:RANGe....1328
SEARch:TRIGger:WIDTh:WIDTh....1329
SEARch:TRIGger:WIDTh[:STATe]....1317
SEARch:TRIGger:WINDow:ACOPy....1318
SEARch:TRIGger:WINDow:DELTa....1330
SEARch:TRIGger:WINDow:RANGe....1330
SEARch:TRIGger:WINDow:TIMerange....1331
SEARch:TRIGger:WINDow:WIDTh....1331
SEARch:TRIGger:WINDow[:STATe]....1317
SENSe[:ROSCillator]:EXTernal:FREQuency....1126
SENSe[:ROSCillator]:SOURce....1126
SINGle....1070
STATus:OPERation:CONDition?....1948
STATus:OPERation:ENABle....1948
STATus:OPERation[:EVENt]?....1948
STATus:QUEStionable:ADCState:CONDition?....1950
STATus:QUEStionable:ADCState:ENABle....1950
STATus:QUEStionable:ADCState:NTRansition....1951
STATus:QUEStionable:ADCState:PTRansition....1951
STATus:QUEStionable:ADCState[:EVENt]?....1951
STATus:QUEStionable:COVerload:CONDition?....1950
STATus:QUEStionable:COVerload:ENABle....1950
STATus:QUEStionable:COVerload:NTRansition....1951
STATus:QUEStionable:COVerload:PTRansition....1951
STATus:QUEStionable:COVerload[:EVENt]?....1951
STATus:QUEStionable:LAMPlitude:CONDition?....1950
STATus:QUEStionable:LAMPlitude:ENABle....1950
STATus:QUEStionable:LAMPlitude:NTRansition....1951
STATus:QUEStionable:LAMPlitude:PTRansition....1951
STATus:QUEStionable:LAMPlitude[:EVENt]?....1951
STATus:QUEStionable:MASK:CONDition?....1950
STATus:QUEStionable:MASK:ENABle....1951
STATus:QUEStionable:MASK:NTRansition....1951
STATus:QUEStionable:MASK:PTRansition....1952
STATus:QUEStionable:MASK[:EVENt]?....1951
STATus:QUEStionable:TEMPerature:CONDition?....1950
STATus:QUEStionable:TEMPerature:ENABle....1950
STATus:QUEStionable:TEMPerature:NTRansition....1951
STATus:QUEStionable:TEMPerature:PTRansition....1951
STATus:QUEStionable:TEMPerature[:EVENt]?....1951
STATus:QUEStionable:ZVCoverload:CONDition?....1950
STATus:QUEStionable:ZVCoverload:ENABle....1951
STATus:QUEStionable:ZVCoverload:NTRansition....1951
STATus:QUEStionable:ZVCoverload:PTRansition....1952

STATus:QUEStionable:ZVCoverload[:EVENt]?....1951
STOP....1070
SYSTem:COMMunicate:PRINter:ENUMerate:FIRSt?....1378
SYSTem:COMMunicate:PRINter:ENUMerate[:NEXT]?....1379
SYSTem:COMMunicate:PRINter:SELect<1..2>....1379
SYSTem:DATE....1049
SYSTem:DEVice:ID?....1049
SYSTem:DFPRint....1051
SYSTem:DISPlay:MESSage:STATe....1047
SYSTem:DISPlay:MESSage[:TEXT]....1048
SYSTem:DISPlay:UPDate....1047
SYSTem:ERRor:ALL?....1051
SYSTem:ERRor:CODE:ALL?....1052
SYSTem:ERRor:CODE[:NEXT]?....1052
SYSTem:ERRor:COUNt?....1052
SYSTem:ERRor[:NEXT]?....1051
SYSTem:EXIT....1048
SYSTem:KLOCk....1047
SYSTem:PRESet....1049
SYSTem:SHUTdown....1048
SYSTem:TIME?....1049
SYSTem:VERSion?....1051
TIMebase:DIVisions?....1072
TIMebase:HORizontal:POSition....1072
TIMebase:RACTime?....1285
TIMebase:RANGe....1071
TIMebase:REFerence....1072
TIMebase:ROLL:ENABle....1075
TIMebase:ROLL:MTIMe....1076
TIMebase:ROLL:STATe?....1076
TIMebase:SCALe....1071
TRACe:REMote:MODE:FILE:DEXecution:DURation....1955
TRACe:REMote:MODE:FILE:ENABle....1954
TRACe:REMote:MODE:FILE:FILTer....1954
TRACe:REMote:MODE:FILE:FORMat....1953
TRACe:REMote:MODE:FILE:FUNCtions....1956
TRACe:REMote:MODE:FILE:NAME....1953
TRACe:REMote:MODE:FILE:PARSer....1955
TRACe:REMote:MODE:FILE:RPC....1955
TRACe:REMote:MODE:FILE:SIZE....1953
TRACe:REMote:MODE:FILE:STARtmode....1954
TRACe:REMote:MODE:FILE:STOPmode....1954
TRIGger<m>:ANEDge:COUPling....1131
TRIGger<m>:ANEDge:CUToff:HIGHpass....1132
TRIGger<m>:ANEDge:CUToff:LOWPass....1133
TRIGger<m>:ANEDge:FILTer....1132
TRIGger<m>:ANEDge:GND....1133
TRIGger<m>:ANEDge:NREJect....1166
TRIGger<m>:ANEDge:SLOPe....1133
TRIGger<m>:ARINc:DATA:CONDition....1598

TRIGger<m>:ARINc:DATA:MAX....1599
TRIGger<m>:ARINc:DATA:MIN....1598
TRIGger<m>:ARINc:ERRor:CODing....1599
TRIGger<m>:ARINc:ERRor:PARity....1599
TRIGger<m>:ARINc:LABel:CONDition....1599
TRIGger<m>:ARINc:LABel:MAX....1600
TRIGger<m>:ARINc:LABel:MIN....1599
TRIGger<m>:ARINc:MAXGap:BITS....1600
TRIGger<m>:ARINc:MAXGap:SELect....1600
TRIGger<m>:ARINc:MINGap:BITS....1600
TRIGger<m>:ARINc:MINGap:SELect....1600
TRIGger<m>:ARINc:SDI....1601
TRIGger<m>:ARINc:SSM....1601
TRIGger<m>:ARINc:TYPE....1598
TRIGger<m>:CAN:ACKerror....1461
TRIGger<m>:CAN:BITSterror....1462
TRIGger<m>:CAN:BORDer....1459
TRIGger<m>:CAN:CRCerror....1462
TRIGger<m>:CAN:DCONdition....1459
TRIGger<m>:CAN:DLC....1460
TRIGger<m>:CAN:DLCCondition....1460
TRIGger<m>:CAN:DMAX....1459
TRIGger<m>:CAN:DMIN....1459
TRIGger<m>:CAN:FDATa:BRS....1458
TRIGger<m>:CAN:FDATa:DPOPerator....1461
TRIGger<m>:CAN:FDATa:DPOSition....1461
TRIGger<m>:CAN:FDATa:DPTO....1461
TRIGger<m>:CAN:FDATa:ESI....1458
TRIGger<m>:CAN:FDATa:FDF....1458
TRIGger<m>:CAN:FDATa:SCERror....1462
TRIGger<m>:CAN:FDATa:STANdard....1456
TRIGger<m>:CAN:FORMerror....1462
TRIGger<m>:CAN:FTYPe....1456
TRIGger<m>:CAN:ICONdition....1457
TRIGger<m>:CAN:IMAX....1458
TRIGger<m>:CAN:IMIN....1458
TRIGger<m>:CAN:ITYPe....1457
TRIGger<m>:CAN:NDBYtes?....1460
TRIGger<m>:CAN:SYMBolic:DMAX....1487
TRIGger<m>:CAN:SYMBolic:DMIN....1487
TRIGger<m>:CAN:SYMBolic:MSGValue....1487
TRIGger<m>:CAN:SYMBolic:SGEValue....1488
TRIGger<m>:CAN:SYMBolic:SIGValue....1487
TRIGger<m>:CAN:SYMBolic:TSIGnals....1487
TRIGger<m>:CAN:TYPE....1455
TRIGger<m>:CMSB:ADVanced:BIT....1713
TRIGger<m>:CMSB:ADVanced:DMAX....1713
TRIGger<m>:CMSB:ADVanced:DMIN....1714
TRIGger<m>:CMSB:ADVanced:DOPerator....1714
TRIGger<m>:CMSB:ADVanced:ERENable....1714

TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle....1714
TRIGger<m>:CMSB:ADVanced:FIENable....1712
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle....1712
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT....1713
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX....1713
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN....1714
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator....1714
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle....1712
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX....1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN....1715
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator....1715
TRIGger<m>:CMSB:ADVanced:FRENable....1712
TRIGger<m>:CMSB:ADVanced:IMAX....1715
TRIGger<m>:CMSB:ADVanced:IMIN....1715
TRIGger<m>:CMSB:ADVanced:IOPerator....1715
TRIGger<m>:CMSB:ICONdition....1711
TRIGger<m>:CMSB:IMAX....1712
TRIGger<m>:CMSB:IMIN....1711
TRIGger<m>:CMSB:NRZ:WRDLength....1712
TRIGger<m>:CMSB:PATTern....1711
TRIGger<m>:CMSB:TYPE....1710
TRIGger<m>:CXPI:CT....1840
TRIGger<m>:CXPI:DATA:DCONdition....1840
TRIGger<m>:CXPI:DATA:DMAX....1841
TRIGger<m>:CXPI:DATA:DMIN....1841
TRIGger<m>:CXPI:DATA:ICONdition....1841
TRIGger<m>:CXPI:DATA:IMAX....1841
TRIGger<m>:CXPI:DATA:IMIN....1842
TRIGger<m>:CXPI:DEXTension:CONDition....1842
TRIGger<m>:CXPI:DEXTension:MAX....1842
TRIGger<m>:CXPI:DEXTension:MIN....1843
TRIGger<m>:CXPI:DLC:CONDition....1843
TRIGger<m>:CXPI:DLC:MAX....1843
TRIGger<m>:CXPI:DLC:MIN....1843
TRIGger<m>:CXPI:ERRor:CRC....1844
TRIGger<m>:CXPI:ERRor:DLC....1844
TRIGger<m>:CXPI:ERRor:IBS....1844
TRIGger<m>:CXPI:ERRor:IFS....1844
TRIGger<m>:CXPI:ERRor:LENGth....1845
TRIGger<m>:CXPI:ERRor:PARity....1845
TRIGger<m>:CXPI:ERRor:UART....1845
TRIGger<m>:CXPI:FID:CONDition....1845
TRIGger<m>:CXPI:FID:MAX....1846
TRIGger<m>:CXPI:FID:MIN....1846
TRIGger<m>:CXPI:NM....1846
TRIGger<m>:CXPI:TYPE....1846
TRIGger<m>:DATatoclock:CSOurce:EDGE....1146
TRIGger<m>:DATatoclock:CSOurce:LEVel....1147
TRIGger<m>:DATatoclock:CSOurce[:VALue]....1146
TRIGger<m>:DATatoclock:HTIMe....1147

TRIGger<m>:DATatoclock:STIMe....................1148
TRIGger<m>:ECOupling....................1167
TRIGger<m>:EDGE:SLOPe....................1131
TRIGger<m>:ETHernet:ERRor:CRC....................1614
TRIGger<m>:ETHernet:ERRor:LENGth....................1615
TRIGger<m>:ETHernet:ERRor:PREamble....................1615
TRIGger<m>:ETHernet:FRAMe:CCONdition....................1615
TRIGger<m>:ETHernet:FRAMe:CMAX....................1616
TRIGger<m>:ETHernet:FRAMe:CMIN....................1616
TRIGger<m>:ETHernet:FRAMe:DCONdition....................1616
TRIGger<m>:ETHernet:FRAMe:DMAX....................1616
TRIGger<m>:ETHernet:FRAMe:DMIN....................1617
TRIGger<m>:ETHernet:FRAMe:SCONdition....................1617
TRIGger<m>:ETHernet:FRAMe:SMAX....................1617
TRIGger<m>:ETHernet:FRAMe:SMIN....................1617
TRIGger<m>:ETHernet:FRAMe:TCONdition....................1618
TRIGger<m>:ETHernet:FRAMe:TMAX....................1618
TRIGger<m>:ETHernet:FRAMe:TMIN....................1618
TRIGger<m>:ETHernet:PATTern....................1614
TRIGger<m>:ETHernet:TYPE....................1614
TRIGger<m>:EVENt:BEEP....................1171
TRIGger<m>:EVENt:PRINt....................1171
TRIGger<m>:EVENt:RUNexec....................1172
TRIGger<m>:EVENt:WFMSave....................1172
TRIGger<m>:EXTern:OVERload....................1172
TRIGger<m>:FINDlevel....................1130
TRIGger<m>:FLXRay:BSSerror....................1531
TRIGger<m>:FLXRay:CENable....................1528
TRIGger<m>:FLXRay:CMAX....................1529
TRIGger<m>:FLXRay:CMIN....................1529
TRIGger<m>:FLXRay:CSTep....................1529
TRIGger<m>:FLXRay:DCONdition....................1530
TRIGger<m>:FLXRay:DMAX....................1531
TRIGger<m>:FLXRay:DMIN....................1531
TRIGger<m>:FLXRay:DPOPerator....................1530
TRIGger<m>:FLXRay:DPOSition....................1530
TRIGger<m>:FLXRay:DPTO....................1530
TRIGger<m>:FLXRay:FCONdition....................1527
TRIGger<m>:FLXRay:FESerror....................1532
TRIGger<m>:FLXRay:FMAX....................1527
TRIGger<m>:FLXRay:FMIN....................1527
TRIGger<m>:FLXRay:FSSerror....................1532
TRIGger<m>:FLXRay:HCRCerror....................1532
TRIGger<m>:FLXRay:NUFRame....................1526
TRIGger<m>:FLXRay:PCONdition....................1528
TRIGger<m>:FLXRay:PCRCerror....................1532
TRIGger<m>:FLXRay:PLPReamble....................1526
TRIGger<m>:FLXRay:PMAX....................1528
TRIGger<m>:FLXRay:PMIN....................1528
TRIGger<m>:FLXRay:STFRame....................1526

TRIGger<m>:FLXRay:SYFRame....1526
TRIGger<m>:FLXRay:SYMBol....1531
TRIGger<m>:FLXRay:TYPE....1525
TRIGger<m>:FORCe....1169
TRIGger<m>:GLITch:POLarity....1134
TRIGger<m>:GLITch:RANGe....1134
TRIGger<m>:GLITch:WIDTh....1134
TRIGger<m>:HBTO:CRC:CONDition....1642
TRIGger<m>:HBTO:CRC:MAX....1642
TRIGger<m>:HBTO:CRC:MIN....1642
TRIGger<m>:HBTO:DADDress:CONDition....1639
TRIGger<m>:HBTO:DADDress:MAX....1640
TRIGger<m>:HBTO:DADDress:MIN....1639
TRIGger<m>:HBTO:DATA:DCONdition....1643
TRIGger<m>:HBTO:DATA:DMAX....1643
TRIGger<m>:HBTO:DATA:DMIN....1643
TRIGger<m>:HBTO:DATA:ICONdition....1644
TRIGger<m>:HBTO:DATA:IMAX....1644
TRIGger<m>:HBTO:DATA:IMIN....1644
TRIGger<m>:HBTO:ERRor:CRC....1645
TRIGger<m>:HBTO:ERRor:PREamble....1644
TRIGger<m>:HBTO:ERRor:SFD....1645
TRIGger<m>:HBTO:LENGth:CONDition....1641
TRIGger<m>:HBTO:LENGth:MAX....1641
TRIGger<m>:HBTO:LENGth:MIN....1641
TRIGger<m>:HBTO:SADDress:CONDition....1640
TRIGger<m>:HBTO:SADDress:MAX....1641
TRIGger<m>:HBTO:SADDress:MIN....1640
TRIGger<m>:HBTO:TYPE....1638
TRIGger<m>:HOLDoff:AUTotime?....1162
TRIGger<m>:HOLDoff:EVENts....1161
TRIGger<m>:HOLDoff:MAX....1162
TRIGger<m>:HOLDoff:MIN....1161
TRIGger<m>:HOLDoff:MODE....1160
TRIGger<m>:HOLDoff:SCALing....1163
TRIGger<m>:HOLDoff:TIME....1161
TRIGger<m>:I2C:ACCess....1391
TRIGger<m>:I2C:ACONdition....1393
TRIGger<m>:I2C:ADDRess....1393
TRIGger<m>:I2C:ADDTo....1393
TRIGger<m>:I2C:ADNack....1392
TRIGger<m>:I2C:ADOR<n>:ADRType....1394
TRIGger<m>:I2C:ADOR<n>:ENABle....1393
TRIGger<m>:I2C:ADOR<n>:RWBit....1394
TRIGger<m>:I2C:ADOR<n>[:VALue]....1394
TRIGger<m>:I2C:AMODe....1392
TRIGger<m>:I2C:DCONdition....1395
TRIGger<m>:I2C:DMAX....1396
TRIGger<m>:I2C:DMIN....1396
TRIGger<m>:I2C:DPOPerator....1395

TRIGger<m>:I2C:DPOSition....1395
TRIGger<m>:I2C:DPTO....1395
TRIGger<m>:I2C:DRNack....1392
TRIGger<m>:I2C:DWNack....1392
TRIGger<m>:I2C:MODE....1391
TRIGger<m>:I2S:SOWords....1563
TRIGger<m>:I2S:TCONdition<n>:CHANnel....1561
TRIGger<m>:I2S:TCONdition<n>:CONDtion....1561
TRIGger<m>:I2S:TCONdition<n>:DMAX....1562
TRIGger<m>:I2S:TCONdition<n>:DMIN....1562
TRIGger<m>:I2S:TYPE....1560
TRIGger<m>:I2S:WSSLope....1563
TRIGger<m>:INTerval:DELTa....1143
TRIGger<m>:INTerval:RANGe....1142
TRIGger<m>:INTerval:SLOPe....1142
TRIGger<m>:INTerval:WIDTh....1143
TRIGger<m>:LEVel<n>:NOISe:ABSolute....1165
TRIGger<m>:LEVel<n>:NOISe:MODE....1164
TRIGger<m>:LEVel<n>:NOISe:PERDivision....1165
TRIGger<m>:LEVel<n>:NOISe:RELative....1165
TRIGger<m>:LEVel<n>:NOISe[:STATe]....1163
TRIGger<m>:LEVel<n>:RUNT:LOWer....1137
TRIGger<m>:LEVel<n>:RUNT:UPPer....1137
TRIGger<m>:LEVel<n>:SLEW:LOWer....1144
TRIGger<m>:LEVel<n>:SLEW:UPPer....1144
TRIGger<m>:LEVel<n>:WINDow:LOWer....1139
TRIGger<m>:LEVel<n>:WINDow:UPPer....1139
TRIGger<m>:LEVel<n>[:VALue]....1130
TRIGger<m>:LIN:BORDer....1496
TRIGger<m>:LIN:CHKSerror....1498
TRIGger<m>:LIN:CRCDatalen....1499
TRIGger<m>:LIN:DCONdition....1495
TRIGger<m>:LIN:DLECondition....1497
TRIGger<m>:LIN:DLENgth....1497
TRIGger<m>:LIN:DMAX....1496
TRIGger<m>:LIN:DMIN....1496
TRIGger<m>:LIN:ERRPattern....1499
TRIGger<m>:LIN:ICONdition....1495
TRIGger<m>:LIN:IDOR<n>:ENABle....1497
TRIGger<m>:LIN:IDOR<n>[:VALue]....1498
TRIGger<m>:LIN:IMAX....1495
TRIGger<m>:LIN:IMIN....1495
TRIGger<m>:LIN:IPERror....1498
TRIGger<m>:LIN:STANdard....1499
TRIGger<m>:LIN:SYERror....1498
TRIGger<m>:LIN:TYPE....1494
TRIGger<m>:MDIO:DATA....1742
TRIGger<m>:MDIO:FRAMetype....1742
TRIGger<m>:MDIO:PHYS....1742
TRIGger<m>:MDIO:REGI....1742

TRIGger<m>:MDIO:ST....1741
TRIGger<m>:MDIO:TYPE....1741
TRIGger<m>:MILStd:CDST:ICONdition....1576
TRIGger<m>:MILStd:CDST:IMAX....1576
TRIGger<m>:MILStd:CDST:IMIN....1576
TRIGger<m>:MILStd:CDST:RCONdition....1573
TRIGger<m>:MILStd:CDST:RMAX....1573
TRIGger<m>:MILStd:CDST:RMIN....1573
TRIGger<m>:MILStd:CMD:CCONdition....1574
TRIGger<m>:MILStd:CMD:CMAX....1574
TRIGger<m>:MILStd:CMD:CMIN....1574
TRIGger<m>:MILStd:CMD:RCONdition....1573
TRIGger<m>:MILStd:CMD:RMAX....1573
TRIGger<m>:MILStd:CMD:RMIN....1573
TRIGger<m>:MILStd:CMD:SCONdition....1574
TRIGger<m>:MILStd:CMD:SMAX....1575
TRIGger<m>:MILStd:CMD:SMIN....1575
TRIGger<m>:MILStd:CMD:TR....1575
TRIGger<m>:MILStd:DATA:DCONdition....1576
TRIGger<m>:MILStd:DATA:DMAX....1577
TRIGger<m>:MILStd:DATA:DMIN....1577
TRIGger<m>:MILStd:DATA:ICONdition....1577
TRIGger<m>:MILStd:DATA:IMAX....1577
TRIGger<m>:MILStd:DATA:IMIN....1578
TRIGger<m>:MILStd:DATA:RCONdition....1573
TRIGger<m>:MILStd:DATA:RMAX....1573
TRIGger<m>:MILStd:DATA:RMIN....1573
TRIGger<m>:MILStd:ERRor:MANChester....1578
TRIGger<m>:MILStd:ERRor:PARity....1578
TRIGger<m>:MILStd:ERRor:SYNC....1578
TRIGger<m>:MILStd:MAXResponse:BITS....1578
TRIGger<m>:MILStd:MAXResponse:SELect....1579
TRIGger<m>:MILStd:MINGap:BITS....1579
TRIGger<m>:MILStd:MINGap:SELect....1579
TRIGger<m>:MILStd:STATus:BCReceived....1579
TRIGger<m>:MILStd:STATus:BUSY....1580
TRIGger<m>:MILStd:STATus:DBCaccept....1580
TRIGger<m>:MILStd:STATus:INSTrument....1580
TRIGger<m>:MILStd:STATus:MERRor....1580
TRIGger<m>:MILStd:STATus:SREQuest....1580
TRIGger<m>:MILStd:STATus:SUBSystem....1581
TRIGger<m>:MILStd:STATus:TERMinal....1581
TRIGger<m>:MILStd:TPSPecifier....1581
TRIGger<m>:MILStd:TYPE....1572
TRIGger<m>:MODE....1169
TRIGger<m>:OFFSet:LIMited....1072
TRIGger<m>:OUT:DELay....1171
TRIGger<m>:OUT:PLENgth....1170
TRIGger<m>:OUT:POLarity....1170
TRIGger<m>:OUT:STATe....1170

TRIGger<m>:PARallel:DATatoclock:CSOurce:EDGE....1879
TRIGger<m>:PARallel:DATatoclock:CSOurce[:VALue]....1876
TRIGger<m>:PARallel:DATatoclock:HTIMe....1880
TRIGger<m>:PARallel:DATatoclock:STIMe....1879
TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:EDGE:SLOPe....1877
TRIGger<m>:PARallel:PATTern:BIT<0..15>....1881
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:PATTern:MODE....1881
TRIGger<m>:PARallel:PATTern:TIMeout:MODE....1881
TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]....1882
TRIGger<m>:PARallel:PATTern:WIDTh:DELTa....1883
TRIGger<m>:PARallel:PATTern:WIDTh:RANGe....1882
TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]....1882
TRIGger<m>:PARallel:SPATtern:CSOurce:EDGE....1883
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]....1876
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:SPATtern:PATTern....1883
TRIGger<m>:PARallel:STATe:BIT<0..15>....1880
TRIGger<m>:PARallel:STATe:CSOurce:EDGE....1880
TRIGger<m>:PARallel:STATe:CSOurce:VALue....1876
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:TIMeout:RANGe....1878
TRIGger<m>:PARallel:TIMeout:TIME....1879
TRIGger<m>:PARallel:TYPE....1876
TRIGger<m>:PARallel:WIDTh:DELTa....1878
TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]....1876
TRIGger<m>:PARallel:WIDTh:POLarity....1877
TRIGger<m>:PARallel:WIDTh:RANGe....1877
TRIGger<m>:PARallel:WIDTh:WIDTh....1878
TRIGger<m>:PATTern:MODE....1151
TRIGger<m>:PATTern:TIMeout:MODE....1151
TRIGger<m>:PATTern:TIMeout[:TIME]....1152
TRIGger<m>:PATTern:WIDTh:DELTa....1153
TRIGger<m>:PATTern:WIDTh:RANGe....1152
TRIGger<m>:PATTern:WIDTh[:WIDTh]....1152
TRIGger<m>:POWerline:SLOPe....1159
TRIGger<m>:QUALify<n>:A:LOGic....1149
TRIGger<m>:QUALify<n>:A[:ENABle]....1148
TRIGger<m>:QUALify<n>:AB:LOGic....1150
TRIGger<m>:QUALify<n>:ABCD:LOGic....1150
TRIGger<m>:QUALify<n>:B:LOGic....1149
TRIGger<m>:QUALify<n>:B[:ENABle]....1148
TRIGger<m>:QUALify<n>:C:LOGic....1149
TRIGger<m>:QUALify<n>:C[:ENABle]....1149
TRIGger<m>:QUALify<n>:CD:LOGic....1150
TRIGger<m>:QUALify<n>:D:LOGic....1149
TRIGger<m>:QUALify<n>:D[:ENABle]....1149
TRIGger<m>:RFReject....1123

TRIGger<m>:RFSReject....1124
TRIGger<m>:RUNT:DELTa....1138
TRIGger<m>:RUNT:POLarity....1137
TRIGger<m>:RUNT:RANGe....1137
TRIGger<m>:RUNT:WIDTh....1138
TRIGger<m>:SCOupling....1147
TRIGger<m>:SENT:CRCerror....1673
TRIGger<m>:SENT:FORMerror....1672
TRIGger<m>:SENT:IRFLength....1674
TRIGger<m>:SENT:PPERioderror....1673
TRIGger<m>:SENT:PULSeerror....1673
TRIGger<m>:SENT:SDCN....1672
TRIGger<m>:SENT:SDMN....1672
TRIGger<m>:SENT:SDMX....1672
TRIGger<m>:SENT:SICN....1671
TRIGger<m>:SENT:SIDType....1671
TRIGger<m>:SENT:SIMN....1671
TRIGger<m>:SENT:SIMX....1672
TRIGger<m>:SENT:STATus....1669
TRIGger<m>:SENT:STYPe....1670
TRIGger<m>:SENT:TDCN....1670
TRIGger<m>:SENT:TDMN....1670
TRIGger<m>:SENT:TDMX....1670
TRIGger<m>:SENT:TTYPe....1669
TRIGger<m>:SENT:TYPE....1668
TRIGger<m>:SEQuence:COUNt....1167
TRIGger<m>:SEQuence:DELay....1167
TRIGger<m>:SEQuence:MODE....1166
TRIGger<m>:SEQuence:RESet:EVENt....1168
TRIGger<m>:SEQuence:RESet:TIMeout:TIME....1168
TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle]....1168
TRIGger<m>:SLEW:DELTa....1146
TRIGger<m>:SLEW:RANGe....1145
TRIGger<m>:SLEW:RATE....1145
TRIGger<m>:SLEW:SLOPe....1144
TRIGger<m>:SOURce:SBSelect....1387
TRIGger<m>:SOURce[:SELect]....1128
TRIGger<m>:SOURce[:SELect]....1386
TRIGger<m>:SOURce[:SELect]....1874
TRIGger<m>:SPATtern:CSOurce:EDGE....1154
TRIGger<m>:SPATtern:CSOurce:LEVel....1154
TRIGger<m>:SPATtern:CSOurce[:VALue]....1153
TRIGger<m>:SPATtern:PATTern....1154
TRIGger<m>:SPI:DPOPerator....1426
TRIGger<m>:SPI:DPOSition....1427
TRIGger<m>:SPI:DPTO....1427
TRIGger<m>:SPI:FCONdition....1427
TRIGger<m>:SPI:MISopattern....1427
TRIGger<m>:SPI:MODE....1425
TRIGger<m>:SPI:MOSipattern....1427

TRIGger<m>:SPI:PALignment....1426
TRIGger<m>:SWIRe:CTYPe....1822
TRIGger<m>:SWIRe:DATA:CONDition....1822
TRIGger<m>:SWIRe:DATA:MAX....1822
TRIGger<m>:SWIRe:DATA:MIN....1823
TRIGger<m>:SWIRe:ERRor:ESC....1823
TRIGger<m>:SWIRe:ERRor:PARity....1823
TRIGger<m>:SWIRe:TIME:CONDition....1823
TRIGger<m>:SWIRe:TIME:MAX....1824
TRIGger<m>:SWIRe:TIME:MIN....1824
TRIGger<m>:SWIRe:TYPE....1824
TRIGger<m>:TIMeout:RANGe....1141
TRIGger<m>:TIMeout:TIME....1142
TRIGger<m>:TV:CUSTom:LDURation....1158
TRIGger<m>:TV:CUSTom:SCANmode....1158
TRIGger<m>:TV:CUSTom:SDURation....1159
TRIGger<m>:TV:CUSTom:STYPe....1159
TRIGger<m>:TV:LFIeld....1158
TRIGger<m>:TV:LINE....1157
TRIGger<m>:TV:MODE....1156
TRIGger<m>:TV:POLarity....1156
TRIGger<m>:TV:STANdard....1155
TRIGger<m>:TYPE....1129
TRIGger<m>:UART:DATA....1447
TRIGger<m>:UART:DPOPerator....1446
TRIGger<m>:UART:DPOSition....1446
TRIGger<m>:UART:DPTO....1447
TRIGger<m>:UART:FCONdition....1447
TRIGger<m>:UART:SOURce....1446
TRIGger<m>:UART:TYPE....1446
TRIGger<m>:USB:ACONdition....1759
TRIGger<m>:USB:AMAX....1759
TRIGger<m>:USB:AMIN....1759
TRIGger<m>:USB:DATA....1759
TRIGger<m>:USB:DCONdition....1760
TRIGger<m>:USB:DPOPerator....1760
TRIGger<m>:USB:DPOSition....1760
TRIGger<m>:USB:ECONdition....1760
TRIGger<m>:USB:EMAX....1761
TRIGger<m>:USB:EMIN....1761
TRIGger<m>:USB:ERRC....1761
TRIGger<m>:USB:FCONdition....1762
TRIGger<m>:USB:FMAX....1762
TRIGger<m>:USB:FMIN....1762
TRIGger<m>:USB:HAND....1762
TRIGger<m>:USB:PATT....1763
TRIGger<m>:USB:PCONdition....1763
TRIGger<m>:USB:PMAX....1764
TRIGger<m>:USB:PMIN....1763
TRIGger<m>:USB:SCONdition....1764

TRIGger<m>:USB:SMAX....1764
TRIGger<m>:USB:SMIN....1764
TRIGger<m>:USB:SPEC....1764
TRIGger<m>:USB:STCO....1765
TRIGger<m>:USB:TCONdition....1765
TRIGger<m>:USB:TMAX....1766
TRIGger<m>:USB:TMIN....1766
TRIGger<m>:USB:TOKen....1766
TRIGger<m>:USB:TYPE....1757
TRIGger<m>:USB:WADD....1766
TRIGger<m>:USB:WEND....1766
TRIGger<m>:USB:WETCheck....1767
TRIGger<m>:USB:WFRN....1767
TRIGger<m>:USB:WPAY....1767
TRIGger<m>:USB:WPID....1767
TRIGger<m>:USB:WPOR....1767
TRIGger<m>:USB:WSEU....1767
TRIGger<m>:USB:WSTC....1768
TRIGger<m>:USBPd:BIT....1799
TRIGger<m>:USBPd:DMAX....1800
TRIGger<m>:USBPd:DMIN....1800
TRIGger<m>:USBPd:DOPerator....1800
TRIGger<m>:USBPd:ERENable....1798
TRIGger<m>:USBPd:ERRor<n>:ENABle....1798
TRIGger<m>:USBPd:FIENable....1799
TRIGger<m>:USBPd:FRAMe<n>:ENABle....1798
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT....1799
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX....1800
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN....1800
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator....1800
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle....1799
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX....1801
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN....1801
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator....1802
TRIGger<m>:USBPd:FRENable....1798
TRIGger<m>:USBPd:IMAX....1801
TRIGger<m>:USBPd:IMIN....1801
TRIGger<m>:USBPd:IOPerator....1802
TRIGger<m>:WIDTh:DELTa....1136
TRIGger<m>:WIDTh:POLarity....1135
TRIGger<m>:WIDTh:RANGe....1135
TRIGger<m>:WIDTh:WIDTh....1136
TRIGger<m>:WINDow:DELTa....1141
TRIGger<m>:WINDow:RANGe....1139
TRIGger<m>:WINDow:TIME....1140
TRIGger<m>:WINDow:WIDTh....1140
TRPRobe:ID:PARTnumber?....1104
TRPRobe:ID:PRDate?....1104
TRPRobe:ID:SRNumber?....1105
TRPRobe:ID:SWVersion?....1104

TRPRobe:PMETer:RESults:COMMon? 1092
TRPRobe:PMETer:RESults:DIFFerential? 1092
TRPRobe:PMETer:RESults:NEGative? 1093
TRPRobe:PMETer:RESults:POSitive? 1093
TRPRobe:PMETer:RESults:SINGle? 1092
TRPRobe:PMETer:VISibility 1091
TRPRobe:SETup:ACCoupling 1096
TRPRobe:SETup:ADVanced:AUDioverload 1097
TRPRobe:SETup:ADVanced:FILTer 1097
TRPRobe:SETup:ATTenuation:DEFProbe 1101
TRPRobe:SETup:ATTenuation:MANual 1094
TRPRobe:SETup:ATTenuation:MODE 1094
TRPRobe:SETup:ATTenuation:UNIT 1094
TRPRobe:SETup:ATTenuation[:AUTO]? 1088
TRPRobe:SETup:BANDwidth? 1087
TRPRobe:SETup:CAPacitance? 1105
TRPRobe:SETup:CMOFfset 1095
TRPRobe:SETup:DEGauss 1103
TRPRobe:SETup:DISPlaydiff 1091
TRPRobe:SETup:GAIN:MANual 1095
TRPRobe:SETup:IMPedance? 1105
TRPRobe:SETup:NAME? 1087
TRPRobe:SETup:PRMode 1098
TRPRobe:SETup:STATe? 1087
TRPRobe:SETup:TYPE? 1087
TRPRobe:SETup:ZAXV 1096
WAVeform<m>:XYCurve:RATio 1198
WAVeform<m>:XYCurve:STATe 1199
WAVeform<m>:XYCurve:SWAP 1199
WAVeform<m>:XYCurve:XSOurce 1200
WAVeform<m>:XYCurve:YSOurce 1200
WGENerator<m>:ACOPy 1886
WGENerator<m>:ARBGen:COPY 1896
WGENerator<m>:ARBGen:MULTichannel:IMPort 1896
WGENerator<m>:ARBGen:MULTichannel:NAME 1896
WGENerator<m>:ARBGen:MULTichannel:OPEN 1897
WGENerator<m>:ARBGen:NAME 1897
WGENerator<m>:ARBGen:OPEN 1897
WGENerator<m>:ARBGen:RUNMode 1897
WGENerator<m>:ARBGen:SAMPles? 1898
WGENerator<m>:ARBGen:SELect 1898
WGENerator<m>:ARBGen:SRATe 1898
WGENerator<m>:ARBGen[:SOURce] 1898
WGENerator<m>:COUPling:ALL 1903
WGENerator<m>:COUPling:AMPLitude 1904
WGENerator<m>:COUPling:OUTPut 1904
WGENerator<m>:COUPling:PHASeshift 1904
WGENerator<m>:COUPling[:FREQuency] 1904
WGENerator<m>:FREQuency 1887
WGENerator<m>:FUNCtion:PULSe[:WIDTh] 1888

WGENerator<m>:FUNCtion:RAMP[:SYMMetry]....1888
WGENerator<m>:FUNCtion:SQUare:DCYCle....1888
WGENerator<m>:FUNCtion[:SELect]....1887
WGENerator<m>:MODulation:AM:DCYCle....1889
WGENerator<m>:MODulation:AM:DEPTh....1889
WGENerator<m>:MODulation:AM:FREQuency....1890
WGENerator<m>:MODulation:AM:SYMMetry....1890
WGENerator<m>:MODulation:AM[:FUNCtion]....1891
WGENerator<m>:MODulation:CARRier:FREQuency....1890
WGENerator<m>:MODulation:CARRier:PERiod....1890
WGENerator<m>:MODulation:FM:DCYCle....1891
WGENerator<m>:MODulation:FM:DEViation....1891
WGENerator<m>:MODulation:FM:FREQuency....1892
WGENerator<m>:MODulation:FM:SYMMetry....1892
WGENerator<m>:MODulation:FM[:FUNCtion]....1892
WGENerator<m>:MODulation:FSK:FONE....1892
WGENerator<m>:MODulation:FSK:FTWO....1893
WGENerator<m>:MODulation:FSK[:RATE]....1893
WGENerator<m>:MODulation:NDCLevel....1901
WGENerator<m>:MODulation:NLABsolute?....1901
WGENerator<m>:MODulation:NLPCent....1901
WGENerator<m>:MODulation:NOISe....1902
WGENerator<m>:MODulation:PWM:DCYCle....1893
WGENerator<m>:MODulation:PWM:DEPTh....1894
WGENerator<m>:MODulation:PWM:FREQuency....1894
WGENerator<m>:MODulation:PWM:SYMMetry....1894
WGENerator<m>:MODulation:PWM[:FUNCtion]....1894
WGENerator<m>:MODulation:TYPE....1889
WGENerator<m>:OUTPut[:LOAD]....1899
WGENerator<m>:PERiod....1887
WGENerator<m>:PRESet....1886
WGENerator<m>:SOURce....1886
WGENerator<m>:SWEep:FSTart....1895
WGENerator<m>:SWEep:TIME....1895
WGENerator<m>:SWEep[:FEND]....1895
WGENerator<m>:VOLTage:DCLevel....1900
WGENerator<m>:VOLTage:HIGH....1900
WGENerator<m>:VOLTage:INVersion....1900
WGENerator<m>:VOLTage:LOW....1900
WGENerator<m>:VOLTage:OFFSet....1901
WGENerator<m>:VOLTage[:VPP]....1899
WGENerator<m>[:ENABle]....1886
ZVC:BANDwidth....1107
ZVC:RESCoupled....1109
ZVC:TYPE....1108
ZVC:Z<m>:I<n>:BANDwidth?....1108
ZVC:Z<m>:I<n>:DATA:HEADer?....1119
ZVC:Z<m>:I<n>:DATA[:VALues]?....1119
ZVC:Z<m>:I<n>:IMPedance....1108
ZVC:Z<m>:I<n>:OFFSet....1109

ZVC:Z<m>:I<n>:OVERload:RSTO....1109
ZVC:Z<m>:I<n>:OVERload:VALue?....1110
ZVC:Z<m>:I<n>:POSition....1110
ZVC:Z<m>:I<n>:SCALe....1111
ZVC:Z<m>:I<n>:SHUNt:EVALue....1111
ZVC:Z<m>:I<n>:SHUNt:MAXCurrent....1111
ZVC:Z<m>:I<n>:SHUNt:MAXVoltage....1112
ZVC:Z<m>:I<n>:SHUNt:MODE....1111
ZVC:Z<m>:I<n>:SHUNt:MXCValue?....1112
ZVC:Z<m>:I<n>:SKEW....1113
ZVC:Z<m>:I<n>:ZERComp:DETect....1117
ZVC:Z<m>:I<n>:ZERComp:STATe?....1118
ZVC:Z<m>:I<n>:ZERComp:USE....1117
ZVC:Z<m>:I<n>[:STATe]....1113
ZVC:Z<m>:ID:NAME?....1116
ZVC:Z<m>:ID:PARTnumber?....1116
ZVC:Z<m>:ID:SRNumber?....1117
ZVC:Z<m>:ID:SWVersion?....1117
ZVC:Z<m>:V<n>:BANDwidth?....1113
ZVC:Z<m>:V<n>:DATA:HEADer?....1119
ZVC:Z<m>:V<n>:DATA[:VALues]?....1119
ZVC:Z<m>:V<n>:IMPedance....1114
ZVC:Z<m>:V<n>:OFFSet....1114
ZVC:Z<m>:V<n>:OVERload:RSTO....1114
ZVC:Z<m>:V<n>:OVERload:VALue?....1115
ZVC:Z<m>:V<n>:POSition....1115
ZVC:Z<m>:V<n>:SCALe....1115
ZVC:Z<m>:V<n>:SKEW....1116
ZVC:Z<m>:V<n>[:STATe]....1116

Index

Symbols

*OPC ... 1981
*OPC? ... 1981
*RST ... 1982
*WAI ... 1981
9.91E37
    Remote control ... 1979
100BASE-T1
    Configuration settings ... 677
    Configuring ... 680
    Trigger settings ... 680
    Triggering ... 686

A

AC coupling
    Trying out ... 39
Acquisition
    Decimation ... 143
    Fast segmentation ... 145
    Modes ... 132
    Settings ... 143
    Single, multiple ... 32
    Start ... 32
    Stop ... 32
    Time ... 133
    Trying out ... 38
ACQUISITION
    Key ... 29
Acquisition modes ... 132, 143
Acquisition time ... 133
Actions
    Mask test ... 393
Active probe
    Differential ... 137
    Micro button ... 159
    Offset compensation ... 173
    Overview ... 136
    ProbeMeter ... 160
    Setup ... 163
Active waveform ... 75
Adapter for Tektronix probes ... 178
ADC overflow ... 1992
ADC sample rate ... 130
ADC samples ... 130
Aligning
    Input channels ... 122
Amplitude measurement ... 311
Amplitude/time measurements
    Settings ... 315
Analog
    Parallel bus display ... 870
Application, external ... 120
Area measurements ... 314
Arithmetic ... 144
Arranging waveforms ... 76
Asynchronous commands ... 1980
Audio
    Configuring ... 605
    Track ... 612
    Trend ... 617
    Trigger settings ... 605
Auto clear
    Search results ... 425
Auto trigger mode ... 32
Auto-hide signal bar ... 78, 119
Auto-logon ... 991
Auto, trigger mode ... 229
Autocorrelation ... 267
Automatic measurements ... 49
    Configuration settings ... 297
    Display settings ... 301
    Gate settings ... 303
Autonaming ... 459
AUTOSET
    Key ... 27
    Trying out ... 36
AutoZero ... 157
Average ... 144
    FFT ... 371
    Mathematics ... 259
Average count ... 144, 230, 260
Average count (N-single count) ... 144, 230, 260

B

Bandwidth ... 149
    Instrument ... 130
    Probe ... 130
    Spectrum measurement ... 324
Bench top operation ... 16
Beta state ... 28
Biased correlation ... 267
Big endian
    CAN ... 532
    LIN ... 567
Bit pattern editor ... 481
Bit string
    serial pattern, MSO ... 888
Blackman Harris window (FFT) ... 369
Blind time ... 145
Block data ... 1976
Boolean parameters ... 1976
Brackets ... 1977
Brochure ... 12
Bug report ... 1960
Burst width ... 312
Buses
    Node labels ... 478

C

Cable, USB ... 24
CAN
    Configuration settings ... 523
    Data length ... 532
    Data pattern ... 532
    Endianness ... 532
    Frame types ... 530
    ID type ... 530
    Identifier ... 531
    Sample point ... 527
    Transfer order ... 532
    Trigger settings ... 528
    Trigger types ... 529

Case-sensitivity
SCPI ... 1973
Center frequency
Spectrum measurement ... 327
CH × keys ... 30
Channel
Bandwidth ... 326
Center frequency ... 327
CH × keys ... 30
Connector ... 23
Couple vertical settings ... 150
Offset ... 31, 149
Settings ... 147
Waveforms ... 74
Channel power
Spectrum measurement ... 324
Cleaning ... 1959
Clear all ... 114
Clear status
Remote ... 1040
Clipping ... 1992
Clock
Parallel bus ... 871
Clock period
SENT configuration ... 702
Clock tolerance
SENT configuration ... 702
Colon ... 1977
Color
Waveform ... 105
Waveforms ... 115
Color tables ... 108
Editing ... 115
Comb
Parallel bus display ... 870
Comma ... 1977
Command sequence
recommendation ... 1982
Remote ... 1044
Commands ... 1969
Asynchronous ... 1980
Brackets ... 1977
Colon ... 1977
Comma ... 1977
Command line structure ... 1978
Double dagger ... 1977
Finding a command ... 1017
GBIP, addressed ... 1971
GBIP, universal ... 1970
instrument control ... 1969
Overlapping ... 1980
Question mark ... 1977
Quotation mark ... 1977
SCPI compliant ... 1969
Sequential ... 1980
Syntax elements ... 1977
Vertical stroke ... 1977
White space ... 1977
Common commands
Syntax ... 1972
Compensation, passive probes ... 196
Computer name
Changing ... 998
CONDition ... 1983
Configuration
hardware ... 1961
SENT ... 700
Connectors
Probe compensation ... 24
Rear panel ... 25
USB ... 24
Coupled zoom ... 251
quick access ... 82, 242
Coupling ... 128
Channel ... 149
Trigger ... 238
CRC calculation
SENT configuration ... 702
CRC version
SENT configuration ... 702
Cross-correlation ... 267
CSV
Waveform export ... 443
CSV export ... 456
Cursor
Saving data ... 457
Settings for data export ... 445
Used for gating ... 303
CURSOR key ... 32
Cursor measurements
Configuring ... 286
Disabling ... 286
Enable ... 288
Envelope waveform ... 291
Peak excursion ... 293, 327
Peak search ... 292
Performing ... 286
Results ... 284
Settings ... 288
Cursors
Configuring display ... 288
Configuring measurement ... 286
Coupling ... 291
Display ... 291, 292
Displaying ... 286
Labels ... 291
Minimizing results ... 48
Position ... 290
Quick access ... 83
Style ... 291
Track waveform ... 286, 290
Trying out ... 48
Type ... 289
Custom:
Filter settings ... 760
Custom: Manchester / NRZ
Configuration settings ... 735
Configuring ... 754
Decoding ... 762
Trigger settings ... 755
Triggering ... 760
Customer support ... 1959
CXPI
Configuration settings ... 847
Configuring ... 850
Trigger settings ... 851
Triggering ... 858
Cycle area measurement ... 314
Cycle measurements ... 311

D

Data
SENT configuration ... 701

Data entry ... 89
Data length
  LIN ... 567
Data Length Code
  CAN ... 532
Data nibbles
  SENT configuration ... 702, 728
Data pattern
  CAN ... 532
  LIN ... 567
Data security ... 1960
Data sheet ... 12
Data2Clock
  Search ... 417
  Trigger ... 216
  Trigger (MSO) ... 883
DCL ... 1970
Decimation ... 38, 132, 143
Decode
  Custom protocol, decode layer ... 754
  CXPI, decode layer ... 849
  Ethernet, decode layer ... 660, 679
  MDIO, decode layer ... 772
  SENT ... 722
  SpaceWire, decode layer ... 835
  USBPD, decode layer ... 819
Decode layer
  Custom protocol ... 754
  CXPI ... 849
  Ethernet ... 660, 679
  MDIO ... 772
  SpaceWire ... 835
  USBPD ... 819
Decode table
  MSO ... 877
Decoded bus
  Parallel bus display ... 870
DEF ... 1975
Default
  File names ... 461
  Path for saving ... 461
Default values
  Remote ... 1043
Delay ... 312
Delay measurement ... 317
Delete ... 84
Device footprint ... 1960
Device-specific commands ... 1969
DHCP ... 997
DHCP server
  LAN configuration ... 998
Diagrams ... 70
  Settings ... 109
Dialog boxes ... 72
  Background transparency ... 29
  Font size ... 118
  Transparency ... 118
  Usage ... 87
Differential probe ... 137
Digital channels ... 867
  Configuration settings ... 868
  Configuring ... 875
  Decode table ... 877
  Export ... 890
  Search ... 892
  Triggering ... 889
Digital resolution ... 874
Digital signals ... 867
Digital waveforms ... 74
Display
  Color tables ... 108, 115
  Diagram settings ... 109
  Intensity ... 106
  Measurement information ... 300
  Overview ... 69
  Persistence ... 105
  Settings overview ... 104
  Signal bar ... 78, 118
  Waveform settings ... 105
  XY-diagram ... 280, 282
  Zoom ... 242
DISPLAY
  Key ... 28
Display elements
  Diagram ... 69
  Input box ... 72
  Menu bar ... 72
  Result box ... 72
  Signal bar ... 69
  Toolbar ... 72
DNS server
  LAN configuration ... 998
Double dagger ... 1977
DOWN ... 1975
Duty cycle
  Eye ... 322
Duty cycle measurement ... 312
DVI-D ... 21, 26
DVI-D connector ... 26

E

Edge
  Search conditions ... 411
  Trigger ... 206
Edge count ... 315
Edge trigger
  MSO ... 880
Electric power (math. function) ... 269
EMI suppression ... 19
ENABle ... 1983
Enable registers
  Remote ... 1042
Endianness
  CAN ... 532
  LIN ... 567
ENTER key ... 34
Envelope ... 38, 144
  FFT ... 371
  Mathematics ... 259
  Measurements ... 316
Envelope waveform
  Cursor measurement ... 291
Error queue ... 1986
Error queues
  recommendations ... 1982
Error report ... 1960
ESC key ... 34
ESE (event status enable register) ... 1987
ESR ... 1985
ESR (event status register) ... 1987
Ethernet
  Configuration settings ... 657
  Configuring ... 660

Trigger settings 661
Triggering 666
EVENt 1983
Event actions
Mask test 393
Event status enable register (ESE) 1987
Remote 1041
Event status register (ESR) 1987
Remote 1041
Export
Data 445
Digital channels 890
History 457
Include horizontal values 449
Integer format 449
Parallel buses 890
Raw values 449
Waveform histograms 450
XML, CSV, BIN 439
Expressions 263
EXT TRIG IN 23
External application 120
External monitor 21, 26
External trigger input 235
External trigger out 231
External trigger output 26
Extinction ratio
Eye 322
Eye
Amplitude 322
Base 322
Bit rate 322
Fall time 322
Height 322
Measurement results 322
Rise time 322
Top 322
Width 322
Eye measurements
Characteristics 321
Results 322
Settings 324

F

Fail condition, mask test 386
Fail criteria, mask test 383
Fall time 312
Fast segmentation 145
FFT 84, 359
Configuring 363
Fundamentals 359
Gating 372
measurement speed 362
MSO 891
Peak excursion 293, 327
Performance considerations 364
Search 406
Setup 366, 370
Trying out 55
Window types 369
FILE
Key 438
File formats
Waveform export 439
File import
Restrictions 439
FILE key 28
File names
Default 459, 461
Filter
Custom 760
Find
Level 85
Remote control command 91
Fingertip zoom 45
FIR (math. function) 269
FIR filter 272
Firmware
shutdown (remote) 1048
Firmware update 94
Flattop2 window (FFT) 369
FlexRay
Configuration settings 579
Trigger settings 583
Triggering 588
Font size
Dialog and result boxes 118
Formula editor
Reference 263
Using 271
Formulas
Advanced expressions 263
Basic editor 261
Frame length
SENT configuration 703
Frame types
CAN 530
Free Run, trigger mode 229
Frontend
information 1961
Frontpanel
information 1961
Full autoset 475
Function generator option 27

G

Gain, vertical 128
Gate
Coupled to cursor 303
Coupled to zoom 303
Saving data 457
Search 422
Search, defining 424
Settings 303
Settings for data export 445
Gate area
Configuring 302
Coupling to zoom 303, 373, 423
Gating
FFT 372
Gaussian window (FFT) 369
GBIP connector 27
GET 1971
Getting started 11
Glitch
Search conditions 412
Trigger 206
GPIB
Address 1013
interface messages 1970
Remote control interface 1009

GPIB bus control
  Remote ... 1042
Grid ... 70
Grounding ... 18
GTL ... 1971

H

Hamming window (FFT) ... 369
Hann window (FFT) ... 369
Hardware
  information ... 1961
Hardware check ... 122
Hardware zoom ... 82, 242
Header files ... 439
Help ... 11
  Open ... 91
  Search for topic ... 92
  Using ... 92
HELP
  Key ... 28
High definition ... 151
High definition mode ... 28
High res ... 143
High signal level measurement ... 311
HiSLIP
  Protocol ... 1012
Histogram
  Saving ... 438
Histogram measurements
  Settings ... 336
Histograms ... 330
  Area ... 72
  averaging ... 310
  Characteristics ... 330
  Configuring ... 333
  Diagram ... 72
  Exporting waveform histograms ... 450
  Horizontal ... 330
  Jitter measurement ... 333
  Markers ... 336
  Measurement histograms ... 348
  Measurement results ... 330
  Measurements ... 53
  Quick access ... 83
  Saving waveform histograms ... 450
  Setup ... 334
  Toolbar icons ... 333
  Trying out ... 53
  Vertical ... 330
History ... 273
  Export ... 445, 457
  Fast segmentation ... 145
  MSO ... 876
  time stamp ... 276
  Trying out ... 46
HISTORY key ... 34
Hold time ... 318
Holdoff ... 226
  TV trigger ... 221
  Using ... 202
Horizontal
  Controls ... 29
  Label ... 70
  Position ... 29
  Reference point ... 29
  Time base ... 138
HORIZONTAL
  Key ... 29
Horizontal position ... 134
Hysteresis
  Noise rejection, search ... 429
  trigger ... 227

I

I²C
  Basics ... 482
  Configuration settings ... 484
  Configuring ... 486, 583
  Decode results ... 493
  Trigger settings ... 487
  Triggering ... 492, 520, 609
Icons
  Measurement status ... 299
ID type
  CAN ... 530
Identification
  Remote ... 1041
Identifier
  CAN ... 531
  LIN ... 566
IFC ... 1970
Import waveforms
  Restrictions ... 439
INF ... 1975
Infinite persistence ... 106
Input ... 23
Input box ... 72
Instrument messages ... 1969
Instrument settings
  Loading ... 432
  Recall ... 1043
  Save ... 1043
  Saving ... 437
Intensity ... 106
INTENSITY key ... 29
Intensity of display elements ... 29
Interface messages ... 1970, 1971
Interfaces
  GPIB ... 1012
  LAN ... 1011
Interlaced scanning ... 225
Interleaved X/Y ... 449
Interrupt ... 1994
Interval
  Search conditions ... 416
  Trigger ... 213
Invalid results
  Remote control ... 1979
IP address ... 1011
  Changing ... 997
  Remote Desktop ... 1008
IST ... 1985
IST flag ... 1987
  Remote ... 1041

J

Jitter
  Measurement via histogram ... 333

K

Kaiser Bessel window (FFT) 369
Keep X-grid fixed 111
Keep Y-grid fixed 111
Keyboard 20
    Connecting 19
    On-screen 89
    Usage 68
Keypad 89
Keywords
    see Mnemonics 1971

L

Label list
    SENT 711
Label lists 478
Labels
    Waveform 113
    Waveforms 112
LAN 1001
    Configuration 996, 1001
    Connector 26
    Environment 996
    Interface 1011
    IP address 1011
    LAN configuration 1004
    Ping 1005
    Remote control interface 1009
    Resource string 1011
    VXI protocol 1012
Language
    Changing 98
Level measurements 311
LEVELS knob 32
License key
    Options 126
Limit checks
    Actions 355
    Configuring 353
    Enable 354
    Performing 353
    Valid range 354
    Violation settings 355
Limits
    Definition 354
LIN
    Configuration settings 563
    Data length 567
    Data pattern 567
    Endianness 567
    Identifier 566
    Transfer order 567
    Trigger settings 565
    Trigger types 566
    Triggering 569
Little endian
    CAN 532
    LIN 567
LLO 1970
Load instrument settings 1043
Loading
    User settings 432
Lock touchscreen 28
Logic analyzer 867
LOGIC key 33
Logic probe connector 26
Logical operations on channels 218
Logical Thresholds
    MSO, configuring 876
    MSO, settings 872
Login 991
Long-term measurements
    Basics 346
    Configuring 346
    Horizontal scale 351
    Number of points 351
    Period of time per point 352
    Saving 454
    Scale mode 352
    Settings 348
Low signal level measurement 311

M

Mainboard
    information 1961
Malfunctions
    reacting 1982
Margins
    Definition 354
Marker limit
    Histograms 337
Markers
    Example 43
    Histograms 330, 336
Mask definition
    Settings 387
    Type 386
Mask setup 387
Mask test 383
    Display settings 396
    Display, configuring 402
    Event actions, configuring 401
    Event actions, settings 393
    Fail condition 386
    Fail criteria 383
    Fail criteria, configuring 401
    History, testing 404
    Performing 403
    Quick access 83
    Results 383
    Setting up 401
    Starting 403
    State, mask test result 403
    Test settings 385
    Trying out 60
    Violation tolerance 386
Mask test results
    Acq. completed 384
    Acq. remaining 384
    Acquisition hits 384
    Fail rate 384
    Sample hits 384
    State 384
    Test result 385
Mask testing 383
Masks
    Creating user mask 397
    Definition 383, 387
    Load settings 387
    Loading 404
    modifying user mask 398

Save settings ... 387
Saving ... 404
Segments, point settings ... 389
Segments, settings ... 388
User mask definition ... 387
Waveform mask definition ... 390
MASKS key ... 33
MATH key ... 31
Math waveforms ... 31, 74
Arithmetic ... 259
Basic editor ... 261
Displaying ... 257
FFT ... 359
FFT Gating ... 372
FFT setup ... 366, 370
Formula editor ... 263
Operators ... 261, 263
Saving ... 257
Scaling ... 259
Setup ... 258
Trying out ... 57
MAX ... 1975
Max value measurement ... 311
MDIO
Configuration settings ... 770
Configuring ... 772
Trigger settings ... 773
Triggering ... 776
MDIO protocol
Basics ... 769
Mean measurement ... 311
MEAS key ... 33, 295
Measurement histograms ... 348
Measurement type ... 298
Measurement types
Area ... 314
Counting ... 315
Eye ... 322
Histograms ... 330
Protocol ... 337
Spectrum ... 324
Time ... 312
Measurements
Amplitude/time ... 310
Configuring ... 296
Delay ... 317
Histograms ... 53
Icons in results table ... 299
Pulse train count ... 319
Quick measurement ... 84
Results ... 298
Setup/Hold ... 318
Starting ... 295
Statistics ... 343
Median
Histograms ... 330
Memory ... 273
Messages
commands ... 1969
instrument ... 1969
instrument responses ... 1970
Interface ... 1970
Micro button ... 137, 159
MIN ... 1975
Min value measurement ... 311
Minimized waveform ... 75
Mixed signal option ... 867
Mixed Signal Option ... 26
Mnemonics ... 1971
Optional ... 1973
MODE
Key (setup) ... 28
Key (trigger) ... 32
Trying out ... 36
Monitor ... 21, 26
Mouse ... 20
Connecting ... 19
Usage ... 68
MSO ... 26, 867
FFT ... 891
History ... 876
Resolution ... 874
Time qualification ... 885
Trigger settings ... 879
Zoom ... 876
Multi channel export
Enable ... 446
Remote control ... 1084, 1362
Results in header file ... 442
Multiple measurements ... 351

N

N dB down
Spectrum ... 326
NAN ... 1975
NAN (not a number)
Remote control ... 1979
Navigation controls
Data input ... 90
Keys ... 35
Overview ... 34
Network
Environment ... 996
NINF ... 1975
Noise
Eye ... 322
hysteresis settings ... 227
Noise reject
Search ... 428
Normal trigger mode ... 32
Normal, trigger mode ... 229
NTRansition ... 1983
Numeric data entry ... 89
Numeric parameters ... 1974
Numeric values
Special ... 1975

O

Occupied bandwidth
Spectrum measurement ... 324, 327
Offset
Active probe ... 173
Channel ... 31, 149
Trying out ... 39
On/Off key ... 27
One File ... 434
Open source acknowledgment ... 12
Operating system ... 991
Access ... 995
service packs ... 993
Settings ... 995

Operation
Concepts ... 67
Manual ... 68
Operation complete
Remote ... 1041
Operators
Basic editor, mathematics ... 261
Formula editor mathematics ... 263
Options
Beta option ... 127
Identification (remote) ... 1042
Installing ... 126
K4, FlexRay ... 579
K5, I²S, Audio ... 598
Key ... 126
R&S RTE-B1 (MSO) ... 26
R&S RTE-B1E (for R&S RT-ZVC) ... 26
R&S RTE-B6 (waveform and function generator) ... 27
R&S RTE-B10 (GBIP) ... 27
Output buffer ... 1985
Overlapping commands ... 1980
Preventing ... 1981
Overshoot measurement ... 311

P

Packing ... 1961
Parallel buses ... 867
Clock setup ... 871
Configuration settings ... 868
Configuring ... 875
Decode table ... 877
Export ... 890
Signal selection ... 872
Triggering ... 889
Parallel poll register enable
Remote ... 1042
Parameters
Block data ... 1976
Boolean ... 1976
Numeric values ... 1974
SCPI ... 1974
Special numeric values ... 1975
String ... 1976
Text ... 1976
Passive probe
Overview ... 136
Settings ... 162
Password ... 991
Remote Desktop ... 1008
Pattern
Qualification, state trigger ... 218
Pattern generator
Setup ... 912
Pattern search condition ... 218
Pattern trigger ... 218
MSO ... 885
Pause pulse
SENT configuration ... 703
Peak detect ... 38, 143
Peak excursion
Cursor measurement ... 293, 327
Search ... 293, 327
Spectrum measurement ... 293, 327
Peak search
Cursor measurement ... 285, 286
Spectrum measurement ... 285, 292
Peak to peak measurement ... 311
Performance
Considerations for FFT ... 364
FFT parameters ... 362
Information ... 114
Persistence
Settings ... 105
Ping ... 1005
Position
Horizontal ... 29, 134
Vertical ... 31, 128
Position mode ... 114
Power
Connector ... 25
Off ... 18
On ... 18
POWER
Key ... 27
PPC ... 1971
PPE ... 1985
PPE register ... 1987
PPU ... 1970
Preset ... 469
PRESET key ... 28
PRINT key ... 28
Printer ... 20
Color mode ... 466
Configuring ... 466
Output ... 466
Paper format ... 466
Printing
Screenshots ... 466
Settings ... 462
Trying out ... 63
Probability markers
Histograms ... 336
ProbeMeter ... 137
Measurement ... 311
Setting ... 160
Probes
Active ... 136
AutoZero ... 157
Compensation ... 24
Differential ... 137
Passive ... 136
passive, compensation ... 196
Setup ... 155
Tektronix ... 178
Progressive scanning ... 225
Projector ... 21
Protocol
SENT ... 697
VXI ... 1012
PROTOCOL key ... 33
Protocols
100BASE-T1 configuration settings ... 677
100BASE-T1 trigger settings ... 680
Audio trigger settings ... 605
CAN configuration settings ... 523
Custom: filter settings ... 760
Custom: Manchester / NRZ configuration settings ... 735
Custom: Manchester / NRZ trigger settings ... 755
CXPI configuration settings ... 847
CXPI trigger settings ... 851
Data format ... 475
Display settings ... 475
Ethernet configuration settings ... 657

Ethernet trigger settings 661
FlexRay configuration settings 579
FlexRay trigger settings 583
I²C configuration settings 484
I²C decode results 493
I²C trigger settings 487
Label lists 478
LIN configuration settings 563
LIN trigger settings 565
MDIO configuration settings 770
MDIO trigger settings 773
SENT 696
SENT configuration 700
SENT decode results 722
SENT label list 711
SENT protocol 697
SENT search 726
SENT trigger 704
SpaceWire configuration settings 833
SPI configuration settings 502
SPI decode results 508
SPI trigger settings 505
UART decode results 520
USB 2.0 configuration settings 789
USB 2.0 trigger settings 793
USBPD configuration settings 817
USBPD trigger settings 820
PTRansition 1983
Pulse count 315
Pulse train count 319

Q

Q-factor
Eye 322
Queries 1969, 1979
Status 1995
Question mark 1977, 1979
Quick measurements
Toolbar 84
Trying out 51
Quotation mark 1977

R

R&S RT-Z2T 178
R&S RT-ZVC
multi-channel power probe 179
Rackmounting 17
Recall instrument settings 1043
Recall intermediate 1043
Recommendations
remote control programming 1982
Record length 130, 133
Rotary knob 29
Rectangular window (FFT) 369
Redo
Toolbar 82
REDO
Key 34
REF IN/OUT 26
REF key 30
Reference levels
Basics 305
Configuring 306
Defining automatically 306
Defining manually 306
displaying 301
Displaying 306
Level settings 307
Settings 307
Reference point 29, 71, 134
Reference signal 26
Reference waveforms 30, 74
Displaying 252
Loading 252
Saving 252
Registers 1985
Release notes 12
Remote commands
Finding a command 1017
Remote control 67
Find command using help 91
GPIB address 1013
Interfaces 1009
Protocols 1009
Starting 1015
Remote Desktop 67, 1007
Remote operation 67, 1006, 1007
RES REC LEN key 29
Reset
Histogram, long-term meas., statistics 350
Reset values
Remote 1043
Resolution 128, 130, 133
16 bit 151
Rotary knob 29
Resolution enhancement 132
Resource string
LAN 1011
Restoring
Settings 470
Result box
Default position 112
Result boxes 72
Background transparency 29
Displaying 85
Font size 118
Transparency 118
Result lines
Displaying 301
Results
Clear 114, 301
Configuring display 85
Lines in display 306
Mask test 383
Measurements 298
Reset long-term measurement 350
Saving 452
Rise time 312
RMS
Acquisition setting 143
RMS measurement 311
Roll mode 142
Time for automatic activation 142
Rotary knobs
Trying out 39
RS232 513
Basics 513
RTE-B10 27
RTxServiceReporter 1960
Run 32
RUN CONT key 32
RUN N× SINGLE key 32

Runt
Search conditions ... 413
Trigger ... 209

S

Safety instructions ... 11
Sample point (CAN) ... 527
Sample rate ... 130
Samples ... 130
History ... 273
Save instrument settings ... 1043
Save intermediate ... 1043
Save/Recall
Remote ... 1353
User settings ... 432, 434
Waveforms ... 445
Saveset ... 432, 437
Saving
File name generation ... 459
History ... 445, 457
Instrument settings ... 432, 437
Long-term measurements ... 454
Preset ... 469
Results ... 452
Screenshots ... 466
Trying out ... 63, 65
Waveform histograms ... 450
Waveform segments ... 445, 457
Waveforms ... 445, 457
Scale
Trying out ... 39
Vertical ... 128, 148
SCALE
Horizontal, rotary knob ... 30
Vertical, rotary knob ... 31
Scaling
Horizontal ... 139
Long-term measurements ... 350
Long-term measurements, horizontal ... 351
Math waveforms ... 259
Measurement histograms ... 350
Scanning system (TV signals) ... 225
SCPI
Finding a command ... 1017
Parameters ... 1974
Syntax ... 1972
version ... 1010
SCPI-compliant commands ... 1969
Screen resolution ... 21
Screenshot
Web interface ... 1006
Screenshots ... 63, 462
Meta information ... 462
Print and save ... 461
SDC ... 1971
Search
Auto clear ... 425
Clearing results ... 426
Conditions ... 406
Control ... 406
Copy trigger settings ... 410
Data2Clock ... 417
Definition ... 406
Edge ... 411
Enable ... 409
FFT ... 406
Gate ... 422
Glitch ... 412
Interval ... 416
Navigating results ... 408
Noise rejection ... 428
On digital signals ... 892
Peak excursion ... 293, 327
Quick access ... 83
Result box ... 407
Results display ... 424
Runt ... 413
Scope ... 406
SENT ... 726
SENT error condition ... 731
SENT search criteria ... 726
SENT settings ... 726
SENT short serial message ... 729
SENT transmission sequence ... 727
Show results ... 425
Slew rate ... 417
Sorting results ... 425
Source ... 410
Timeout ... 415
Trigger level ... 411, 412, 413, 416, 418
Trigger search ... 411
Trying out ... 58
Width ... 412
Window ... 414
Zoom window ... 407, 426
Zoom window, configuring ... 427
SEARCH
Key ... 33
Search conditions
Pattern ... 218
Search zoom window ... 407
Searching
In help ... 92
Secured environment ... 1960
Select ... 84
Waveform ... 75
Selected waveform ... 75
Self-alignment ... 122, 1040
Self-test
Remote ... 1044
Selftest ... 122
SENT
About ... 696
Configuration ... 700
Configuring ... 703
Decode results ... 722
Label list ... 711
Protocol ... 697
Search ... 726
Search criteria ... 726
Search error condition ... 731
Search settings ... 726
Search short serial message ... 729
Search transmission sequence ... 727
Trigger error condition ... 710
Trigger serial message setup ... 708
Trigger settings ... 704
Trigger transmission sequence ... 706
Trigger types ... 705
Triggering ... 710
SENT configuration
Clock period ... 702
Clock tolerance ... 702

CRC calculation ... 702
CRC version ... 702
Data ... 701
Data nibbles ... 702, 728
Frame length ... 703
Pause pulse ... 703
Serial protocol ... 702
Threshold ... 701
Sequence
Search ... 729
Trigger ... 708
Sequential commands ... 1980
Serial message
Search sequence ... 729
Sequence ... 708, 729
Trigger sequence ... 708
Serial pattern trigger
MSO ... 888
Serial Pattern trigger ... 219
Serial protocol
SENT configuration ... 702
Service manual ... 11
Service packs ... 993
Service request (SRQ) ... 1986, 1994
Service request enable register (SRE) ... 1986
Remote ... 1043
Setting commands ... 1969
Settling time ... 312
Setup
Controls ... 27
SETUP
Key ... 28
Trying out ... 36
Setup and hold
see Data2Clock ... 216
Setup time ... 318
Setup/Hold measurement ... 318
Setup/Hold ratio ... 318
Show labels ... 113
Shut down ... 18
Shutdown
firmware (remote) ... 1048
Signal bar ... 70
Adjusting ... 78, 118
Auto-hide ... 78, 119
Colors ... 79, 119
Hide and show ... 78, 118
Signal colors
Editing ... 115
Signal icons ... 70
Signal label ... 70
Signal levels
Configuring ... 306
SIGNAL OFF key ... 31
Signal view ... 70
Signals
Color settings ... 107
Skew ... 194
Slew rate
Trigger ... 214
Slew Rate
Search conditions ... 417
SLOPE
Key ... 32
Trying out ... 36
SmartGrid ... 76
SOURCE key ... 32
SpaceWire
Basics ... 832
Configuration settings ... 833
Configuring ... 835
SPD ... 1970
SPE ... 1970
Spectrogram ... 85
Spectrum measurements
Channel BW ... 326
Channel CF ... 327
Cursor measurement ... 285, 286
N dB down ... 326
Occupied bandwidth ... 327
Peak detection ... 286
Peak excursion ... 293, 327
Peak search ... 285
Results ... 324
Settings ... 326
Trying out ... 55
SPI
Configuration settings ... 502
Configuring ... 505
Decode results ... 508
Trigger settings ... 505
Triggering ... 508
SPI protocol
basics ... 501
SRE ... 1985
SRE (service request enable register) ... 1986
SRQ (service request) ... 1986, 1994
Standard
TV trigger ... 222
Standard deviation
Histograms ... 330
Standard deviation measurement ... 311
Start up ... 18
State trigger ... 217
MSO ... 884
Statistics
Enable ... 348
Measurement results ... 343
Status
Measurements ... 299
Queries ... 1995
Status byte
Remote ... 1040, 1043
Status byte (STB) ... 1986
Status registers
CONDition ... 1983
ENABle ... 1983
EVENt ... 1983
model ... 1983
NTRansition ... 1983
Overview ... 1984
parts ... 1983
PTRansition ... 1983
STATus:OPERation ... 1988
Status reporting system
Common commands ... 1040
Status reports ... 1983
application ... 1994
STB ... 1985
Storing ... 1961
String in remote commands ... 1976
Style (waveforms) ... 106
Suffixes ... 1973
Support ... 1960

Switch off
Waveform 76
Switch on
Waveform 75
Switching of transitions 315
Symbolic names, see label lists 478
Syntax elements
SCPI 1977

T

T-SCREEN LOCK key 28
Technical support 1960
Temperature
Changes 122
Termination 128
Text entry 89
Text paramters in remote commands 1976
Threshold
SENT configuration 701
Thresholds
MSO, configuring 876
MSO, settings 872
Time base 30, 133, 138
Time measurements 312
Time qualification
MSO 885
Time scale 30, 133, 139
Time stamp 276
Timebase 139
Timeout
Search conditions 415
Trigger 212
Timeout trigger
MSO 882
Toolbar 79
Coupled zoom 82, 242
Cursor 83
Delete 84
FFT 84
Find level 85
Hardware zoom 82, 242
Hide/show icons 80
Histogram 83
Histogram icons 333
Load saveset 82
Masks 83
Measurement 83, 295
Overview 80
Quick measurement 84
Redo 82
Search 83
Select 84
Show signal bar 82
Show tooltips 82
Undo 82
Zoom 82, 242
Tooltips 82
Show 91
Total harmonic distortion
Spectrum measurement 324
Touchscreen
Adjusting 100
Compared with mouse 68
Control elements 72
Lock, unlock 28
Usage 68
Track 612
Enable 348
Track waveform
Cursor measurement 286, 290
Transfer order
CAN 532
LIN 567
Transition switching count 315
Transition trigger
see Slew rate trigger 214
Transparency 29
Dialog and result boxes 118
Trend 348, 617
Trigger
100BASE-T1 settings 680
Audio settings 605
CAN settings 528
Controls 31
Custom: Manchester / NRZ settings 755
CXPI settings 851
Data2Clock 216
Data2Clock (MSO) 883
Delay (Holdoff) 226
Edge 206
Edge (MSO) 880
Ethernet settings 661
Event (definition) 200
Event (remote) 1044
External input 23, 235
External output 26
FlexRay settings 583
Force 231
Glitch 206
Holdoff 226
I²C settings 487
Information 201
Interval 213
Label 70
Level 32, 71
Level, search 411, 412, 413, 416, 418
LIN settings 565
MDIO settings 773
Mode 32, 36, 229
MSO settings 879
Pattern 218
Pattern (MSO) 885
Position 71
Run 231
Runt 209
Search 411
SENT 704
SENT error condition 710
SENT serial message setup 708
SENT transmission sequence 706
SENT trigger type 705
Sequence (definition) 200
Serial Pattern 219
Serial pattern (MSO) 888
Settings, copy to search 410
Setup/hold 216
Slew rate 214
Slope 32, 36
Source 32
Sources, MSO 879
SPI settings 505
State 201, 217
State (MSO) 884

Timeout 212
Timeout (MSO) 882
TV (video) 221
USB 2.0 settings 793
USBPD settings 820
Width 208
Width (MSO) 881
Window 210
TRIGGER
Key 31
Trigger offset 134
Trigger out
Pulse on limit violation 356
Pulse on mask test 395
Settings 231
Trigger sources
MSO 879
Trigger types
CAN 529
Data2Clock 216
Edge 206
Glitch 206
Interval 213
LIN 566
Pattern 218
Runt 209
Slew rate 214
Timeout 212
Width 208
Window 210
Trigger Types 205
Serial Pattern 219
State 217
TV/Video 221
TV trigger 221
Type
Trigger 205

U

UART 513
Basics 513
Configuring 517
Decode results 520
Ultra segmentation, see Fast segmentation 145
Unbiased correlation 267
Undersampling 130
Undo
Toolbar 82
UNDO
Key 34
Unlock touchscreen 28
UP 1975
USB
Cable 24
Configuring 792
Connecting 19
Connector 24
USB 2.0
Configuration settings 789
Trigger settings 793
Triggering 802
USB flash drive 20
USB protocol
Basics 784
USBPD
Configuration settings 817
Configuring 820
Trigger settings 820
Triggering 823
User ID
Remote Desktop 1008
User manual 11
User mask
Settings 387
User name 991
User settings
Saving 432

V

Vertical
Controls 30
Position 128
Position / Offset 31
Resolution 128
Scale 128, 148
Vertical stroke 1977
Vertical system 128
Video (TV) trigger 221
Violation
Limit checks 355
Violation of limits
Actions 355
Violation tolerance, mask test 386
Virus protection 993
VISA 1010
Resource string 1011
VNC 67, 1006
VXI protocol 1012

W

Wait
Remote 1044
Waveform
Arithmetic 132
Changing color 115
Color 105
Data query 1084
Loading 438
Saving 438
Waveform arithmetic 144
Waveform count
Histograms 330
Waveform export
CSV 443
Data query 1084
Files and formats 439
Header files 439
Remote control 1362
Value files 443
XML 443
Waveform generator
AM modulation 901
Arbitrary 906
Coupling 916
DC offset alignment 921
FM modulation 903
FSK modulation 904
Function generator 895
Modulation 899

Noise 911
Output 910
PWM modulation 902
Settings 893
Sweep 905
Waveform generator option 27
Waveform mask
Settings 390
Waveform samples 130
Waveform Value files 443
Waveforms
Arrange 76
Channel 74
Clear 114
Clipping 1992
Display intensity 29
Labels 112
Math 74
Minimize 76
Names 112
Overview and usage 74
Persistence 105
Reference 74
Saving 445, 457
Select 75
States 75
Style 106
Switch off 76
Switch on 75
XY 74
Zoom 74
Zooming 240
Web browser 1003
Web control 1006
Web interface 67, 1001
Browser 1003
LAN configuration 1004
White space 1977
Width
Search conditions 412
Trigger 208
Width trigger
MSO 881
Window
Search conditions 414
Trigger 210
Windows 991
Access 20, 995
Settings 995

X

XML
Waveform export 443
XY-diagram
Displaying 282
settings 280
XY-waveforms 74

Z

Zoom 240
Area 71
Area, coupling 303, 373, 423
Coupled 251
Diagram 71
Fingertip 45
Methods 240
MSO 876
Multiple 249
On the touchscreen 246
Position/Range 249
Procedures 246
Quick access 82, 242
Saving data 457
Search results 426
Settings 242, 244
Settings for data export 445
Standard 44
Start-stop values 249
Start/Stop settings 243
Trying out 43
Used for gating 303
Waveforms 74
ZOOM
Key 33