Application Note

BRT_AN_020

FT90X Revision C User Manual

Version 1.1

Issue Date: 29-08-2023

This document provides details about the peripherals of the FT90X Revision C as well as the general system registers

Use of Bridgetek devices in life support and/or safety applications is entirely at the user's risk, and the user agrees to defend, indemnify and hold Bridgetek harmless from any and all damages, claims, suits or expense resulting from such use.

Bridgetek Pte Ltd (BRTChip)
1 Tai Seng Avenue, Tower A #03-05, Singapore 536464
Tel: +65 6547 4827
Web Site: http://www.brtchip.com
Copyright © Bridgetek Pte Ltd

Table of Contents

1 Introduction 17
2 FT90X Revision C System Architecture 18
2.1 Architecture Overview 18
2.2 Memory Organization 19
2.3 FT90X Boot Control 20
2.4 Debugging Support 20
3 Register Map 21
4 Notations 23
4.1 Bit Type Notation 23
4.2 Default Bit Value Notation 23
4.3 X Default Value 23
4.4 Bit Range Symbol in Bit Column 23
5 General System Registers 25
5.1 Register Summary 25
5.2 Register Details 27
5.2.1 CHIPID - Chip ID Register (address offset: 0x00) 27
5.2.2 CHIPCFG - Chip Configuration Register (address offset: 0x04) 27
5.2.3 CLKCFG - Clock Configuration Register (address offset: 0x08) 28
5.2.4 PMCFG - Power Management Register (address offset: 0x0C) 29
5.2.5 PTSTNSET - Test & Set Register (address offset: 0x10) 32
5.2.6 PTSTNSETR - Test & Set Shadow Register (address offset: 0x14) 32
5.2.7 MSC0CFG - Miscellaneous Configuration Register (address offset: 0x18) 32
5.2.8 GPIO Function Registers (address offset: 0x1C – 0x5F) 34
5.2.9 GPIO Configuration Registers (address offset: 0x60 – 0x83) 39
5.2.10 GPIO Value Registers (address offset: 0x84 – 0x8F) 42
5.2.11 GPIO Interrupt Enable Registers (address offset: 0x90 – 0x9B) 43
5.2.12 Interrupt Pending Registers (address offset: 0x9C – 0xA7) 43
5.2.13 ETH_PHY_CFG - Ethernet PHY Miscellaneous Configuration Register (address offset: 0xA8) 44
5.2.14 ETH_PHY_ID - Ethernet PHY ID Register (address offset: 0xAC) 45

Copyright © Bridgetek Pte Ltd

5.2.15 DAC_ADC_CONF - ADC/DAC Configuration/Status Register (address offset: 0xB0) ... 45
5.2.16 DAC_ADC_CNT - ADC/DAC Count Register (address offset: 0xB4) ... 47
5.2.17 DAC_ADC_DATA - ADC/DAC Data Register (address offset: 0xB8) ... 48
5.2.18 RESET_STATUS – Global Reset/Status Register (address offset: 0xBC) ... 48

6 Interrupt Controller ... 49

6.1 Register Summary ... 50

6.2 Register Details ... 50

6.2.1 Interrupt 00-03 Priority Assignment Register (address offset: 0x00) ... 50
6.2.2 Interrupt 04-07 Priority Assignment Register (address offset: 0x04) ... 50
6.2.3 Interrupt 08-11 Priority Assignment Register (address offset: 0x08) ... 51
6.2.4 Interrupt 12-15 Priority Assignment Register (address offset: 0x0C) ... 51
6.2.5 Interrupt 16-19 Priority Assignment Register (address offset: 0x10) ... 51
6.2.6 Interrupt 20-23 Priority Assignment Register (address offset: 0x14) ... 51
6.2.7 Interrupt 24-27 Priority Assignment Register (address offset: 0x18) ... 52
6.2.8 Interrupt 28-31 Priority Assignment Register (address offset: 0x1C) ... 52
6.2.9 Interrupt Control Register (address offset: 0x20) ... 52

7 USB Host ... 53

7.1 Register Summary ... 53

7.2 EHCI Operational Registers ... 54

7.2.1 HC Capability Register (address offset: 0x00) ... 54
7.2.2 HCSPARAMS – HC Structural Parameters (address offset: 0x04) ... 54
7.2.3 HCCPARAMS – HC Capability Parameters (address offset: 0x08) ... 54
7.2.4 USBCMD – HC USB Command Register (address offset: 0x10) ... 54
7.2.5 USBSTS – HC USB Status Register (address offset: 0x14) ... 56
7.2.6 USBINTR – HC USB Interrupt Enable Register (address offset: 0x18) ... 56
7.2.7 FRINDEX – HC Frame Index Register (address offset: 0x1C) ... 57
7.2.8 PERIODICLISTBASE – HC Periodic Frame List Base Address Register (address offset: 0x24) ... 57
7.2.9 ASYNCLISTADDR – HC Current Asynchronous List Address Register (address offset: 0x28) ... 58
7.2.10 PORTSC – HC Port Status and Control Register (address offset: 0x30) ... 58

7.3 Configuration Registers ... 59

7.3.1 EOF Time & Asynchronous Schedule Sleep Timer Register (address offset: 0x34) ... 59
7.3.2 Bus Monitor Control / Status Register (address offset: 0x40) ... 60

Copyright © Bridgetek Pte Ltd

7.3.3 HPROT – Master Protection Information Setting Register (address offset: 0x78) 61

7.4 USB Host Testing Registers 61

7.4.1 Vendor Specific IO Control Register (address offset: 0x54) 61

7.4.2 Vendor Specific Status Register (address offset: 0x58) 61

7.4.3 Test Register (address offset: 0x50)..... 62

8 USB Device..... 63

8.1 Register Summary.....63

8.2 Initialization Registers.....65

8.2.1 DC_ADDRESS_ENABLE – Address Register (address offset: 0x18) 65

8.2.2 DC_MODE – Mode Register (address offset: 0x10) 66

8.2.3 DC_INT_ENABLE – Interrupt Enable Register (address offset: 0x08) 66

8.2.4 DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register (address offset: 0x0C) 66

8.3 Control Endpoint Data Flow Registers 66

8.3.1 DC_EP0_CONTROL – Endpoint 0 Control Register (address offset: 0x1C) 67

8.3.2 DC_EP0_STATUS – Endpoint 0 Status Register (address offset: 0x20)..... 67

8.3.3 DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register (address offset: 0x24) 67

8.3.4 DC_EP0_BUFFER – Endpoint 0 Buffer Register (address offset: 0x28)..... 68

8.4 Other Endpoint Data Flow Registers.....68

8.4.1 DC_EP(x)_CONTROL – Endpoint Control Registers (address offset: 0x2C/0x3C/0x4C/0x5C/0x6C/0x7C/0x8C) 68

8.4.2 DC_EP(x)_STATUS – Endpoint Status Registers (address offset: 0x30/0x40/0x50/0x60/0x70/0x80/0x90) 69

8.4.3 DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers (address offset: 0x34/0x44/0x54/0x64/0x74/0x84/0x94) 70

8.4.4 DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers (address offset: 0x35/0x45/0x55/0x65/0x75/0x85/0x95) 70

8.4.5 DC_EP(x)_BUFFER – Endpoint Buffer Registers (address offset: 0x38/0x48/0x58/0x68/0x78/0x88/0x98) 70

8.5 General Registers.....71

8.5.1 DC_INT_STATUS – Interrupt Status Register (address offset: 0x00) 71

8.5.2 DC_EP_INT_STATUS – Endpoints Interrupt Status Register (address offset: 0x04)..... 71

8.5.3 DC_FRAME_NUMBER_LSB – Frame Number LSB Register (address offset: 0x14) 71

8.5.4 DC_FRAME_NUMBER_MSB – Frame Number MSB Register (address offset: 0x15)..... 71

8.6 High-bandwidth Isochronous IN Support & Registers.....71

8.6.1 HBWCTRL0 – High-bandwidth Isochronous Control Register 0 (address offset: 0x00) 72

Copyright © Bridgetek Pte Ltd

8.6.2 HBWCTRL1 – High-bandwidth Isochronous Control Register 1 (address offset: 0x04) . 72
8.6.3 HBWCTRL2 – High-bandwidth Isochronous Control Register 2 (address offset: 0x08) . 73
8.6.4 HBWCTRL3 – High-bandwidth Isochronous Control Register 3 (address offset: 0x0C) . 73
8.6.5 HBW_INT_ENABLE – High-bandwidth Interrupt Enable Register (address offset: 0x10) 74
8.6.6 HBW_INT_STATUS – High-bandwidth Interrupt Status Register (address offset: 0x14)74
8.6.7 Behavior of the HAVE_SPACE Bit..... 74
8.6.8 High-bandwidth Isochronous Pipe in UVC 74
8.6.9 Programming Guide for Using HBW Isochronous Support 76

9 Ethernet 77

9.1 Register Summary.....77

9.2 Register Details.....79

9.2.1 ETH_INT_STATUS – Interrupt Status Register (address offset: 0x0)..... 79
9.2.2 ETH_INT_ENABLE – Interrupt Enable Register (address offset: 0x1)..... 79
9.2.3 ETH_RX_CNTL – Receive Control Register (address offset: 0x02)..... 80
9.2.4 ETH_TX_CNTL – Transmit Control Register (address offset: 0x03) 80
9.2.5 ETH_DATA_N0 – Data Register (octet n) (address offset: 0x04) 80
9.2.6 ETH_DATA_N1 – Data Register (octet n+1) (address offset: 0x05) 81
9.2.7 ETH_DATA_N2 – Data Register (octet n+2) (address offset: 0x06) 81
9.2.8 ETH_DATA_N3 – Data Register (octet n+3) (address offset: 0x07) 81
9.2.9 ETH_ADDR_1 – Address Register (octet 1) (address offset: 0x08) 81
9.2.10 ETH_ADDR_2 – Address Register (octet 2) (address offset: 0x09)..... 81
9.2.11 ETH_ADDR_3 – Address Register (octet 3) (address offset: 0x0A)..... 81
9.2.12 ETH_ADDR_4 – Address Register (octet 4) (address offset: 0x0B)..... 81
9.2.13 ETH_ADDR_5 – Address Register (octet 5) (address offset: 0x0C)..... 81
9.2.14 ETH_ADDR_6 – Address Register (octet 6) (address offset: 0x0D) 81
9.2.15 ETH_THRESHOLD – Threshold Register (address offset: 0x0E)..... 82
9.2.16 ETH_MNG_CNTL – Management Control Register (address offset: 0x0F) 82
9.2.17 ETH_MNG_DIV – Management Divider Register (address offset: 0x10) 82
9.2.18 ETH_MNG_ADDR – Management Address Register (address offset: 0x11) 82
9.2.19 ETH_MNG_TX0 – Management Transmit Data 0 Register (address offset: 0x12) 83
9.2.20 ETH_MNG_TX1 – Management Transmit Data 1 Register (address offset: 0x13) 83
9.2.21 ETH_MNG_RX0 – Management Receive Data 0 Register (address offset: 0x14)..... 83
9.2.22 ETH_MNG_RX1 – Management Receive Data 1 Register (address offset: 0x15)..... 83
9.2.23 ETH_NUM_PKT – Number of Packets Register (address offset: 0x16)..... 83

Copyright © Bridgetek Pte Ltd

9.2.24 ETH_TR_REQ – Transmission Request Register (address offset: 0x17) 83

10 CAN Bus Controller 84

10.1 Register Summary 85

10.2 Register Details 86

10.2.1 CAN_MODE – Mode Register (address offset: 0x00) 86

10.2.2 CAN_CMD – Command Register (address offset: 0x01) 87

10.2.3 CAN_STATUS – Status Register (address offset: 0x02) 87

10.2.4 CAN_INT_STATUS – Interrupt Status Register (address offset: 0x03) 88

10.2.5 CAN_INT_ENABLE – Interrupt Enable Register (address offset: 0x04) 89

10.2.6 CAN_RX_MSG – Receive Message Register (address offset: 0x05) 89

10.2.7 CAN_BUS_TIM_0 – Bus Timing 0 Register (address offset: 0x06) 89

10.2.8 CAN_BUS_TIM_1 – Bus Timing 1 Register (address offset: 0x07) 90

10.2.9 CAN_TX_BUF – Transmit Buffer Register 90

10.2.10 CAN_RX_BUF – Receive Buffer Register 91

10.2.11 CAN Acceptance Filter 92

10.2.12 CAN_ERR_CODE – Error Code Capture Register (address offset: 0x18) 95

10.2.13 CAN_RX_ERR_CNTR – Receive Error Counter Register (address offset: 0x19) 96

10.2.14 CAN_TX_ERR_CNTR – Transmit Error Counter Register (address offset: 0x1A) 96

10.2.15 CAN_ARB_LOST_CODE – Arbitration Lost Code Capture Register (address offset: 0x1B) 96

11 SD Host 97

11.1 Register Summary 97

11.2 Register Details 99

11.2.1 SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register (address offset: 0x00) 99

11.2.2 SDH_BLK_SIZE – Block Size Register (address offset: 0x04) 100

11.2.3 SDH_BLK_COUNT – Block Count Register (address offset: 0x06) 100

11.2.4 SDH_ARG_1 – Argument 1 Register (address offset: 0x08) 100

11.2.5 SDH_TNSFER_MODE – Transfer Mode Register (address offset: 0x0C) 100

11.2.6 SDH_CMD – Command Register (address offset: 0x0E) 101

11.2.7 SDH_RESPONSE – Response Register (address offset: 0x10-0x1C) 102

11.2.8 SDH_BUF_DATA – Buffer Data Port Register (address offset: 0x20) 102

11.2.9 SDH_PRESENT_STATE – Present State Register (address offset: 0x24) 102

11.2.10 SDH_HST_CNTL_1 – Host Control 1 Register (address offset: 0x28) 104

Copyright © Bridgetek Pte Ltd

11.2.11 SDH_PWR_CNTL – Power Control Register (address offset: 0x29) 104
11.2.12 SDH_BLK_GAP_CNTL – Block Gap Control Register (address offset: 0x2A) 105
11.2.13 SDH_CLK_CNTL – Clock Control Register (address offset: 0x2C) 106
11.2.14 SDH_TIMEOUT_CNTL – Timeout Control Register (address offset: 0x2E) 106
11.2.15 SDH_SW_RST – Software Reset Register (address offset: 0x2F) 106
11.2.16 SDH_NRML_INT_STATUS – Normal Interrupt Status Register (address offset: 0x30) 107
11.2.17 SDH_ERR_INT_STATUS – Error Interrupt Status Register (address offset: 0x32) ..107
11.2.18 SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register (address offset: 0x34) 108
11.2.19 SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register (address offset: 0x36) 108
11.2.20 SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register (address offset: 0x38) 109
11.2.21 SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register (address offset: 0x3A) 109
11.2.22 SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register (address offset: 0x3C) 110
11.2.23 SDH_HOST_CNTL_2 – Host Control 2 Register (address offset: 0x3E).....110
11.2.24 SDH_CAP_1 – Capabilities Register 1 (address offset: 0x40).....110
11.2.25 SDH_CAP_2 – Capabilities Register 2 (address offset: 0x44).....112
11.2.26 SDH_RSRV_1 – Reserved 1 Register (address offset: 0x48) 112
11.2.27 SDH_RSRV_2 – Reserved 2 Register (address offset: 0x4C) 112
11.2.28 SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status (address offset: 0x50) 112
11.2.29 SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register (address offset: 0x52 113
11.2.30 SDH_RSRV_3 – Reserved 3 Register (address offset: 0x54) 113
11.2.31 SDH_RSRV_4 – Reserved 4 Register (address offset: 0x58) 113
11.2.32 SDH_PRST_INIT – Preset value for initialization (address offset: 0x60).....114
11.2.33 SDH_PRST_DFLT_SPD – Preset value for default speed (address offset: 0x62) 114
11.2.34 SDH_PRST_HIGH_SPD – Preset value for the high speed (address offset: 0x64) ...114
11.2.35 SDH_PRST_SDR12 – Preset value for SDR12 (address offset: 0x66).....115
11.2.36 SDH_PRST_SDR25 – Preset value for SDR25 (address offset: 0x68).....115
11.2.37 SDH_PRST_SDR50 – Preset value for SDR50 (address offset: 0x6A).....116
11.2.38 SDH_PRST_SDR104 – Preset value for SDR104 (address offset: 0x6C) 116
11.2.39 SDH_PRST_DDR50 – Preset value for DDR50 (address offset: 0x6E) 116
11.2.40 SDH_RSRV_5 – Reserved 5 Register (address offset: 0xFC) 117

Copyright © Bridgetek Pte Ltd

11.2.41 SDH_HC_VER – Host Controller Version Register (address offset: 0xFE).....117
11.2.42 SDH_VNDR_0 – Vendor-defined 0 Register (address offset: 0x100).....117
11.2.43 SDH_VNDR_1 – Vendor-defined 1 Register (address offset: 0x104).....119
11.2.44 SDH_VNDR_2 – Vendor-defined 2 Register (address offset: 0x108).....120
11.2.45 SDH_VNDR_3 – Vendor-defined 3 Register (address offset: 0x10C).....121
11.2.46 SDH_VNDR_4 – Vendor-defined 4 Register (address offset: 0x110).....121
11.2.47 SDH_VNDR_5 – Vendor-defined 5 Register (address offset: 0x114).....122
11.2.48 SDH_VNDR_6 – Vendor-defined 6 Register (address offset: 0x118).....122
11.2.49 SDH_VNDR_7 – Vendor-defined 7 Register (address offset: 0x11C).....122
11.2.50 SDH_VNDR_8 – Vendor-defined 8 Register (address offset: 0x120).....122
11.2.51 SDH_VNDR_9 – Vendor-defined 9 Register (address offset: 0x124).....122
11.2.52 SDH_RSRV_6 – Reserved 6 Register (address offset: 0x128).....122
11.2.53 SDH_HW_ATTR – Hardware Attributes Register (address offset: 0x178).....122
11.2.54 SDH_CPR_MOD_CNTL – Cipher Mode Control Register (address offset: 0x180).....123
11.2.55 SDH_CPR_MOD_STATUS – Cipher Mode Status Register (address offset: 0x184) ..123
11.2.56 SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register (address offset: 0x188).....124
11.2.57 SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register (address offset: 0x18A) 124
11.2.58 SDH_IN_DATA_LSB – Input Data LSB Register (address offset: 0x18C) 124
11.2.59 SDH_IN_DATA_MSB – Input Data MSB Register (address offset: 0x190).....124
11.2.60 SDH_IN_KEY_LSB – Input Key LSB Register (address offset: 0x194).....124
11.2.61 SDH_IN_KEY_MSB – Input Key MSB Register (address offset: 0x198) 124
11.2.62 SDH_OUT_DATA_LSB – Output Data LSB Register (address offset: 0x19C) 124
11.2.63 SDH_OUT_DATA_MSB – Output Data MSB Register (address offset: 0x1A0).....125
11.2.64 SDH_SCRT_CONS_DATA – Secret Constant Table Data Port (address offset: 0x1A4) 125

12 UART 126

12.1 Register Summary.....127

12.2 UART Mode Selection 129

12.3 FIFOSEL, CLKSEL and INTSEL Configuration 130

12.4 Baud Rate Generation 131

12.5 Standard 16550 Compatible Registers 131

12.5.1 UART_RBR – Receiver Buffer Register (address offset: 0x00, requires UART_LCR[7] = 0) 131

Copyright © Bridgetek Pte Ltd

12.5.2 UART_THR – Transmitter Holding Register (address offset: 0x00, requires UART_LCR[7] = 0) 132

12.5.3 UART_DLL – Divisor Latch LSB Register (address offset: 0x00, requires UART_LCR[7] = 1) 132

12.5.4 UART_DLM – Divisor Latch MSB Register (address offset: 0x01, requires UART_LCR[7] = 1) 132

12.5.5 UART_IER – Interrupt Enable Register (address offset: 0x01, requires UART_LCR[7] = 0 and UART_ACR[7] = 0) 132

12.5.6 UART_ISR – Interrupt Status Register (address offset: 0x02, requires that the last value written to UART_LCR was not 0xBF) 132

12.5.7 UART_FCR – FIFO Control Register (address offset: 0x02, requires that the last value written to UART_LCR was not 0xBF) 133

12.5.8 UART_LCR – Line Control Register (address offset: 0x03, reading requires UART_ACR[7] = 0) 135

12.5.9 UART_MCR – Modem Control Register (address offset: 0x04, requires that the last value written to UART_LCR was not 0xBF, reading requires UART_ACR[7] = 0) 136

12.5.10 UART_LSR – Line Status Register (address offset: 0x05, requires that the last value written to UART_LCR was not 0xBF, reading requires UART_ACR[6] = 0) 137

12.5.11 UART_MSR – Modem Status Register (address offset: 0x06, requires that the last value written to UART_LCR was not 0xBF) 139

12.5.12 UART_SPR – SPR Register (address offset: 0x07, requires that the last value written to UART_LCR was not 0xBF) 139

12.6 16650 COMPATIBLE REGISTERS 140

12.6.1 UART_EFR – Enhanced Feature Register (address offset: 0x02) 140

12.6.2 UART_XON1 – XON1 Register (address offset: 0x04) 140

12.6.3 UART_XON2 – XON2 Register (address offset: 0x05) 140

12.6.4 UART_XOFF1 – XOFF1 Register (address offset: 0x06) 140

12.6.5 UART_XOFF2 – XOFF2 Register (address offset: 0x07) 141

12.7 16950 COMPATIBLE REGISTERS 141

12.7.1 UART_ASR – Additional Status Register (address offset: 0x01, requires UART_LCR[7] = 0) 141

12.7.2 UART_RFL – Receiver FIFO Level Register (address offset: 0x03) 141

12.7.3 UART_TFL – Transmitter FIFO Level Register (address offset: 0x04, requires that the last value written to UART_LCR was not 0xBF) 141

12.7.4 UART_ICR – ICR Register (address offset: 0x05, requires that the last value written to UART_LCR was not 0xBF, reading requires UART_ACR[6] = 1) 142

12.8 INDEXED CONTROL REGISTERS 142

12.8.1 UART_ACR – Additional Control Register (SPR offset: 0x00) 144

12.8.2 UART_CPR – Clock Pre-scaler Register (SPR offset: 0x01) 145

Copyright © Bridgetek Pte Ltd

12.8.3 UART_TCR – Times Clock Register (SPR offset: 0x02) 145
12.8.4 UART_CKS – Clock Select Register (SPR offset: 0x03) 145
12.8.5 UART_TTL – Transmitter Trigger Level Register (SPR offset: 0x04) 146
12.8.6 UART_RTL – Receiver Trigger Level Register (SPR offset: 0x05) 146
12.8.7 UART_FCL – Flow Control Level LSB Register (SPR offset: 0x06) 146
12.8.8 UART_FCH – Flow Control Level Register MSB (SPR offset: 0x07) 146
12.8.9 UART_ID1 – Identification 1 Register (SPR offset: 0x08) 146
12.8.10 UART_ID2 – Identification 2 Register (SPR offset: 0x09) 147
12.8.11 UART_ID3 – Identification 3 Register (SPR offset: 0x0A) 147
12.8.12 UART_REV – Revision Register (SPR offset: 0x0B) 147
12.8.13 UART_CSR – Channel Software Reset Register (SPR offset: 0x0C) 147
12.8.14 UART_NMR – Nine Bit Mode Register (SPR offset: 0x0D) 147
12.8.15 UART_MDM – Modem Disable Mask Register (SPR offset: 0x0E) 148
12.8.16 UART_RFC – Readable FCR Register (SPR offset: 0x0F) 148
12.8.17 UART_GDS – Good Data Status Register (SPR offset: 0x10) 148
12.8.18 UART_RSRV_1 – Reserved Register 1 (SPR offset: 0x11) 149
12.8.19 UART_PIDX – Port Index Register (SPR offset: 0x12) 149
12.8.20 UART_RSRV_2 – Reserved Register 2 (SPR offset: 0x13) 149

13 Timers and Watchdog 150

13.1 Register Summary 151

13.2 Register Details 152

13.2.1 TIMER_CONTROL_0 – Timers Control Register 0 (address offset: 0x00) 152
13.2.2 TIMER_CONTROL_1 – Timers Control Register 1 (address offset: 0x01) 152
13.2.3 TIMER_CONTROL_2 – Timers Control Register 2 (address offset: 0x02) 152
13.2.4 TIMER_CONTROL_3 – Timers Control Register 3 (address offset: 0x03) 153
13.2.5 TIMER_CONTROL_4 – Timers Control Register 4 (address offset: 0x04) 153
13.2.6 TIMER_INT – Timers Interrupt Register (address offset: 0x05) 153
13.2.7 TIMER_SELECT – Timers A..D Select Register (address offset: 0x06) 153
13.2.8 TIMER_WDG – Watchdog Start Value (address offset: 0x07) 154
13.2.9 TIMER_WRITE_LSB – Timer A..D Start Value 7:0 (address offset: 0x08) 154
13.2.10 TIMER_WRITE_MSB – Timer A..D Start Value 15:8 (address offset: 0x09) 154
13.2.11 TIMER_PRESC_LSB – Pre-scaler Start Value 7:0 (address offset: 0x0A) 154
13.2.12 TIMER_PRESC_MSB – Pre-scaler Start Value 15:8 (address offset: 0x0B) 154
13.2.13 TIMER_READ_LSB – Timer A..D Current Value 7:0 (address offset: 0x0C) 154
13.2.14 TIMER_READ_MSB – Timer A..D Current Value 15:8 (address offset: 0x0D) 155

Copyright © Bridgetek Pte Ltd

13.2.15 TIMER_RESERVED – Reserved Register (address offset: 0x0E).....155
13.2.16 WDT_STATUS – Watchdog Status Register (address offset: 0x0F).....155
14 I²S..... 156
14.1 Register Summary.....158
14.2 Register Details.....158
14.2.1 I2SCR – Configuration Register 1 (address offset: 0x00).....158
14.2.2 I2SCR2 – Configuration Register 2 (address offset: 0x02).....159
14.2.3 I2SIRQEN – Interrupt Enable Register (address offset: 0x04).....160
14.2.4 I2SIRQPEND – Interrupt Pending Register (address offset: 0x06).....160
14.2.5 I2SRWDATA – Transmit / Receive Data Register (address offset: 0x08).....160
14.2.6 I2SRXCOUNT – RX Count Register (address offset: 0x0C).....161
14.2.7 I2STXCOUNT – TX Count Register (address offset: 0x0E).....161
15 SPI Master 162
15.1 Register Summary.....162
15.2 Register/FIFO Access Modes and FIFO Size.....163
15.3 Block Diagram.....164
15.4 SPI Master Interrupts 165
15.4.1 Receiver FIFO Trigger Level Interrupt 165
15.4.2 Transmitter FIFO Trigger Level Interrupt.....165
15.4.3 Receiver FIFO Timeout Interrupt 165
15.4.4 Transfer Error Interrupt 165
15.4.5 Transmitter FIFO Empty Interrupt 166
15.4.6 FIFO Polling Mode 166
15.5 Register Details.....166
15.5.1 SPIM_CNTL – Control Register (address offset: 0x00).....166
15.5.2 SPIM_STATUS – Status Register (address offset: 0x04).....167
15.5.3 SPIM_DATA – Receiver and Transmitter Data Registers (address offset: 0x08).....168
15.5.4 SPIM_SLV_SEL_CNTL – Slave Select Control Register (address offset: 0x0C).....168
15.5.5 SPIM_FIFO_CNTL – FIFO Control Register (address offset: 0x10).....168
15.5.6 SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14) 169
15.5.7 SPIM_ALT_DATA – Alternative SPI Master Data Register (address offset: 0x18).....169
15.5.8 SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register (address offset: 0x1C) 170

Copyright © Bridgetek Pte Ltd

15.5.9 SPIM_CNTL_2 – Control Register 2 (address offset: 0x00) 170
15.5.10 SPIM_STATUS_2 – Status Register 2 (address offset: 0x10) 170
15.5.11 SPIM_FIFO_CNTL_2 – FIFO Control Register 2 (address offset: 0x14) 171
15.5.12 SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register (address offset: 0x18) 171
15.5.13 SPIM_BAUD – SPI Master Baud Register (address offset: 0x1C) 171

16 SPI Slaves 173

16.1 Register Summary 173

16.2 Register/FIFO Access Modes and FIFO Size 174

16.3 Block Diagram 175

16.4 SPI Slave Interrupts 176

16.4.1 Receiver FIFO Trigger Level Interrupt 176
16.4.2 Transmitter FIFO Trigger Level Interrupt 176
16.4.3 Receiver FIFO Timeout Interrupt 176
16.4.4 Transmitter FIFO Empty Interrupt 176
16.4.5 FIFO Polling Mode 177

16.5 Register Details 177

16.5.1 SPIS_CNTL – Control Register (address offset: 0x00) 177
16.5.2 SPIS_STATUS – Status Register (address offset: 0x04) 177
16.5.3 SPIS_DATA – Receiver and Transmitter Data Registers (address offset: 0x08) 178
16.5.4 SPIS_RESERVED – Reserved Register (address offset: 0x0C) 178
16.5.5 SPIS_FIFO_CNTL – FIFO Control Register (address offset: 0x10) 178
16.5.6 SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14) 180
16.5.7 SPIS_ALT_DATA – Alternative SPI Slave Data Register (address offset: 0x18) 180
16.5.8 SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register (address offset: 0x1C) .181
16.5.9 SPIS_CNTL_2 – Control Register 2 (address offset: 0x00) 181
16.5.10 SPIS_STATUS_2 – Status Register 2 (address offset: 0x10) 181
16.5.11 SPIS_FIFO_CNTL_2 – FIFO Control Register 2 (address offset: 0x14) 183
16.5.12 SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register (address offset: 0x18) 183

17 I²C Master 184

17.1 Register Summary 184

17.2 I²C Master FIFO Mode 185

17.3 Register Details 185

Copyright © Bridgetek Pte Ltd

17.3.1 I2CM_SLV_ADDR – Slave Address Register (address offset: 0x00) 185
17.3.2 I2CM_CNTL – Control Register (address offset: 0x01).....186
17.3.3 I2CM_STATUS – Status Register (address offset: 0x01).....188
17.3.4 I2CM_DATA – Receive / Transmit Data Register (address offset: 0x02) 188
17.3.5 I2CM_TIMER_PERIOD – Timer Period Register (address offset: 0x03).....188
17.3.6 I2CM_HS_TIMER_PERIOD – High-speed Timer Period Register (address offset: 0x03) 189
17.3.7 I2CM_FIFO_LEN – FIFO Mode Byte Length (address offset: 0x04) 189
17.3.8 I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05).....189
17.3.9 I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06) 190
17.3.10 I2CM_FIFO_DATA – FIFO Data Register (address offset: 0x07) 190
17.3.11 I2CM_TRIG – FIFO Operation Trigger Register (address offset: 0x08) 190

18 I²C Slave 191

18.1 Register Summary.....191

18.2 I²C Slave FIFO Mode 191

18.3 Register Details.....192

18.3.1 I2CS_OWN_ADDR – Own Address Register (address offset: 0x00) 192
18.3.2 I2CS_CNTL – Control Register (address offset: 0x01) 192
18.3.3 I2CS_STATUS – Status Register (address offset: 0x01) 193
18.3.4 I2CS_DATA – Receive / Transmit Data Register (address offset: 0x02) 194
18.3.5 I2CS_FIFO_STATUS – FIFO Mode Status Register (address offset: 0x04) 194
18.3.6 I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05) 194
18.3.7 I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06).....194
18.3.8 I2CS_FIFO_DATA – FIFO Data Register (address offset: 0x07) 195

19 RTC 196

19.1 Register Summary.....196

19.2 Functional Description 196

19.2.1 RTC Alive Module 196
19.2.2 RTC Wrapper Module 197
19.2.3 RTC Operation 197
19.2.4 RTC Auto Calibration using a 32,768 Hz Reference Clock.....197

19.3 Register Details.....198

19.3.1 RTC_TIME – Time Register (address offset: 0x00).....198
19.3.2 RTC_DATE – Date Register (address offset: 0x04).....199

Copyright © Bridgetek Pte Ltd

19.3.3 RTC_AALARM1 – Auto Alarm 1 Register (address offset: 0x08) 199
19.3.4 RTC_AALARM2 – Auto Alarm 2 Register (address offset: 0x0C) 200
19.3.5 RTC_CTRL – Control Register (address offset: 0x10) 200
19.3.6 RTC_STAT – Status Register (address offset: 0x14) 201
19.3.7 RTC_TRIM – Trimming Register (address offset: 0x18) 202
19.3.8 RTC_SLEEP_TIME Sleep Time Register (address offset: 0x1C) 202
19.3.9 RTC_SLEEP_DATE – Sleep Date Register (address offset: 0x20) 202
19.3.10 RTC_COMP_PER – Compare Period Register (address offset: 0x20) 203
19.3.11 RTC_MISC – Miscellaneous Register.....203

20 PWM..... 204

20.1 Register Summary.....204

20.2 Functional Description 206

20.2.1 Block Diagram 206
20.2.2 Duty Cycle Generation Example 206

20.3 Register Details.....208

20.3.1 PWM_CTRL0 – PCM Control Register (address offset: 0x00) 208
20.3.2 PWM_CTRL1 – PWM Control Register (address offset: 0x01).....208
20.3.3 PWM_PRESCALER – PWM Pre-scaler Register (address offset: 0x02) 209
20.3.4 PWM_CNTL – PWM Counter Register (LSB) (address offset: 0x03) 209
20.3.5 PWM_CNTH – PWM Counter Register (MSB) (address offset: 0x04) 209
20.3.6 PWM_CMP0L – Comparator 0 Value Register (LSB) (address offset: 0x05) 209
20.3.7 PWM_CMP0H – Comparator 0 Value Register (MSB) (address offset: 0x06) 209
20.3.8 PWM_CMP1L – Comparator 1 Value Register (LSB) (address offset: 0x07) 209
20.3.9 PWM_CMP1H – Comparator 1 Value Register (MSB) (address offset: 0x08) 209
20.3.10 PWM_CMP2L – Comparator 2 Value Register (LSB) (address offset: 0x09).....210
20.3.11 PWM_CMP2H – Comparator 2 Value Register (MSB) (address offset: 0x0A) 210
20.3.12 PWM_CMP3L – Comparator 3 Value Register (LSB) (address offset: 0x0B) 210
20.3.13 PWM_CMP3H – Comparator 3 Value Register (MSB) (address offset: 0x0C) 210
20.3.14 PWM_CMP4L – Comparator 4 Value Register (LSB) (address offset: 0x0D) 210
20.3.15 PWM_CMP4H – Comparator 4 Value Register (MSB) (address offset: 0x0E) 210
20.3.16 PWM_CMP5L – Comparator 5 Value Register (LSB) (address offset: 0x0F).....210
20.3.17 PWM_CMP5H – Comparator 5 Value Register (MSB) (address offset: 0x10) 211
20.3.18 PWM_CMP6L – Comparator 6 Value Register (LSB) (address offset: 0x11).....211
20.3.19 PWM_CMP6H – Comparator 6 Value Register (MSB) (address offset: 0x12) 211

Copyright © Bridgetek Pte Ltd

20.3.20 PWM_CMP7L – Comparator 7 Value Register (LSB) (address offset: 0x13).....211
20.3.21 PWM_CMP7H – Comparator 7 Value Register (MSB) (address offset: 0x14) 211
20.3.22 PWM_TOGGLE0 – Channel 0 OUT Toggle Comparator Mask Register (address offset: 0x15) 211
20.3.23 PWM_TOGGLE1 – Channel 1 OUT Toggle Comparator Mask Register (address offset: 0x16) 212
20.3.24 PWM_TOGGLE2 – Channel 2 OUT Toggle Comparator Mask Register (address offset: 0x17) 212
20.3.25 PWM_TOGGLE3 – Channel 3 OUT Toggle Comparator Mask Register (address offset: 0x18) 212
20.3.26 PWM_TOGGLE4 – Channel 4 OUT Toggle Comparator Mask Register (address offset: 0x19) 212
20.3.27 PWM_TOGGLE5 – Channel 5 OUT Toggle Comparator Mask Register (address offset: 0x1A) 212
20.3.28 PWM_TOGGLE6 – Channel 6 OUT Toggle Comparator Mask Register (address offset: 0x1B) 213
20.3.29 PWM_TOGGLE7 – Channel 7 OUT Toggle Comparator Mask Register (address offset: 0x1C) 213
20.3.30 PWM_OUT_CLR_EN – PWM OUT Clear Enable Register (address offset: 0x1D) 213
20.3.31 PWM_CTRL_BL_CMP8 – Control Block CMP8 Value Register (address offset: 0x1E) 213
20.3.32 PWM_INIT – PWM Initialization Register (address offset: 0x1F).....213
20.3.33 PWM_INTMASK – PWM Interrupt Mask Register (address offset: 0x20) 213
20.3.34 PWM_INTSTATUS – PWM Interrupt Status Register (address offset: 0x21) 214
20.3.35 PWM_SAMPLE_FREQ_H – PWM Data Sampling Frequency High Byte Register (address offset: 0x22) 214
20.3.36 PWM_SAMPLE_FREQ_L – PWM Data Sampling Frequency Low Byte Register (address offset: 0x23) 214
20.3.37 PCM_VOLUME – PCM Volume Register (address offset: 0x24) 215
20.3.38 PWM_BUFFER – PCM Buffer Register (address offset: 0x3C) 215

21 Data Capture Interface 216

21.1 Register Summary.....216

21.2 Register Details.....216

21.2.1 DCAP_REG1 – Data Capture Interface Register 1 (address offset: 0x00) 216
21.2.2 DCAP_REG2 – Data Capture Interface Register 2 (address offset: 0x04) 216
21.2.3 DCAP_REG3 – Data Capture Interface Register 3 (address offset: 0x08) 217
21.2.4 DCAP_REG4 – Data Capture Interface Register 4 (address offset: 0x0C) 217

22 Flash Controller..... 218

Copyright © Bridgetek Pte Ltd

22.1 Register Summary.....218

22.2 Register Details.....219

22.2.1 RSADDR0 – Memory Start Address Register (LSB) (address offset: 0x00) 219

22.2.2 RSADDR1 – Memory Start Address Register (Byte 1) (address offset: 0x01).....219

22.2.3 RSADDR2 – Memory Start Address Register (MSB) (address offset: 0x02) 219

22.2.4 FSADDR0 – Flash Start Address Register (LSB) (address offset: 0x03) 220

22.2.5 FSADDR1 – Flash Start Address Register (Byte 1) (address offset: 0x04).....220

22.2.6 FSADDR2 – Flash Start Address Register (MSB) (address offset: 0x05).....220

22.2.7 BLENGTH0 – Data Byte Length Register (LSB) (address offset: 0x06).....220

22.2.8 BLENGTH1 – Data Byte Length Register (Byte 1) (address offset: 0x07) 220

22.2.9 BLENGTH2 – Data Byte Length Register (MSB) (address offset: 0x08) 220

22.2.10 COMMAND – Command Register (address offset: 0x09) 221

22.2.11 SEMAPHORE – Semaphore Register (address offset: 0x0B).....221

22.2.12 CONFIG – Configuration Register (address offset: 0x0C) 221

22.2.13 STATUS – Status Register (address offset: 0x0D).....221

22.2.14 CRCL – Flash Content CRC Register (LSB) (address offset: 0x0E) 222

22.2.15 CRCH – Flash Content CRC Register (MSB) (address offset: 0x0F).....222

22.2.16 CHIPID0 – Chip ID Register (LSB) (address offset: 0x7C).....222

22.2.17 CHIPID1 – Chip ID Register (Byte 1) (address offset: 0x7D).....222

22.2.18 CHIPID2 – Chip ID Register (Byte 2) (address offset: 0x7E) 222

22.2.19 CHIPID3 – Chip ID Register (MSB) (address offset: 0x7F) 222

22.2.20 DRWDATA – Data Register (address offset: 0x80) 223

22.3 Flash Controller Commands 223

23 Analog Interfaces 227

23.1 ADC.....227

23.2 DAC.....227

24 Security via OTP..... 229

24.1 OTP Bitmap & Definition.....229

24.2 OTP Access.....229

25 Contact Information..... 230

Appendix A – References 231

Document References 231

Copyright © Bridgetek Pte Ltd

Acronyms and Abbreviations.....231

Appendix B – List of Tables & Figures 233

List of Tables.....233

List of Figures 245

Appendix C – Revision History 246

Copyright © Bridgetek Pte Ltd

1 Introduction

The FT90X series consists of several programmable System-on-Chip devices with a 32-bit general purpose embedded microprocessor core and a plethora of connectivity options. It is developed for high speed, data bridging tasks. With a parallel data capture interface, 10/100 Base-TX Ethernet interface, CAN bus, and USB 2.0 Hi-Speed device and host ports, this series offers excellent interconnect capabilities and blazing computational power.

This document describes the general system registers, as well as the register set of various peripheral interfaces for Revision C of the FT90x devices. For information about the previous revision, please refer to AN_324 FT900 User Manual.

Copyright © Bridgetek Pte Ltd

2 FT90X Revision C System Architecture

2.1 Architecture Overview

The FT90X core contains the 32-bit CPU (FT32), with control logic, flash memory and RAM. The flash memory size is 256 KB. The RAM consists of 256 KB shadow program memory and 64 KB data memory. Upon reset, the content of the flash memory is copied into the shadow program memory for fastest execution. The outside connections for the FT90X core are the memory-mapped I/O interface, the interrupt interface, synchronous reset and the clock.

The peripherals of the FT90x series include:

- 100 MHz 32-bit capable RISC core
- 1 x high-speed USB host controller, which supports USB Battery Charging Specification Revision 1.2. It can be configured as SDP, CDP or DCP.
- 1 x high-speed USB device controller, which supports USB Battery Charging Specification Revision 1.2. It can perform BCD mode detection.
- 2 x programmable UARTs
- 1 x SPI master controller
- 2 x SPI slave controllers
- 8 x PWM channels with digital filter on channel 0 and 1
- 1 x I^2C master controller
- 1 x I^2C slave controller
- 1 x I^2S master / slave controller
- 1 x SD host controller
- 2 x CAN bus controllers
- 1 x Ethernet controller
- 1 x real-time clock
- 1 x watchdog timer
- 4 x 16-bit general purpose timers
- One-wire debug interface
- 7-channel 8-bit/10-bit 960 kS/s ADC
- 2-channel 10-bit 1 MS/s DAC
- 67 multi-purpose GPIOs
- Nested Vectored Interrupt Controller (NVIC)

Copyright © Bridgetek Pte Ltd

The block diagram in Figure 2.1 illustrates the main IP blocks of FT900 Revision C.

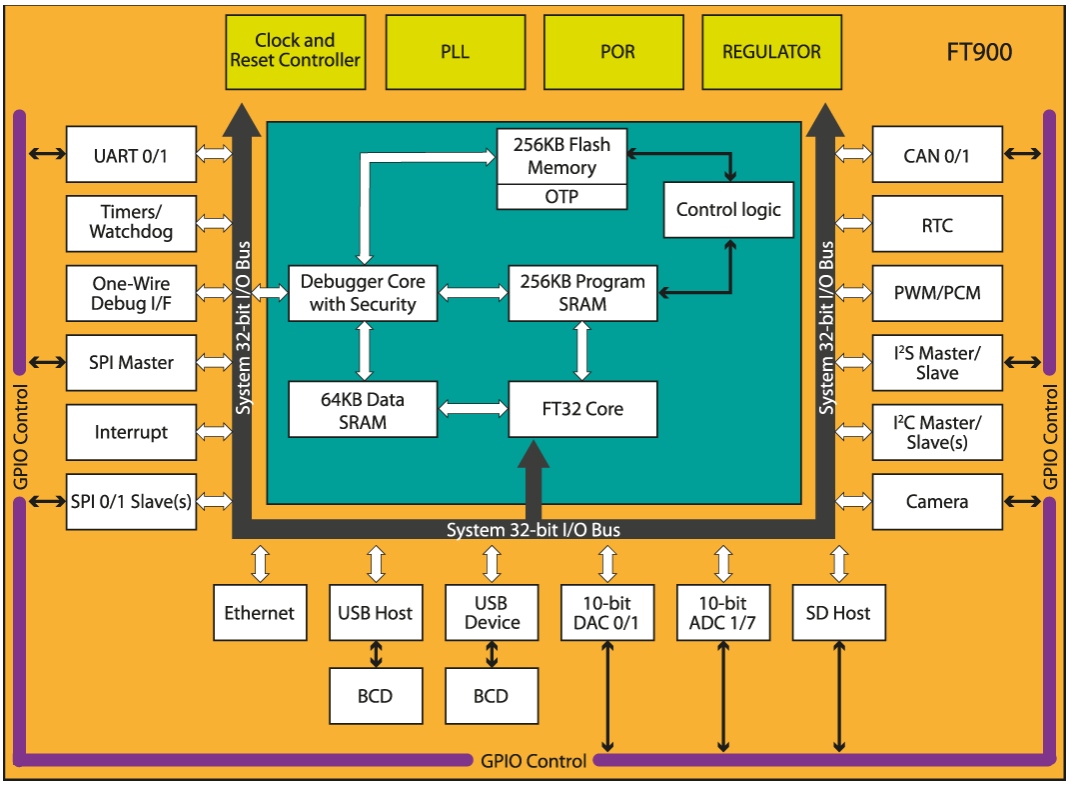


Figure 2.1 - FT900 Revision C System Architecture

2.2 Memory Organization

The first 144 bytes in the Program Memory contains the followings:

- Reset vector
- Watchdog vector
- 32 interrupt vectors
- 1 non-maskable interrupt vector (reserved for the debugger)
- Program entry point

Address	Function
0x00	Reset vector
0x04	Watchdog vector
0x08	Interrupt vector 0
0x0C	Interrupt vector 1
…	…
0x80	Interrupt vector 30
0x84	Interrupt vector 31
0x88	Interrupt vector 32 (NMI)
0x8C	Program entry point

Table 2.1 - FT900 Program Memory Organization

Copyright © Bridgetek Pte Ltd

2.3 FT90X Boot Control

Upon reset, boot control takes control of the memory buses and puts the CPU in a reset state.

It automatically transfers the data from the flash memory to the CPU program memory, starting from address 0 on both sides. Boot control calculates a CRC check over the entire contents of flash (256 KB) and the result is placed in CRCH and CRCL registers found in the flash control module.

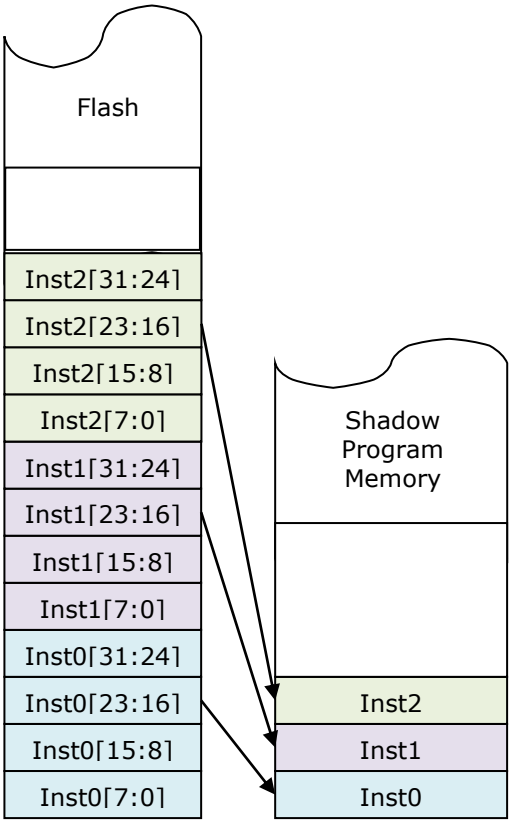


Figure 2.2 - FT90X Boot Control

2.4 Debugging Support

Debugging the FT90X series is carried out via the one-wire debug interface. The debugging support is implemented in the FT90X bootloader. The protocol used for debugging is the GDB remote protocol, and a port of GDB is available in the Bridgetek FT9XX Toolchain. The GDB serial debug protocol commands are interpreted by a debug interpreter in the bootloader.

In addition, the debug interpreter:

- saves all machine states
- executes commands received over the debug interface
- restores all machine states and returns

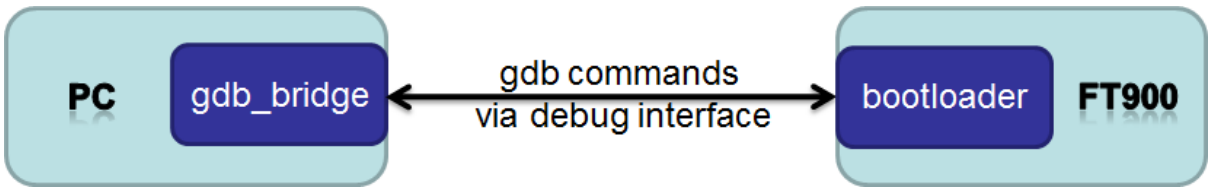


Figure 2.3 - FT90X Debugging Support

Copyright © Bridgetek Pte Ltd

3 Register Map

This section lists the I/O map for registers / memory in the device.

Refer to the table below for five types of peripherals availability across the models of FT900 series. All other peripherals that are not mentioned are available on all models in the series.

	CAN	Ethernet	Camera	SD Host	I2S
FT900Q/FT900L	X	X	X	X	X
FT901Q/FT901L	-	X	X	X	X
FT902Q/FT902L	X	-	X	X	X
FT903Q/FT903L	-	-	X	X	X
FT905Q/FT905L	X	X	-	-	-
FT906Q/FT906L	-	X	-	-	-
FT907Q/FT907L	X	-	-	-	-
FT908Q/FT908L	-	-	-	-	-

Legend: X indicates the peripheral exists on the model.

- (minus) indicates the peripheral is not available on the model.

Table 3.1 - Peripheral Availability in FT90X Series Revision C Models

Refer to the table below for register map of peripherals:

Function	Address Base Range		Access Mechanism*
General Setup	0x10000	0x100BF	DW, W, B
Interrupt Controller	0x100C0	0x100FF	DW, W, B
USB Host	0x10100	0x1017F	DW, W, B
USB Host RAM	0x11000	0x12FFF	DW, W, B
USB Device	0x10180	0x1021F	B (DW, W, B for EP buffer registers)
USB Device (High-Bandwidth Isochronous)	0x10A1C	0x10A33	DW, W, B
Ethernet	0x10220	0x1023F	DW, W, B (DW for FIFO)
CAN 0	0x10240	0x1025F	B
CAN 1	0x10260	0x1027F	B
SPI Master	0x102A0	0x102BF	DW
SPI Master (extended)	0x10940	0x1095F	DW
SPI Slave 0	0x102C0	0x102DF	DW
SPI Slave 0 (extended)	0x10960	0x1097F	DW
SPI Slave 1	0x102E0	0x102FF	DW
SPI Slave 1 (extended)	0x10980	0x1099F	DW
I2C Master	0x10300	0x1030F	B

Copyright © Bridgetek Pte Ltd

Function	Address Base Range		Access Mechanism*
I2C Slave	0x10310	0x1031F	B
UART 0	0x10320	0x1032F	B
UART 1	0x10330	0x1033F	B
Timers/Watchdog	0x10340	0x1034F	B
I2S (Master/Slave)	0x10350	0x1035F	W
Data Capture	0x10360	0x1036F	DW
PWM	0x103C0	0x103FF	B for registers, W for FIFO
SD Host	0x10400	0x107FF	DW
Flash Controller	0x10800	0x108BF	B
RTC	0x10900	0x1093F	DW

***Legend:**

DW (Double-Word): 32-bit

W (Word): 16-bit

B (Byte): 8-bit

Table 3.2 - Register Map for FT90X Series Revision C

Copyright © Bridgetek Pte Ltd

4 Notations

4.1 Bit Type Notation

These notations are used in the register descriptions to describe the bit type:

Terms	Description
Reserved	Reserved location, do not use / modify
RAC	Read-able with automatic clear
RO	Read-only
ROC	Read-only / Cleared-when-read
RW	Read- and Write-able
RW1C	Read and Write-1-to-clear
RW1S	Read and Write-1-to-set
RWAC	Read- and Write-able with automatic clear
W1S	Write-1-to-set
W1T	Write-1-to-trigger-event
WO	Write-only

Table 4.1 - Notations used in Register Description

4.2 Default Bit Value Notation

When several bits form a field in a register, the following notation is used to represent the value of the field.

<Field length>'<Value Format><Value>

For example, a field with a default value of **8'h04** means that there are 8 bits in the field and the value of the field is 04 hex (i.e., 0x04).

The formats used for the default values are either hex ('h) or binary ('b).

4.3 X Default Value

"X" denotes "don't care" value which means a non-determinable (unknown) default value of a bit at a particular point of time.

4.4 Bit Range Symbol in Bit Column

Colon (:) will be used to represent the bit range in "Bit" column of Register tables. Bit range is usually given in descending order in the "Bit" column.

N:M

where N is a greater number than M. This indicates a bit range from bit N to bit M (in descending order). For example, 26:20 indicates a bit range from bit 26 to bit 20.

Copyright © Bridgetek Pte Ltd

Example 1:

Bit	Name	Type	Default Value	Description
31	CAN_ACTIVE	RO	X	CAN modules available; the value depends on device model; 1 – available; 0 – not available.
30	MAC_ACTIVE	RO	X	Ethernet module available; default value depends on device model; 1 – available; 0 – not available.
29	100_PIN	RO	X	1 - the device is a 100-pin device; 0 - the device is a 76-pin (QFN) or 80-pin (LQFP) device.
28	Reserved	RO	0	Always read as '0'
27	ONE-WIRE_ACTIVE	RO	X	If set, one-wire debug interface is enabled; otherwise, it's permanently disabled.
26:20	Reserved	RO	X	Reserved
19	FLASH_WR_B3_ENA	RO	X	0 - Flash is permanently non-writable/ non-erasable. 1 - Allow the Flash to write/erase from 0x30000 to 0x3ffff bytes.

Example 2:

A register is assigned to bits[39:8] of the command field.

Bit	Name	Type	Default Value	Description
31:0	ARG1	RW	32'h00000000	Command argument

Copyright © Bridgetek Pte Ltd

5 General System Registers

This section describes the registers that govern the general behavior of the FT90X series.

5.1 Register Summary

Listed below are the registers with their offset from the base address (0x10000). All registers can be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit) mode.

Address Offset	Register	Default Value	References
0x00	CHIPID - Chip ID Register	0x09XX XXXX	Section 5.2.1
0x04	CHIPCFG - Chip Configuration Register	0xXXXX XXXX	Section 5.2.2
0x08	CLKCFG - Clock Configuration Register	0x0000 0000	Section 5.2.3
0x0C	PMCFG - Power Management Register	0x0000 0000	Section 5.2.4
0x10	PTSTNSET - Test & Set Register	0x0000 1000	Section 5.2.5
0x14	PTSTNSETR - Test & Set Shadow Register	0x0000 0000	Section 5.2.6
0x18	MSC0CFG - Miscellaneous Configuration Register	0x0000 0000	Section 5.2.7
0x1C	Pin 00 – 03 Register	0x0408 0808	Section 5.2.8.1
0x20	Pin 04 – 07 Register	0x0404 0404	Section 5.2.8.2
0x24	Pin 08 – 11 Register	0x0404 0404	Section 5.2.8.3
0x28	Pin 12 – 15 Register	0x0404 0404	Section 5.2.8.4
0x2C	Pin 16 – 19 Register	0x0404 0404	Section 5.2.8.5
0x30	Pin 20 – 23 Register	0x0404 0404	Section 5.2.8.6
0x34	Pin 24 – 27 Register	0x0404 0404	Section 5.2.8.7
0x38	Pin 28 – 31 Register	0x0404 0404	Section 5.2.8.8
0x3C	Pin 32 – 35 Register	0x0404 0404	Section 5.2.8.9
0x40	Pin 36 – 39 Register	0x0404 0404	Section 5.2.8.10
0x44	Pin 40 – 43 Register	0x0404 0404	Section 5.2.8.11
0x48	Pin 44 – 47 Register	0x0404 0404	Section 5.2.8.12
0x4C	Pin 48 – 51 Register	0x0404 0404	Section 5.2.8.13
0x50	Pin 52 – 55 Register	0x0404 0404	Section 5.2.8.14
0x54	Pin 56 – 59 Register	0x0404 0404	Section 5.2.8.15

Copyright © Bridgetek Pte Ltd

0x58	Pin 60 – 63 Register	0x0404 0404	Section 5.2.8.16
0x5C	Pin 64 – 66 Register	0x0404 0404	Section 5.2.8.17
0x60	GPIO 00 – 07 Configuration Register	0x0000 0000	Section 5.2.9.1
0x64	GPIO 08 – 15 Configuration Register	0x0000 0000	Section 5.2.9.2
0x68	GPIO 16 – 23 Configuration Register	0x0000 0000	Section 5.2.9.3
0x6C	GPIO 24 – 31 Configuration Register	0x0000 0000	Section 5.2.9.4
0x70	GPIO 32 – 39 Configuration Register	0x0000 0000	Section 5.2.9.5
0x74	GPIO 40 – 47 Configuration Register	0x0000 0000	Section 5.2.9.6
0x78	GPIO 48 – 55 Configuration Register	0x0000 0000	Section 5.2.9.7
0x7C	GPIO 56 – 63 Configuration Register	0x0000 0000	Section 5.2.9.8
0x80	GPIO 64 – 66 Configuration Register	0x0000 0000	Section 5.2.9.9
0x84	GPIO 00 – 31 Value Register	0x0000 0000	Section 5.2.10.1
0x88	GPIO 32 – 63 Value Register	0x0000 0000	Section 5.2.10.2
0x8C	GPIO 64 – 66 Value Register	0x0000 0000	Section 5.2.10.3
0x90	GPIO 00 – 31 Interrupt Enable Register	0x0000 0000	Section 5.2.11.1
0x94	GPIO 32 – 63 Interrupt Enable Register	0x0000 0000	Section 5.2.11.2
0x98	GPIO 64 – 66 Interrupt Enable Register	0x0000 0000	Section 5.2.11.3
0x9C	GPIO 00 – 31 Interrupt Pending Register	0x0000 0000	Section 5.2.12.1
0xA0	GPIO 32 – 63 Interrupt Pending Register	0x0000 0000	Section 5.2.12.2
0xA4	GPIO 64 – 66 Interrupt Pending Register	0x0000 0000	Section 5.2.12.3
0xA8	ETH_PHY_CFG - Ethernet PHY Miscellaneous Configuration Register	0x0007 0300	Section 5.2.13
0xAC	ETH_PHY_ID - Ethernet PHY ID Register	0x0000 0000	Section 5.2.14
0xB0	DAC_ADC_CONF - ADC/DAC Configuration/Status Register	0x0000 0000	Section 5.2.15
0xB4	DAC_ADC_CNT - ADC/DAC Count Register	0x6300 0000	Section 5.2.16
0xB8	DAC_ADC_DATA - ADC/DAC Data Register	0x0000 0000	Section 5.2.17
0xBC	RESET_STATUS – Global Reset/Status Register	0x0000 0000	Section 5.2.18

Table 5.1 - Overview of General System Registers

Copyright © Bridgetek Pte Ltd

5.2 Register Details

5.2.1 CHIPID - Chip ID Register (address offset: 0x00)

This register is read-only.

Bit	Name	Type	Default Value	Description
31:0	CHIP_ID	RO	0x09XX XXXX	The two MSBs (09XX) depict FT90x series and the two LSBs (XXXX) show the revision of the chip.

Table 5.2 - CHIPID - Chip ID Register

The table below lists the pre-configured bits of the chip ID register (CHIPID) and the chip configuration register (CHIPCFG, section 5.2.2) for different models in **revision B** of the FT90X series. Note that these bits are always read-only.

	CHIPID [31..16]	CHIPID [15..0]	CHIPCFG [31]	CHIPCFG [30]	CHIPCFG [29]
FT900Q/FT900L	0X0900	0X0001	1	1	1
FT901Q/FT901L	0X0901	0X0001	0	1	1
FT902Q/FT902L	0X0902	0X0001	1	0	1
FT903Q/FT903L	0X0903	0X0001	0	0	1
FT905Q/FT905L	0X0905	0X0001	1	1	0
FT906Q/FT906L	0X0906	0X0001	0	1	0
FT907Q/FT907L	0X0907	0X0001	1	0	0
FT908Q/FT908L	0X0908	0X0001	0	0	0

Table 5.3 - FT90X Series Revision B Configuration

The table below lists the pre-configured bits of the chip ID register (CHIPID) and the chip configuration register (CHIPCFG, section 5.2.2) for different models in **revision C** of the FT90X series. Note that these bits are always read-only.

	CHIPID [31..16]	CHIPID [15..0]	CHIPCFG [31]	CHIPCFG [30]	CHIPCFG [29]
FT900Q/FT900L	0X0900	0X0002	1	1	1
FT901Q/FT901L	0X0901	0X0002	0	1	1
FT902Q/FT902L	0X0902	0X0002	1	0	1
FT903Q/FT903L	0X0903	0X0002	0	0	1
FT905Q/FT905L	0X0905	0X0002	1	1	0
FT906Q/FT906L	0X0906	0X0002	0	1	0
FT907Q/FT907L	0X0907	0X0002	1	0	0
FT908Q/FT908L	0X0908	0X0002	0	0	0

Table 5.4 - FT90X Series Revision C Configuration

5.2.2 CHIPCFG - Chip Configuration Register (address offset: 0x04)

This register contains read-only information. Bits [19:16] reflects the write-protection bits in the OTP. See Section 24.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
31	CAN_ACTIVE	RO	X	CAN modules available; the value depends on device model; 1 – available; 0 – not available.
30	MAC_ACTIVE	RO	X	Ethernet module available; default value depends on device model; 1 – available; 0 – not available.
29	100_PIN	RO	X	1 - the device is a 100-pin device; 0 - the device is a 76-pin (QFN) or 80-pin (LQFP) device.
28	Reserved	RO	0	Always read as '0'.
27	ONE-WIRE_ACTIVE	RO	X	If set, one-wire debug interface is enabled; otherwise, it is permanently disabled.
26:20	Reserved	RO	X	Reserved
19	FLASH_WR_B3_ENA	RO	X	0 - Flash is permanently non-writable/non-erasable. 1 - Allow the Flash to write/erase from 0x30000 to 0x3ffff bytes.
18	FLASH_WR_B2_ENA	RO	X	0 - Flash is permanently non-writable/non-erasable. 1 - Allow the Flash to write/erase from 0x20000 to 0x2ffff bytes.
17	FLASH_WR_B1_ENA	RO	X	0 - Flash is permanently non-writable/non-erasable. 1 - Allow the Flash to write/erase from 0x10000 to 0x1ffff bytes.
16	FLASH_WR_B0_ENA	RO	X	0 - Flash is permanently non-writable/non-erasable. 1 - Allow the Flash to write/erase from 0x00000 to 0x0ffff bytes.
15:0	Reserved	RO	X	Reserved

Table 5.5 - CHIPCFG - Chip Configuration Register

5.2.3 CLKCFG - Clock Configuration Register (address offset: 0x08)

Bit	Name	Type	Default Value	Description
31:20	Reserved	-	-	Reserved
19:16	CPU_CLK_DIV	RW	0	CPU Clock frequency setting: 0: CPU runs at system clock speed (100 MHz) 1: CPU runs at system clock / 2 2: CPU runs at system clock / 4 3: CPU runs at system clock / 8 4: CPU runs at system clock / 64 5: CPU runs at system clock / 128 6: CPU runs at system clock / 512 Others: CPU is stopped.

Copyright © Bridgetek Pte Ltd

15	EHCI_ENA	RW	0	1: enable USB Host
14	DEV_ENA	RW	0	1: enable USB Device
13	MAC_ENA	RW	0	1: enable Ethernet
12	SD_ENA	RW	0	1: enable SD Host
11	CAN0_ENA	RW	0	1: enable CAN0
10	CAN1_ENA	RW	0	1: enable CAN1
9	I2CM_ENA	RW	0	1: enable I2C Master
8	I2CS_ENA	RW	0	1: enable I2C Slave
7	SPIM_ENA	RW	0	1: enable SPI Master
6	SPIS0_ENA	RW	0	1: enable SPI Slave 0
5	SPIS1_ENA	RW	0	1: enable SPI Slave 1
4	UART0_ENA	RW	0	1: enable UART 0
3	UART1_ENA	RW	0	1: enable UART 1
2	PWM _ENA	RW	0	1: enable PWM
1	I2S _ENA	RW	0	1: enable I2S
0	CAM _ENA	RW	0	1: enable Data Capture Interface

Table 5.6 - CLKCFG - Clock Configuration Register

5.2.4 PMCFG - Power Management Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
31	RTC_ALARM_IRQ_PEND	RW1C	0	RTC alarm interrupt pending; write 1 to clear. If RTC_VBAT_IRQ_EN=1, an interrupt will be generated.
30	RTC_VBAT_LOW_PEND	RW1C	0	RTC battery low detected pending: Write 1 to clear. If RTC_VBAT_IRQ_EN=1, an interrupt will be generated. *Clear this bit before setting RTC_VBAT_IRQ_EN to "1".*
29	RTC VBAT REPLACED_PEND	RW1C	0	RTC good battery replacement detected pending: Write 1 to clear. If RTC_VBAT_IRQ_EN=1, an interrupt will be generated. *Clear this bit before setting RTC_VBAT_IRQ_EN to "1".*
28:26	Reserved	RO	0	
25	PM_GPIO_IRQ_PEND	RW1C	0	GPIO interrupt during system shut down with clock not running
24	SLOWCLK_5ms_IRQ_PEND	RW1C	0	Slow clock 5ms timer interrupt pending. If enabled, an interrupt will be generated.
23	RM_WK_HOST	RW1C	0	Remote Wakeup Interrupt pending

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
				to USB Host; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
22	DEV_CONN_HOST	RW1C	0	Device Connect Interrupt pending to USB Host; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
21	DEV_DIS_HOST	RW1C	0	Device Disconnect Interrupt pending to USB Host; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
20	DEV_CONN_DEV	RW1C	0	Device Connect Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
19	DEV_DIS_DEV	RW1C	0	Device Disconnect Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
18	HOST_RST_DEV	RW1C	0	Host Reset Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
17	HOST_RESUME_DEV	RW1C	0	Host Resume Interrupt pending to USB Device; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
16	OC_DETECT	RW1C	0	Over current detected Interrupt pending if set; write 1 to clear. If enabled, an interrupt will be generated on PM IRQ.
15	PSWN	RW	1	0: Set PSWN to 0 (Enables external VBUS) 1: Set PSWN to 1 (Disables external VBUS) *When OC_DETECT_EN is set and an over current condition occurs, hardware will automatically set this bit.* *This bit cannot be cleared by software as long as the condition exists.*
14	RTC_VBAT_STATUS_DETECT	RW	0	Chooses which RTC battery condition is detected. 0: Detects if the RTC battery is low. 1: Detects if a good battery has been replaced.
13	RTC_VBAT_IRQ_EN	RW	0	1: Enables the interrupt for the battery condition that was selected via bit 14.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
12	RTC_ALARM_IRQ_EN	RW	0	1 : Enable RTC Alarm Interrupt.
11	Reserved	RO	0	-
10	DEV_PHY_EN	RW	0	1: Enable USB Device PHY
9	PM_PWRDN_MODE	RW	0	1: disable system oscillator when powering down. 0: do not disable system oscillator when powering down.
8	PM_PWRDN	RW	0	1: power down system. This bit should be cleared after the system wakes up, or at least 60-100us prior to setting it to 1 again.
7	SLOWCLOCK_5ms_IRQ_EN	RW	0	1: enable slow clock 5ms timer interrupt.
6	SLOWCLOCK_5ms_START	RWAC	0	1: To start the 1-shot slow clock 5ms timer; once started, it cannot be stopped. This bit will be cleared automatically when the timer expires.
5	FORCE_HOST_DET	RW	0	Normally USB host activity detection is performed only when required; setting this bit will force the PM to check for host connection activities regardless.
4	FORCE_DEV_DET	RW	0	Normally, USB device activity detection is performed only when required; setting this bit will force the PM to check for device connection activities regardless.
3	RM_WK_HOST_EN	RW	0	1: enable remote wake up detection to USB host. Enable interrupt to PM IRQ when RM_WK_HOST is set.
2	HOST_DETECT_EN	RW	0	1: enable device connect/disconnect detection to USB Host. Enable interrupt to PM IRQ when either DEV_CONN_HOST or DEV_DIS_HOST is set.
1	DEV_DETECT_EN	RW	0	1: enable device connect/disconnect to external host or external host reset detection. Enable interrupt to PM IRQ when any of DEV_CONN_DEV, DEV_DIS_DEV, HOST_RESUME_DEV and HOST_RST_DEV is set.
0	OC_DETECT_EN	RW	0	1: Enable Over current detection. Enable interrupt to PM IRQ when OC_DETECT is set

Table 5.7 - PMCFG - Power Management Register

Copyright © Bridgetek Pte Ltd

5.2.5 PTSTNSET - Test & Set Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
31:1	Reserved	RO	0	
0	TEST_SET	RW1C	0	This register is used as a binary semaphore. Upon a read, a 0 indicates that the semaphore has been granted to the reader and a 1 indicates that it has already been taken. **CAUTION**: Only the granted thread or process shall write a 1 to return the semaphore.

Table 5.8 - PTSTNSET - Test & Set Register

5.2.6 PTSTNSETR - Test & Set Shadow Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
31:1	Reserved	RO	0	
0	TEST_SET	RW1C	0	This register is used as a binary semaphore. Upon a read, a 0 indicates that the semaphore has been granted to the reader and a 1 indicates that it has already been taken. **CAUTION**: Only the granted thread or process shall write a 1 to return the semaphore.

Table 5.9 - PTSTNSETR - Test & Set Shadow Register

5.2.7 MSC0CFG - Miscellaneous Configuration Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
31	PERI_SOFTRESET	RWAC	0	Write 1 to cause soft reset to all peripherals. It is automatically cleared.
30	PWM_SOFTRESET	RWAC	0	Write 1 to cause soft reset to PWM. It is automatically cleared.
29	I2C_SWOP	RW	0	0: I^2C master function is available at GPIO44 and GPIO45, and I^2C slave function is available at GPIO46 and GPIO47. 1: swap the I^2C master and I^2C slave GPIO positions.
28:26	PWM_TRIG_SEL	RW	0	PWM external trigger selection (see PWM_CTRL1 - PWM Control Register). One of the following GPIOs can be used to trigger the PWMs to start. If a GPIO is used for this purpose, the corresponding pin must be configured as a GPIO input and used solely for this purpose (see 5.2.8 and 5.2.9). 0: none 1: GPIO 18 2: GPIO 26 3: GPIO 35 4: GPIO 40

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
				5: GPIO 46 6: GPIO 52 7: GPIO 58
25	CAN0_SLOW	RW	0	1: Extends further the divider of CAN 0 by a factor of 16.
24	CAN1_SLOW	RW	0	1: Extends further the divider of CAN 1 by a factor of 16.
23	DEV_HIGH_SPEED	RO	0	1: Indicates USB Device is in HS mode 0: Indicates USB Device is in FS mode
22	Reserved	RO	0	
21	UART0_CLKSEL	RW	0	Clock Select for UART 0. This bit configures the internal UART clock pre-scaler. Refer to UART section 12.3 for more information.
20	UART0_FIFOSEL	RW	0	FIFO Select for UART 0. This bit is used together with several UART registers to select the UART operation mode. Refer to UART section 12.3 for more information.
19	UART0_INTSEL	RW	0	Interrupt Select for UART 0. This bit is used to control the internal UART interrupt signal to the CPU. Refer to UART section 12.3 for more information.
18	UART1_CLKSEL	RW	0	Clock Select for UART 1. This bit configures the internal UART clock pre-scaler. Refer to UART section 12.3 for more information.
17	UART1_FIFOSEL	RW	0	FIFO Select for UART 1. This bit is used together with several UART registers to select the UART operation mode. Refer to UART section 12.3 for more information.
16	UART1_INTSEL	RW	0	Interrupt Select for UART 1. This bit is used to control the internal UART interrupt signal to the CPU. Refer to UART section 12.3 for more information.
15	HOST_RESET_ALL	RWAC	0	Write 1 to cause USB Host EHCI and PHY reset; it is automatically cleared immediately. Software needs to wait for EHCI to complete its reset (~200ms).
14	HOST_RESET_EHCI	RWAC	0	Write 1 to cause USB Host EHCI reset; it is automatically cleared immediately. Software needs to wait for EHCI to complete its reset (~200ms).
13	HOST_RESET_PHY	RWAC	0	Write 1 to cause USB Host PHY reset; it is automatically cleared immediately.
12	DEV_RMWAKEUP	RW	0	1: Drive K-state on Device USB port; software must maintain the 1ms requirement before turning it off.
11	DEV_RESET_ALL	RWAC	0	Write 1 to cause USB Device Controller and PHY reset; it is automatically cleared immediately.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
10	DEV_RESET_CONTROLLER	RWAC	0	Write 1 to cause USB Device Controller reset; it is automatically cleared immediately.
9	DEV_RESET_PHY	RWAC	0	Write 1 to cause USB Device PHY reset; it is automatically cleared immediately.
8	MAC_RESET_PHY	RW	0	Write 1 to cause Ethernet PHY reset; it is automatically cleared immediately.
7:6	BCDHOST_MODE	WO	0	Battery Charging Device (BCD) Host Mode: 0: Standard Downstream Port (SDP) 1: Dedicated Charging Port (DCP) 2: Reserved 3: Charging Downstream Port (CDP)
5	BCDHOST_EN	WO	0	1: enable BCD Host
4	BCD_SOFTRESET	WO	0	1: Generate software reset to - BCD Host if BCDHOST_EN is 1. - BCD Dev if BCDDEV_EN is 1. It is automatically cleared immediately.
4	BCDDEV_DETECT_RUNNING	RO	0	1: indicates BCD Device detection is running
3	BCDDEV_EN	WO	0	1: enables BCD Device
3	BCDDEV_DETECT_COMPLETE	RO	0	1: indicates BCD Device detection is done
2	BCDDEV_SD_EN	WO	1	1: enables secondary detection
2	BCDDEV_SDP_FOUND	RO	0	1: SDP detected
1	BCDDEV_VDP_EN_POST_DCP	WO	0	1: enables connection of V_{DP_SRC} after DCP detection
1	BCDDEV_CDP_FOUND	RO	0	1: CDP detected
0	BCDDEV_LGC_COMP_INHIB	WO	1	1: disables logic comparison during BCD detections
0	BCDDEV_DCP_FOUND	RO	0	1: DCP detected

Table 5.10 - MSC0CFG - Miscellaneous Configuration Register

5.2.8 GPIO Function Registers (address offset: 0x1C – 0x5F)

The GPIOs of the FT90x MCUs can be configured to work as a different function, such as UART, SPI, I^2C etc. The GPIO Function Registers are provided for configuring the function. Each register houses the configuration for 4 GPIOs except the last register, which only configures 3 GPIOs (64 to 66). Each byte of the register configures 1 GPIO. The signal direction for each of the special functions is fixed and will be set automatically. Attributes such as Schmitt, slew rate can be adjusted by setting appropriate bits in the register. By default, all GPIOs are configured as GPIOs. A GPIO that is configured as a GPIO can be further configured. Refer to section 5.2.9.

The bit layout for the GPIO Function Registers is as follows:

Bit	Description	Value	Configuration
31:30 23:22 15:14 7:6	GPIO Function	00 01 10 11	GPIO Function Special Function 1 Special Function 2 (if available) Special Function 3 (if available)
29:28 21:20 13:12 5:4	Output drive capability	00 01 10 11	4mA 8mA 12mA 16mA

Copyright © Bridgetek Pte Ltd

27:26 19:18 11:10 3:2	With Pull-up / Pull-down	00 01 10 11	None 75kΩ Pull-down 75kΩ Pull-up 75kΩ Keeper
25 17 9 1	Schmitt	0 1	Normal Schmitt
24 16 8 0	Slew Rate	0 1	Fast Slow

Table 5.11 – GPIO Function Register Description

The following tables give more details about each GPIO Function Register. The "GPIO Function Bits" section refers to bits 31:30, 23:22, 15:14, 7:6 in each register for configuring the corresponding GPIO to perform a specific function. Refer to table 5.11 above.

5.2.8.1 GPIO 00 – 03 Function Register (address offset: 0x1C)

Note: When GPIO 3 is configured as VBUS Detect, the VBUS level can be read back as GPIO 3 value. When GPIO 3 is configured as GPIO, VBUS Detect is turned off internally. This can be used as a switch to force the VBUS detection cycle again.

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO03_FUNCTION	RW	8'h04	GPIO 3	VBUS Detect	-	-
23:16	GPIO02_FUNCTION	RW	8'h08	GPIO 2	-	-	-
15:8	GPIO01_FUNCTION	RW	8'h08	GPIO 1	OCN	-	-
7:0	GPIO00_FUNCTION	RW	8'h08	GPIO 0	-	-	-

Table 5.12 - GPIO 00 – 03 Function Register

5.2.8.2 GPIO 04 – 07 Function Register (address offset: 0x20)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO07_FUNCTION	RW	8'h04	GPIO 7	Cam PCLK	-	ADC 2
23:16	GPIO06_FUNCTION	RW	8'h04	GPIO 6	Cam Ext. CLK	RTC Ref. CLK	ADC 1
15:8	GPIO05_FUNCTION	RW	8'h04	GPIO 5	Ethernet LED 1	-	-
7:0	GPIO04_FUNCTION	RW	8'h04	GPIO 4	Ethernet LED 0	-	

Table 5.13 - GPIO 04 – 07 Function Register

5.2.8.3 GPIO 08 – 11 Function Register (address offset: 0x24)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO11_FUNCTION	RW	8'h04	GPIO 11	Cam D6	-	ADC 6
23:16	GPIO10_FUNCTION	RW	8'h04	GPIO 10	Cam D7	-	ADC 5
15:8	GPIO09_FUNCTION	RW	8'h04	GPIO 9	Cam HD	-	ADC 4
7:0	GPIO08_FUNCTION	RW	8'h04	GPIO 8	Cam VD	-	ADC 3

Table 5.14 - GPIO 08 – 11 Function Register

Copyright © Bridgetek Pte Ltd

5.2.8.4 GPIO 12 – 15 Function Register (address offset: 0x28)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO15_FUNCTION	RW	8'h04	GPIO 15	Cam D2	CAN 0 TX	-
23:16	GPIO14_FUNCTION	RW	8'h04	GPIO 14	Cam D3	-	DAC 0
15:8	GPIO13_FUNCTION	RW	8'h04	GPIO 13	Cam D4	-	DAC 1
7:0	GPIO12_FUNCTION	RW	8'h04	GPIO 12	Cam D5	-	ADC 7

Table 5.15 - GPIO 12 – 15 Function Register

5.2.8.5 GPIO 16 – 19 Function Register (address offset: 0x2C)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO19_FUNCTION	RW	8'h04	GPIO 19	SD CLK	-	-
23:16	GPIO18_FUNCTION	RW	8'h04	GPIO 18	-	CAN 1 RX	-
15:8	GPIO17_FUNCTION	RW	8'h04	GPIO 17	Cam D0	CAN 1 TX	-
7:0	GPIO16_FUNCTION	RW	8'h04	GPIO 16	Cam D1	CAN 0 RX	-

Table 5.16 - GPIO 16 – 19 Function Register

5.2.8.6 GPIO 20 – 23 Function Register (address offset: 0x30)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO23_FUNCTION	RW	8'h04	GPIO 23	SD DAT1	-	-
23:16	GPIO22_FUNCTION	RW	8'h04	GPIO 22	SD DAT2	-	-
15:8	GPIO21_FUNCTION	RW	8'h04	GPIO 21	SD DAT3	-	-
7:0	GPIO20_FUNCTION	RW	8'h04	GPIO 20	SD CMD	-	-

Table 5.17 - GPIO 20 – 23 Function Register

5.2.8.7 GPIO 24 – 27 Function Register (address offset: 0x34)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	**01**	**10**	**11**
31:24	GPIO27_FUNCTION	RW	8'h04	GPIO 27	SPIM SCK	-	-
23:16	GPIO26_FUNCTION	RW	8'h04	GPIO 26	SD WP	-	-
15:8	GPIO25_FUNCTION	RW	8'h04	GPIO 25	SD CD	-	-
7:0	GPIO24_FUNCTION	RW	8'h04	GPIO 24	SD DAT0	-	-

Table 5.18 - GPIO 24 – 27 Function Register

5.2.8.8 GPIO 28 – 31 Function Register (address offset: 0x38)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO31_FUNCTION	RW	8'h04	GPIO 31	SPIM IO2	-	-
23:16	GPIO30_FUNCTION	RW	8'h04	GPIO 30	SPIM MISO	-	-
15:8	GPIO29_FUNCTION	RW	8'h04	GPIO 29	SPIM MOSI	-	-
7:0	GPIO28_FUNCTION	RW	8'h04	GPIO 28	SPIM SS0	-	-

Table 5.19 - GPIO 28 – 31 Function Register

Copyright © Bridgetek Pte Ltd

5.2.8.9 GPIO 32 – 35 Function Register (address offset: 0x3C)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO35_FUNCTION	RW	8'h04	GPIO 35	SPIM SS3	-	-
23:16	GPIO34_FUNCTION	RW	8'h04	GPIO 34	SPIM SS2	-	-
15:8	GPIO33_FUNCTION	RW	8'h04	GPIO 33	SPIM SS1	-	-
7:0	GPIO32_FUNCTION	RW	8'h04	GPIO 32	SPIM IO3	-	-

Table 5.20 - GPIO 32 – 35 Function Register

5.2.8.10 GPIO 36 – 39 Function Register (address offset: 0x40)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO39_FUNCTION	RW	8'h04	GPIO 39	SPIS0 MISO	-	-
23:16	GPIO38_FUNCTION	RW	8'h04	GPIO 38	SPIS0 MOSI	-	-
15:8	GPIO37_FUNCTION	RW	8'h04	GPIO 37	SPIS0 SS	-	-
7:0	GPIO36_FUNCTION	RW	8'h04	GPIO 36	SPIS0 SCK	-	-

Table 5.21 - GPIO 36 – 39 Function Register

5.2.8.11 GPIO 40 – 43 Function Register (address offset: 0x44)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO43_FUNCTION	RW	8'h04	GPIO 43	SPIS1 MISO	-	-
23:16	GPIO42_FUNCTION	RW	8'h04	GPIO 42	SPIS1 MOSI	-	-
15:8	GPIO41_FUNCTION	RW	8'h04	GPIO 41	SPIS1 SS	-	-
7:0	GPIO40_FUNCTION	RW	8'h04	GPIO 40	SPIS1 SCK	-	-

Table 5.22 - GPIO 40 – 43 Function Register

5.2.8.12 GPIO 44 – 47 Function Register (address offset: 0x48)

Bit	Name	Type	Default Value	GPIO Function Bits *(note that the GPIOs for I²C Master & Slave can be swopped)*			
				00	01	10	11
31:24	GPIO47_FUNCTION	RW	8'h04	GPIO 47	I^2CS SDA	-	-
23:16	GPIO46_FUNCTION	RW	8'h04	GPIO 46	I^2CS SCL	-	-
15:8	GPIO45_FUNCTION	RW	8'h04	GPIO 45	I^2CM SDA	-	-
7:0	GPIO44_FUNCTION	RW	8'h04	GPIO 44	I^2CM SCL	-	-

Table 5.23 - GPIO 44 – 47 Function Register

Copyright © Bridgetek Pte Ltd

5.2.8.13 GPIO 48 – 51 Function Register (address offset: 0x4C)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO51_FUNCTION	RW	8'h04	GPIO 51	-	-	UART0 CTS
23:16	GPIO50_FUNCTION	RW	8'h04	GPIO 50	-	-	UART0 RTS
15:8	GPIO49_FUNCTION	RW	8'h04	GPIO 49	-	-	UART0 RXD
7:0	GPIO48_FUNCTION	RW	8'h04	GPIO 48	-	-	UART0 TXD

Table 5.24 - GPIO 48 – 51 Function Register

5.2.8.14 GPIO 52 – 55 Function Register (address offset: 0x50)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO55_FUNCTION	RW	8'h04	GPIO 55	PWM Ch7	UART1 CTS	UART0 RI
23:16	GPIO54_FUNCTION	RW	8'h04	GPIO 54	PWM Ch6	UART1 RTS	UART0 DCD
15:8	GPIO53_FUNCTION	RW	8'h04	GPIO 53	PWM Ch5	UART1 RXD	UART0 DSR
7:0	GPIO52_FUNCTION	RW	8'h04	GPIO 52	PWM Ch4	UART1 TXD	UART0 DTR

Table 5.25 - GPIO 52 – 55 Function Register

5.2.8.15 GPIO 56 – 59 Function Register (address offset: 0x54)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	Reserved	-	-	-	-	-	-
23:16	GPIO58_FUNCTION	RW	8'h04	GPIO 58	PWM Ch2	-	-
15:8	GPIO57_FUNCTION	RW	8'h04	GPIO 57	PWM Ch1	-	-
7:0	GPIO56_FUNCTION	RW	8'h04	GPIO 56	PWM Ch0	-	-

Table 5.26 - GPIO 56 – 59 Function Register

Copyright © Bridgetek Pte Ltd

5.2.8.16 GPIO 60 – 63 Function Register (address offset: 0x58)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	GPIO63_FUNCTION	RW	8'h04	GPIO 63	I²SM LRCLK	I²SS LRCLK	-
23:16	GPIO62_FUNCTION	RW	8'h04	GPIO 62	I²SM BCLK	I²SS BCLK	-
15:8	GPIO61_FUNCTION	RW	8'h04	GPIO 61	I²S SDAI	-	-
7:0	GPIO60_FUNCTION	RW	8'h04	GPIO 60	I²S SDAO	-	-

Table 5.27 - GPIO 60 – 63 Function Register

5.2.8.17 GPIO 64 – 66 Function Register (address offset: 0x5C)

Bit	Name	Type	Default Value	GPIO Function Bits			
				00	01	10	11
31:24	Reserved	RW	8'h04	-	-	-	-
23:16	GPIO66_FUNCTION	RW	8'h04	GPIO 66	I²SM CLK24	-	-
15:8	GPIO65_FUNCTION	RW	8'h04	GPIO 65	I²SM CLK22	-	-
7:0	GPIO64_FUNCTION	RW	8'h04	GPIO 64	I²S MCLK	-	-

Table 5.28 - GPIO 64 – 66 Function Register

5.2.9 GPIO Configuration Registers (address offset: 0x60 – 0x83)

These registers control the direction and interrupt settings for the GPIOs. Each register houses the configuration for 8 GPIOs except the last register, which only configures 3 GPIOs (64 to 66). Each nibble of the register configures 1 GPIO.

All GPIOs can function as an interrupt. The polarity can be either positive edge or negative edge if its interrupt capability is enabled. **If this feature is desired, the GPIO direction must be configured as input. Otherwise, unpredictable behavior may result.**

There is no de-bouncing for all GPIO's. If they are used as general inputs, and de-bouncing is needed, then software must handle this. If the GPIO is to be used as an interrupt, the external interrupt source should be glitch free.

The bit layout for the GPIO Configuration Registers is as follows:

Bit	Description	Value	Configuration
31:30 27:26 23:22 19:18 15:14 11:10 7:6 3:2	GPIO Direction	00 01 1X	Input Output OD Output
29 25 21 17 13 9 5 1	Interrupt Capable (Also see section 5.2.11, "GPIO Interrupt Enable Registers")	0 1	No Yes

Copyright © Bridgetek Pte Ltd

Bit	Description	Value	Configuration
28 24 20 16 12 8 4 0	Interrupt Edge (only if Interrupt Capable bit is 1)	0 1	Falling Edge Rising Edge

Table 5.29 - GPIO Configuration Register Description

The following tables will give more details about the bits controlling specific GPIOs using the values from Table 5.29.

5.2.9.1 GPIO 00 – 07 Configuration Register (address offset: 0x60)

Bit	Name	Type	Default Value	Description
31:28	GPIO07_CFG	RW	0	For GPIO 7
27:24	GPIO06_CFG	RW	0	For GPIO 6
23:20	GPIO05_CFG	RW	0	For GPIO 5
19:16	GPIO04_CFG	RW	0	For GPIO 4
15:12	GPIO03_CFG	RW	0	For GPIO 3
11:8	GPIO02_CFG	RW	0	For GPIO 2
7:4	GPIO01_CFG	RW	0	For GPIO 1
3:0	GPIO00_CFG	RW	0	For GPIO 0

Table 5.30 - GPIO 00 – 07 Configuration Register

5.2.9.2 GPIO 08 – 15 Configuration Register (address offset: 0x64)

Bit	Name	Type	Default Value	Description
31:28	GPIO15_CFG	RW	0	For GPIO 15
27:24	GPIO14_CFG	RW	0	For GPIO 14
23:20	GPIO13_CFG	RW	0	For GPIO 13
19:16	GPIO12_CFG	RW	0	For GPIO 12
15:12	GPIO11_CFG	RW	0	For GPIO 11
11:8	GPIO10_CFG	RW	0	For GPIO 10
7:4	GPIO09_CFG	RW	0	For GPIO 9
3:0	GPIO08_CFG	RW	0	For GPIO 8

Table 5.31 - GPIO 08 – 15 Configuration Register

5.2.9.3 GPIO 16 – 23 Configuration Register (address offset: 0x68)

Bit	Name	Type	Default Value	Description
31:28	GPIO23_CFG	RW	0	For GPIO 23
27:24	GPIO22_CFG	RW	0	For GPIO 22
23:20	GPIO21_CFG	RW	0	For GPIO 21
19:16	GPIO20_CFG	RW	0	For GPIO 20
15:12	GPIO19_CFG	RW	0	For GPIO 19
11:8	GPIO18_CFG	RW	0	For GPIO 18
7:4	GPIO17_CFG	RW	0	For GPIO 17
3:0	GPIO16_CFG	RW	0	For GPIO 16

Table 5.32 - GPIO 16 – 23 Configuration Register

5.2.9.4 GPIO 24 – 31 Configuration Register (address offset: 0x6C)

Bit	Name	Type	Default Value	Description
31:28	GPIO31_CFG	RW	0	For GPIO 31
27:24	GPIO30_CFG	RW	0	For GPIO 30

Copyright © Bridgetek Pte Ltd

23:20	GPIO29_CFG	RW	0	For GPIO 29
19:16	GPIO28_CFG	RW	0	For GPIO 28
15:12	GPIO27_CFG	RW	0	For GPIO 27
11:8	GPIO26_CFG	RW	0	For GPIO 26
7:4	GPIO25_CFG	RW	0	For GPIO 25
3:0	GPIO24_CFG	RW	0	For GPIO 24

Table 5.33 - GPIO 24 – 31 Configuration Register

5.2.9.5 GPIO 32 – 39 Configuration Register (address offset: 0x70)

Bit	Name	Type	Default Value	Description
31:28	GPIO39_CFG	RW	0	For GPIO 39
27:24	GPIO38_CFG	RW	0	For GPIO 38
23:20	GPIO37_CFG	RW	0	For GPIO 37
19:16	GPIO36_CFG	RW	0	For GPIO 36
15:12	GPIO35_CFG	RW	0	For GPIO 35
11:8	GPIO34_CFG	RW	0	For GPIO 34
7:4	GPIO33_CFG	RW	0	For GPIO 33
3:0	GPIO32_CFG	RW	0	For GPIO 32

Table 5.34 - GPIO 32 – 39 Configuration Register

Copyright © Bridgetek Pte Ltd

5.2.9.6 GPIO 40 – 47 Configuration Register (address offset: 0x74)

Bit	Name	Type	Default Value	Description
31:28	GPIO47_CFG	RW	0	For GPIO 47
27:24	GPIO46_CFG	RW	0	For GPIO 46
23:20	GPIO45_CFG	RW	0	For GPIO 45
19:16	GPIO44_CFG	RW	0	For GPIO 44
15:12	GPIO43_CFG	RW	0	For GPIO 43
11:8	GPIO42_CFG	RW	0	For GPIO 42
7:4	GPIO41_CFG	RW	0	For GPIO 41
3:0	GPIO40_CFG	RW	0	For GPIO 40

Table 5.35 - GPIO 40 – 47 Configuration Register

5.2.9.7 GPIO 48 – 55 Configuration Register (address offset: 0x78)

Bit	Name	Type	Default Value	Description
31:28	GPIO55_CFG	RW	0	For GPIO 55
27:24	GPIO54_CFG	RW	0	For GPIO 54
23:20	GPIO53_CFG	RW	0	For GPIO 53
19:16	GPIO52_CFG	RW	0	For GPIO 52
15:12	GPIO51_CFG	RW	0	For GPIO 51
11:8	GPIO50_CFG	RW	0	For GPIO 50
7:4	GPIO49_CFG	RW	0	For GPIO 49
3:0	GPIO48_CFG	RW	0	For GPIO 48

Table 5.36 - GPIO 48 – 55 Configuration Register

5.2.9.8 GPIO 56 – 63 Configuration Register (address offset: 0x7C)

Bit	Name	Type	Default Value	Description
31:28	GPIO63_CFG	RW	0	For GPIO 63
27:24	GPIO62_CFG	RW	0	For GPIO 62
23:20	GPIO61_CFG	RW	0	For GPIO 61
19:16	GPIO60_CFG	RW	0	For GPIO 60
15:12	Reserved	-	-	-
11:8	GPIO58_CFG	RW	0	For GPIO 58
7:4	GPIO57_CFG	RW	0	For GPIO 57
3:0	GPIO56_CFG	RW	0	For GPIO 56

Table 5.37 - GPIO 56 – 63 Configuration Register

5.2.9.9 GPIO 64 – 66 Configuration Register (address offset: 0x80)

Bit	Name	Type	Default Value	Description
31:12	Reserved	-	-	-
11:8	GPIO66_CFG	RW	0	For GPIO 66
7:4	GPIO65_CFG	RW	0	For GPIO 65
3:0	GPIO64_CFG	RW	0	For GPIO 64

Table 5.38 - GPIO 64 – 66 Configuration Register

5.2.10 GPIO Value Registers (address offset: 0x84 – 0x8F)

These registers contain the values for the GPIOs. Each register contains the value of 32 GPIOs except the last register, which only contains the value of 3 GPIOs (64 to 66). Each bit of the register maps to the corresponding digital pin.

Copyright © Bridgetek Pte Ltd

5.2.10.1 GPIO 00 – 31 Value Register (address offset: 0x84)

Bit	Name	Type	Default Value	Description
31:0	GPIO_VAL_IN[31:0]	RO	X	Input values of GPIO 31 – 0
31:0	GPIO_VAL_OUT[31:0]	WO	0	Output values of GPIO 31 – 0

Table 5.39 - GPIO 00 – 31 Value Register

5.2.10.2 GPIO 32 – 63 Value Register (address offset: 0x88)

Bit	Name	Type	Default Value	Description
31:0	GPIO_VAL_IN[63:32]	RO	X	Input values of GPIO 63 – 32
31:0	GPIO_VAL_OUT[63:32]	WO	0	Output values of GPIO 63 - 32

Table 5.40 - GPIO 32 – 63 Value Register

5.2.10.3 GPIO 64 – 66 Value Register (address offset: 0x8C)

Bit	Name	Type	Default Value	Description
31:3	Reserved	-	-	-
2:0	GPIO_VAL_IN[66:64]	RO	X	Input values of GPIO 66 – 64
2:0	GPIO_VAL_OUT[66:64]	WO	0	Output values of GPIO 66 - 64

Table 5.41 - GPIO 64 – 66 Value Register

5.2.11 GPIO Interrupt Enable Registers (address offset: 0x90 – 0x9B)

When a GPIO has been configured as an input with interrupt capability, the GPIO Interrupt Enable Register can be used to enable interrupt generation. Each register enables interrupt generation for 32 GPIOs except the last register, which only enables interrupt generation for 3 GPIOs (64 to 66). Each bit of the register enables interrupt generation for 1 GPIO.

These should be used only for pads that have been properly configured as interrupt enabled GPIO inputs.

5.2.11.1 GPIO 00 – 31 Interrupt Enable Register (address offset: 0x90)

Bit	Name	Type	Default Value	Description
31:0	GPIO_INT_EN[31:0]	RW	0	GPIO 31-0 interrupt enable when set.

Table 5.42 - GPIO 00 – 31 Interrupt Enable Register

5.2.11.2 GPIO 32 – 63 Interrupt Enable Register (address offset: 0x94)

Bit	Name	Type	Default Value	Description
31:0	GPIO_INT_EN[63:32]	RW	0	GPIO 63-32 interrupt enable when set.

Table 5.43 - GPIO 32 – 63 Interrupt Enable Register

5.2.11.3 GPIO 64 – 66 Interrupt Enable Register (address offset: 0x98)

Bit	Name	Type	Default Value	Description
31:3	Reserved	-	-	-
2:0	GPIO_INT_EN[66:64]	RW	0	GPIO 66-64 interrupt enable when set.

Table 5.44 - GPIO 64 – 66 Interrupt Enable Register

5.2.12 Interrupt Pending Registers (address offset: 0x9C – 0xA7)

These registers hold the interrupt pending flags for the GPIOs. Each register holds the flags for 32 GPIOs except the last register, which only holds the flags for 3 GPIOs (64 to 66). Each bit of the register holds the flag for 1 GPIO.

Copyright © Bridgetek Pte Ltd

5.2.12.1 GPIO 00 – 31 Interrupt Pending Register (address offset: 0x9C)

Bit	Name	Type	Default Value	Description
31:0	GPIO_INT_PEND[31:0]	RW1C	0	GPIO 31-0 interrupt pending when set; write a 1 to clear.

Table 5.45 - GPIO 00 – 31 Interrupt Pending Register

5.2.12.2 GPIO 32 – 63 Interrupt Pending Register (address offset: 0xA0)

Bit	Name	Type	Default Value	Description
31:0	GPIO_INT_PEND[63:32]	RW1C	0	GPIO 63-32 interrupt pending when set; write a 1 to clear.

Table 5.46 - GPIO 32 – 63 Interrupt Pending Register

5.2.12.3 GPIO 64 – 66 Interrupt Pending Register (address offset: 0xA4)

Bit	Name	Type	Default Value	Description
31:3	Reserved	-	-	-
2:0	GPIO_INT_PEND[66:64]	RW1C	0	GPIO 66-64 interrupt pending when set; write a 1 to clear.

Table 5.47 - GPIO 64 – 66 Interrupt Pending Register

5.2.13 ETH_PHY_CFG - Ethernet PHY Miscellaneous Configuration Register (address offset: 0xA8)

Bit	Name	Type	Default Value	Description
31	LINK_DONE	RO	0	Link status: 0: Link is down 1: Link is up
30	Reserved	-	-	-
29	LINK_TRAFFIC_LED	RO	X	Link/Traffic LED Link State: 0: indicates a good link status for 10/100M. Traffic State: The traffic status is always ON (LOW) when the link is OK and toggles between the TX and RX state.
28	TX_ACT	RO	X	Active LOW, indicates the presence of an active transmission.
27	RX_ACT	RO	X	Active LOW, indicates the presence of an active reception.
26	COLLISION_LED	RO	X	Active LOW, indicates the presence of a collision.
25	FULL_DUPLEX_LED	RO	X	Active LOW, indicates the status of the full-duplex mode
24	LINK_SPEED	RO	-	Active LOW, indicates the link speed is 100 Mbps
23	Reserved	-	-	-
22:20	ETHERNET_LED1_SEL	RW	0	Ethernet LED 1 source selection: 0: LINKLED

Copyright © Bridgetek Pte Ltd

				1: TXLED 2: RXLED 3: COLLED 4: FDXLED Others: SPDLED
19	Reserved	-	-	-
18:16	ETHERNET_LED2_SEL	RW	3'h7	Ethernet LED 2 source selection: 0: LINKLED 1: TXLED 2: RXLED 3: COLLED 4: FDXLED Others: SPDLED
15:10	Reserved	-	-	-
9	ETHERNET_PWRSV	RW	1	1: Enable Ethernet PHY Power Save (Preferably Set this to 0 prior to enabling the MAC clock)
8	ETHERNET_PWRDN	RW	1	1: Ethernet PHY Power Down (Preferably clear this bit prior to enabling the MAC clock)
7:5	Reserved	-	-	-
4:0	ETHERNET_PHYAD	RW	0	Ethernet PHY Address. After writing the PHY address, the PHY should be reset via MAC_RESET_PHY. After that, the new PHY address will be reflected in ETH_MNG_ADDR.

Table 5.48 - ETH_PHY_CFG - Ethernet PHY Miscellaneous Configuration Register

5.2.14 ETH_PHY_ID - Ethernet PHY ID Register (address offset: 0xAC)

Bit	Name	Type	Default Value	Description
31:0	ETHERNET_PHYID	RW	0	Ethernet PHY ID

Table 5.49 - ETH_PHY_ID - Ethernet PHY ID Register

5.2.15 DAC_ADC_CONF - ADC/DAC Configuration/Status Register (address offset: 0xB0)

Bit	Name	Type	Default Value	Description
31	Reserved	-	-	-
30	ADC_BUF_ERROR	RO	0	1: Indicates ADC buffer error. See Chapter 23.
29	DAC1_BUF_ERROR	RO	0	1: Indicates DAC 1 buffer error. See Chapter 23.
28	DAC0_BUF_ERROR	RO	0	1: Indicates DAC 0 buffer error. See Chapter 23.
27	Reserved	-	-	-
26	ADC_IRQ_PEND	RW1C	0	ADC Interrupt Pending when set. Write a 1 to clear this bit.
25	DAC1_IRQ_PEND	RW1C	0	DAC 1 Interrupt Pending when set. Write a 1 to clear this bit.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
24	DAC0_IRQ_PEND	RW1C	0	DAC 0 Interrupt Pending when set. Write a 1 to clear this bit.
23	ADC_8BIT_MODE	RW	0	0: 10-bit mode 1: 8-bit mode
22	ADC_HALF_RATE	RW	0	0: ADC clock is 12.5 MHz 1: ADC clock is 6.25 MHz
21:19	Reserved	-	-	-
18	ADC_IRQ_EN	RW	0	1: Enable ADC interrupt
17	DAC1_IRQ_EN	RW	0	1: Enable DAC 1 interrupt
16	DAC0_IRQ_EN	RW	0	1: Enable DAC 0 interrupt
15	ADC_START	RWAC	0	Write 1 to start ADC operations. This bit will be automatically cleared if ADC_CONT is 0; otherwise, if ADC_CONT, ADC operation runs till this is cleared by software.
14	Reserved	-	-	-
13	ADC_RAIL_RAIL_REF	RW	0	1: Enable Rail-Rail Reference. The range is 0 - 3.3V. 0: Disable Rail-Rail Reference. The range is 0.33 - 2.97V.
12	ADC_CONT	RW	0	1: Enable ADC continuous mode; ADC samples are available in the ADC FIFO after conversion.
11	ADC_PDB	RW	0	0: Power down ADC. Set to 0 if ADC Is not used in the chip configuration.
10:8	ADC_CHANNEL	RW	0	0: No channel is selected 1: Channel 1 selected 2: Channel 2 selected 3: Channel 3 selected 4: Channel 4 selected 5: Channel 5 selected 6: Channel 6 selected 7: Channel 7 selected
7	DAC1_START	RW	0	Write 1 to start DAC 1 conversion. When DAC1_CONT is not set, this bit is automatically cleared after a single conversion. When DAC1_CONT is set, conversion proceeds (samples are taken from FIFO) until this bit is cleared by software.
6	Reserved	-	-	-
5	DAC1_CONT	RW	0	1: Enable DAC 1 continuous conversion mode; DAC samples are read from the DAC FIFO
4	DAC1_PDB	RW	0	0: Power down DAC 1. Set to 0 if DAC 1 is not used in the chip configuration.
3	DAC0_START	RWAC	0	Write 1 to start DAC 0 conversion. When DAC0_CONT is not set, this bit is automatically cleared after a single conversion.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
				When DAC0_CONT is set, conversion proceeds (samples are taken from FIFO) until this bit is cleared by software.
2	Reserved	-	-	-
1	DAC0_CONT	RW	0	1: Enable DAC0 in continuous conversion mode; DAC samples are read from the DAC FIFO
0	DAC0_PDB	RW	0	0: Power down DAC 0. Set to 0 if DAC 0 is not used in the chip configuration.

Table 5.50 - DAC_ADC_CONF - ADC/DAC Configuration/Status Register

5.2.16 DAC_ADC_CNT - ADC/DAC Count Register (address offset: 0xB4)

Bit	Name	Type	Default Value	Description
31	Reserved	-	-	-
30:24	DAC_DIVIDER	RW	7'h63	This determines the DAC1/0 conversion rate. The rate is determined by System clock / (DAC_DIVIDER+1) where system clock is 100 MHz. The maximum conversion rate is 1MHz.
23:16	ADC_DATA_COUNT	RO	8'hFF	The number of samples available for reading in the ADC FIFO at the most recent interrupts. It is equal to ADC_DATA_COUNT plus one. ADC_DATA_COUNT / Samples in FIFO / Note 0 / 1 / 1 / 2 / … / … / 127 / 128 / Full 255 / 0 / Empty Note that more conversions may have been performed since the last interrupt, so the FIFO contains at least (ADC_DATA_COUNT+1) samples for readout.
15:8	DAC1_DATA_COUNT	RO	8'hFF	The number of samples still available for conversion in the DAC 1 FIFO. It is equal to DAC1_DATA_COUNT plus one. DAC1_DATA_COUNT / Samples in FIFO / Note 0 / 1 / 1 / 2 / … / … / 127 / 128 / Full 255 / 0 / Empty

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
7:0	DAC0_DATA_COUNT	RO	8'hFF	The number of samples still available for conversion in the DAC 0 FIFO. It is equal to DAC0_DATA_COUNT plus one. DAC0_DATA_COUNT / Samples in FIFO / Note 0 / 1 / 1 / 2 / … / … / 127 / 128 / Full 255 / 0 / Empty

Table 5.51 - DAC_ADC_CNT - ADC/DAC Count Register

5.2.17 DAC_ADC_DATA - ADC/DAC Data Register (address offset: 0xB8)

Bit	Name	Type	Default Value	Description
31:26	Reserved	RO	0	-
25:16	DAC1_DATA	WO	0	Write window for DAC 1 FIFO; If Byte access is used, write to the FIFO occurs only when the high byte is written. Hence the upper bits should be written last in this case.
15:10	Reserved	RO	0	
9:0	ADC_DATA	RO	0	Read window for ADC FIFO. Note that the data is read per sample. As a result, in 10-bit mode (when ADC_8BIT_MODE is cleared), this register should be read with Word (16-bit) or Double-Word (32-bit) access. All unused upper bits will be zeros.
9:0	DAC0_DATA	WO	0	Write window for DAC 0 FIFO; If Byte access is used, write to the FIFO occurs only when the high byte is written. Hence the upper bits should be written last in this case.

Table 5.52 - DAC_ADC_DATA - ADC/DAC Data Register

5.2.18 RESET_STATUS – Global Reset/Status Register (address offset: 0xBC)

Bit	Name	Type	Default Value	Description
31:2	Reserved	-	-	-
1	SOFT_RESET	RW	0	Soft Global Reset Write "1" to reset the system to POR state. Write "0" to clear this bit. This bit must be cleared before it can be enabled again. When "1" is read from this bit, it indicates the soft global reset has been performed from the previous operation.
0	HW_RESET	RW1C	0	External Hardware Reset If it is set, it indicates the external hardware reset has been performed. Write "1" to clear it.

Table 5.53 - RESET_STATUS – Global Reset/Status Register

Copyright © Bridgetek Pte Ltd

6 Interrupt Controller

The interrupt controller takes in 32 interrupts, and based on the interrupt priorities assigned, it generates the interrupt to the FT90X together with an ISR address. Nested interrupts are allowed if enabled. By default, it is disabled. Up to 16 levels of nesting is allowed which defaults to only 1 level if nesting is enabled.

When nesting is enabled, only interrupts with higher priorities can interrupt the current interrupt. Interrupts of same or lower priorities will be queued as long as the interrupt sources are not cleared.

The interrupt vector table (IVT) resides in program memory addresses 8 to 135. It contains thirty-two 32-bit entries corresponding to interrupt priorities 0 to 31, with a lower number indicating a higher priority. While in principle, an extremely short interrupt handler could be stored entirely inside the IVT, in practice, each entry should contain "JMP address", where "address" is the address of the interrupt service routine (ISR) for the corresponding interrupt.

Each peripheral is assigned a fixed interrupt number, for example USB Host is assigned interrupt number 1, and USB Device is assigned interrupt number 2. Refer the table 6.1 below for the complete list of the interrupt numbers. By default, interrupts 0 to 31 are assigned to entries 0 to 31 in the IVR respectively. It is possible to assign an interrupt to another entry in the IVT, thus changing its priority, by setting the appropriate registers in section 6.2. It is possible to assign multiple interrupts to the same entry/priority, in which case the ISR may need to check the appropriate interrupt flags to differentiate the interrupt source.

Note that after the priority of an interrupt is changed, the address of the ISR should be updated in the corresponding IVT entry.

A global interrupt mask bit is also available. Setting it to 1 will temporarily block all interrupts except the interrupt assigned as interrupt 0 which is non-maskable by this global mask.

In the FT90X, the interrupts connections from the peripherals are listed in the table below. Interrupts 23 to 31 are unused by default.

Peripheral	Interrupt Number	Default Priority
Power Management	0	0 (Highest) – Non-maskable
USB Host	1	1
USB Device	2	2
Ethernet	3	3
SD Host	4	4
CAN 0	5	5
CAN 1	6	6
Data Capture Interface	7	7
SPI Master	8	8
SPI Slave 0	9	9
SPI Slave 1	10	10
I2C Master	11	11
I2C Slave	12	12
UART 0	13	13
UART 1	14	14
I2S	15	15
PWM	16	16
TIMERS	17	17
GPIO	18	18
RTC	19	19
ADC	20	20
DAC	21	21
SLOWCLOCK Timer	22	22
UNUSED	**23-31**	**23-31**

Table 6.1 – Interrupt Assignment Table

Copyright © Bridgetek Pte Ltd

6.1 Register Summary

The base address for the interrupt assignment registers is 0x100C0. All registers and RAM locations can be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit) mode.

Address Offset	Register	Default value	References
0x00	Interrupt 00-03 Priority Assignment Register	0x0302 0100	Section 6.2.1
0x04	Interrupt 04-07 Priority Assignment Register	0x0706 0504	Section 6.2.2
0x08	Interrupt 08-11 Priority Assignment Register	0x0B0A 0908	Section 6.2.3
0x0C	Interrupt 12-15 Priority Assignment Register	0x0F0E 0D0C	Section 6.2.4
0x10	Interrupt 16-19 Priority Assignment Register	0x1312 1110	Section 6.2.5
0x14	Interrupt 20-23 Priority Assignment Register	0x1716 1514	Section 6.2.6
0x18	Interrupt 24-27 Priority Assignment Register	0x1B1A 1918	Section 6.2.7
0x1C	Interrupt 28-31 Priority Assignment Register	0x1F1E 1D1C	Section 6.2.8
0x20	Interrupt Control Register	0x8000 0000	Section 6.2.9

Table 6.2 – Overview of Interrupt Control Registers

6.2 Register Details

6.2.1 Interrupt 00-03 Priority Assignment Register (address offset: 0x00)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR03ASSIGN	RW	5'h03	Priority assignment for interrupt 3
23:21	Reserved	-	-	-
20:16	PR02ASSIGN	RW	5'h02	Priority assignment for interrupt 2
15:13	Reserved	-	-	-
12:8	PR01ASSIGN	RW	5'h01	Priority assignment for interrupt 1
7:5	Reserved	-	-	-
4:0	PR00ASSIGN	RW	5'h00	Priority assignment for interrupt 0

Table 6.3 – Interrupt 00-03 Priority Assignment Register

6.2.2 Interrupt 04-07 Priority Assignment Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR07ASSIGN	RW	5'h07	Priority assignment for interrupt 7
23:21	Reserved	-	-	-
20:16	PR06ASSIGN	RW	5'h06	Priority assignment for interrupt 6
15:13	Reserved	-	-	-
12:8	PR05ASSIGN	RW	5'h05	Priority assignment for interrupt 5
7:5	Reserved	-	-	-
4:0	PR04ASSIGN	RW	5'h04	Priority assignment for interrupt 4

Table 6.4 – Interrupt 04-07 Priority Assignment Register

Copyright © Bridgetek Pte Ltd

6.2.3 Interrupt 08-11 Priority Assignment Register (address offset: 0x08)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR11ASSIGN	RW	5'h0B	Priority assignment for interrupt 11
23:21	Reserved	-	-	-
20:16	PR10ASSIGN	RW	5'h0A	Priority assignment for interrupt 10
15:13	Reserved	-	-	-
12:8	PR09ASSIGN	RW	5'h09	Priority assignment for interrupt 9
7:5	Reserved	-	-	-
4:0	PR08ASSIGN	RW	5'h08	Priority assignment for interrupt 8

Table 6.5 – Interrupt 08-11 Priority Assignment Register

6.2.4 Interrupt 12-15 Priority Assignment Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR15ASSIGN	RW	5'h0F	Priority assignment for interrupt 15
23:21	Reserved	-	-	-
20:16	PR14ASSIGN	RW	5'h0E	Priority assignment for interrupt 14
15:13	Reserved	-	-	-
12:8	PR13ASSIGN	RW	5'h0D	Priority assignment for interrupt 13
7:5	Reserved	-	-	-
4:0	PR12ASSIGN	RW	5'h0C	Priority assignment for interrupt 12

Table 6.6 – Interrupt 12-15 Priority Assignment Register

6.2.5 Interrupt 16-19 Priority Assignment Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR19ASSIGN	RW	5'h13	Priority assignment for interrupt 19
23:21	Reserved	-	-	-
20:16	PR18ASSIGN	RW	5'h12	Priority assignment for interrupt 18
15:13	Reserved	-	-	-
12:8	PR17ASSIGN	RW	5'h11	Priority assignment for interrupt 17
7:5	Reserved	-	-	-
4:0	PR16ASSIGN	RW	5'h10	Priority assignment for interrupt 16

Table 6.7 – Interrupt 16-19 Priority Assignment Register

6.2.6 Interrupt 20-23 Priority Assignment Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR23ASSIGN	RW	5'h17	Priority assignment for interrupt 23
23:21	Reserved	-	-	-
20:16	PR22ASSIGN	RW	5'h16	Priority assignment for interrupt 22
15:13	Reserved	-	-	-
12:8	PR21ASSIGN	RW	5'h15	Priority assignment for interrupt 21
7:5	Reserved	-	-	-
4:0	PR20ASSIGN	RW	5'h14	Priority assignment for interrupt 20

Table 6.8 – Interrupt 20-23 Priority Assignment Register

Copyright © Bridgetek Pte Ltd

6.2.7 Interrupt 24-27 Priority Assignment Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR27ASSIGN	RW	5'h1B	Priority assignment for interrupt 27
23:21	Reserved	-	-	-
20:16	PR26ASSIGN	RW	5'h1A	Priority assignment for interrupt 26
15:13	Reserved	-	-	-
12:8	PR25ASSIGN	RW	5'h19	Priority assignment for interrupt 25
7:5	Reserved	-	-	-
4:0	PR24ASSIGN	RW	5'h18	Priority assignment for interrupt 24

Table 6.9 – Interrupt 24-27 Priority Assignment Register

6.2.8 Interrupt 28-31 Priority Assignment Register (address offset: 0x1C)

Bit	Name	Type	Default Value	Description
31:29	Reserved	-	-	-
28:24	PR31ASSIGN	RW	5'h1F	Priority assignment for interrupt 31
23:21	Reserved	-	-	-
20:16	PR30ASSIGN	RW	5'h1E	Priority assignment for interrupt 30
15:13	Reserved	-	-	-
12:8	PR29ASSIGN	RW	5'h1D	Priority assignment for interrupt 29
7:5	Reserved	-	-	-
4:0	PR28ASSIGN	RW	5'h1C	Priority assignment for interrupt 28

Table 6.10 – Interrupt 28-31 Priority Assignment Register

6.2.9 Interrupt Control Register (address offset: 0x20)

Bit	Name	Type	Default Value	Description
31	Global Interrupt Mask	RW	1	Set to 1 to mask all interrupts.
30:8	Reserved	-	-	-
7	Nested Interrupt	RW	0	Set to 1 to enable nested interrupts. This bit shall only be set or cleared while the Global Interrupt Mask is enabled.
6:4	Reserved	-	-	-
3:0	Nested Depth	RW	4'h00	Maximum number of nested interrupts permitted (Nested Depth + 1); Minimum 1 level, and maximum 16 levels.

Table 6.11 – Interrupt Control Register

Copyright © Bridgetek Pte Ltd

7 USB Host

This is a single-port USB host controller which is compliant with the USB 2.0 specification and compatible with the Enhanced Host Controller Interface (EHCI) specification. It supports HS/FS/LS transactions, control/bulk/interrupt/isochronous transfers and split-transaction of the hub. An 8-kB RAM arranged as (2 kB x 32) is attached to the host as buffers.

7.1 Register Summary

Listed below are the registers with their offset from the base address (0x10100). All registers and RAM locations can be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit) mode.

Address Offset	Register	Default Value	References
	EHCI Operational Registers		
0x00	HC Capability Register	0x0100 0010	Section 7.2.1
0x04	HCSPARAMS – HC Structural Parameters	0x0000 0001	Section 7.2.2
0x08	HCCPARAMS – HC Capability Parameters	0x0000 0006	Section 7.2.3
0x10	USBCMD – HC USB Command Register	0x0008 0B00	Section 7.2.4
0x14	USBSTS – HC USB Status Register	0x0000 1000	Section 7.2.5
0x18	USBINTR – HC USB Interrupt Enable Register	0x0000 0000	Section 7.2.6
0x1C	FRINDEX – HC Frame Index Register	0x0000 0000	Section 7.2.7
0x24	PERIODICLISTBASE – HC Periodic Frame List Base Address Register	0x0000 0000	Section 7.2.8
0x28	ASYNCLISTADDR – HC Current Asynchronous List Address Register	0x0000 0000	Section 7.2.9
0x30	PORTSC – HC Port Status and Control Register	0x0000 0000	Section 7.2.10
	Configuration Registers		
0x34	EOF Time & Asynchronous Schedule Sleep Timer Register	0x0000 0041	Section 7.3.1
0x40	Bus Monitor Control / Status Register	0x0000 0000	Section 7.3.2
0x78	HPROT – Master Protection Information Setting Register	0x0000 0003	Section 7.3.3
	USB Testing Registers		
0x54	Vendor Specific IO Control Register	0x0000 0020	Section 7.4.1
0x58	Vendor Specific Status Register	0xXXXX XXXX	Section 7.4.2
0x50	Test Register	0x0000 0000	Section 7.4.3

Table 7.1 – Overview of USB Host Controller Registers

Copyright © Bridgetek Pte Ltd

7.2 EHCI Operational Registers

7.2.1 HC Capability Register (address offset: 0x00)

This register provides information about the host controller interface specification number implemented in this host controller.

Bit	Name	Type	Default Value	Description
31:16	HCIVERSION	RO	16'h0100	Host Controller Interface Version Number It is a 2-byte register containing a BCD encoding of the EHCI revision number supported by the host controller.
15:8	Reserved	-	-	-
7:0	CAPLENGTH	RO	8'h10	Capability Register Length It is used as an offset added to the register base to find out the beginning of the Operational Register Space.

Table 7.2 – HC Capability Register

7.2.2 HCSPARAMS – HC Structural Parameters (address offset: 0x04)

This register specifies the number of downstream ports implemented in this host controller.

Bit	Name	Type	Default Value	Description
31:4	Reserved	-	-	-
3:0	N_PORTS	RO	4'h1	Number of Ports This specifies the number of the physical downstream ports implemented on the host controller

Table 7.3 – HCSPARAMS – HC Structural Parameters

7.2.3 HCCPARAMS – HC Capability Parameters (address offset: 0x08)

Bit	Name	Type	Default Value	Description
31:3	Reserved	-	-	-
2	ASYN_SCH_PARK_CAP	RO	1'b1	Asynchronous Schedule Park Capability The host controller supports the park feature for HS queue heads in the Asynchronous Schedule. This feature can be disabled or enabled and set to a specific level by using the *Asynchronous Schedule Park Mode Enable* and *Asynchronous Schedule Park Mode Count* fields in the USBCMD register.
1	PROG_FR_LIST_FLAG	RO	1'b1	Programmable Frame List Flag When this bit is set to 1, the system software can specify and use a smaller frame list and configure the host controller via Frame List Size field of the USBCMD register. This requirement ensures that the frame list is always physically contiguous.
0	Reserved	-	-	-

Table 7.4 – HCCPARAMS – HC Capability Parameters

7.2.4 USBCMD – HC USB Command Register (address offset: 0x10)

The command register is used by the software to schedule the command to be executed by the USB host controller hardware.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
31:24	Reserved	-	-	-
23:16	INT_THRC	RW	8'h08	Interrupt Threshold Control. This is used by the system software to select the maximum rate at which the host controller will issue the interrupts. The only valid values are as below: 0x00: Reserved 0x01: 1 micro-frame (125us) 0x02: 2 micro-frames (250us) 0x04: 4 micro-frames (500us) 0x08: 8 micro-frames (default: 1ms) 0x10: 16 micro-frames (2ms) 0x20: 32 micro-frames (4ms) 0x40: 64 micro-frames (8ms) **Note:** In the FS mode, these bits are reserved.
15:12	Reserved	-	-	-
11	ASYN_PK_EN	RW	1'b1	Asynchronous Schedule Park Mode Enable. Software uses this to enable or disable the Park mode. When this is set to 1, the Park mode is enabled.
10	Reserved	-	-	-
9:8	ASYN_PK_CNT	RW	2'h3	Asynchronous Schedule Park Mode Count. This contains a count for the number of successive transactions that the host controller is allowed to execute from a high-speed queue head on the asynchronous schedule.
7	Reserved	-	-	-
6	INT_OAAD	RW	1'b0	Interrupt on Asynchronous Advance Doorbell. This is used as a doorbell by software to ring the host controller to issue and interrupt at the next advance of the synchronous schedule.
5	ASCH_EN	RW	1'b0	Asynchronous Schedule Enable. This controls whether the host controller skips the processing of asynchronous schedule. 0: Do not process the asynchronous schedule 1: Use the ASYNCLISTADDR register to access the asynchronous schedule
4	PSCH_EN	RW	1'b0	Periodic Schedule Enable. This controls whether the host controller skips the processing of the period schedule. 0: Do not process the period schedule 1: Use the PERIODICLISTBASE register to access the period schedule
3:2	FRL_SIZE	RW	2'h0	Frame List Size This specifies the size of the frame list. 00: 1024 elements (default value, 4096 bytes) 01: 512 elements (2048 bytes) 10: 256 elements (1024 bytes) 11: reserved
1	HC_RESET	RW	1'b0	Host Controller Reset This is used by the software to reset the host controller.
0	RS	RW	1'b0	Run/Stop When this is set to 1, the host controller proceeds with the execution of schedule. 0: Stop 1: Run

Table 7.5 – USBCMD – HC USB Command Register

Copyright © Bridgetek Pte Ltd

7.2.5 USBSTS – HC USB Status Register (address offset: 0x14)

This register indicates the status of the USB host controller. This register is updated by the USB host controller hardware. Software clears a bit by writing 1 to the bit.

Bit	Name	Type	Default Value	Description
31:16	Reserved	-	-	-
15	ASCH_STS	RO	1'b0	Asynchronous Schedule Status This reports the actual status of the asynchronous schedule.
14	PSCH_STS	RO	1'b0	Periodic Schedule Status This reports the actual status of the periodic schedule.
13	Reclamation	RO	1'b0	Reclamation This is a read-only status bit, and used to detect an empty of the asynchronous schedule
12	HCHalted	RO	1'b1	Host Controller Halted This is a 0 whenever the Run/Stop bit is set to 1. The host controller sets this to 1 after it has stopped the section as a result of the Run/Stop bit being set to 0.
11:6	Reserved	-	-	-
5	INT_OAA	RW1C	1'b0	Interrupt on Async Advance This bit indicates the assertion of interrupt on *Async Advance Doorbell*
4	H_SYSERR	RW1C	1'b0	Host System Error The host controller sets this to 1 when a serious error occurs during a host system access involving the host controller module.
3	FRL_ROL	RW1C	1'b0	Frame List Rollover The host controller sets this to 1 when the *Frame List Index* rolls over from its maximum value to zero.
2	PO_CHG_DET	RW1C	1'b0	Port Change Detect The host controller sets this to 1 when any port has a change bit transition from 0 to 1. In addition, this bit is loaded with the logical OR of all of the PORTSC change bits.
1	USBERR_INT	RW1C	1'b0	USB Error Interrupt The host controller sets this to 1 when the completion of a USB transaction results in an error condition
0	USB_INT	RW1C	1'b0	USB Interrupt The host controller sets this to 1 upon completion of a USB transaction

Table 7.6 – USBSTS – HC USB Status Register

7.2.6 USBINTR – HC USB Interrupt Enable Register (address offset: 0x18)

This register enables the required host controller interrupts. The interrupts enabled by this register toggle the interrupt pin when the interrupt condition occurs. Interrupts that are not in this register do not toggle the interrupt but the statuses of these bits can be read by polling the interrupt status register.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
31:6	Reserved	-	-	-
5	INT_OAA_EN	RW	1'b0	Interrupt on Async Advance Enable When this is 1 and the *Interrupt on Async Advance* bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.
4	H_SYSERR_EN	RW	1'b0	Host System Error Enable When this is 1 and the *Host System Error Status* bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.
3	FRL_ROL_EN	RW	1'b0	Frame List Rollover Enable When this is 1 and the *Frame List Rollover* bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.
2	PO_CHG_INT_EN	RW	1'b0	Port Change Interrupt Enable When this is 1 and the *Port Change Detect* bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.
1	USBERR_INT_EN	RW	1'b0	USB Error Interrupt Enable When this is 1 and the *USBERRINT* bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.
0	USB_INT_EN	RW	1'b0	USB Interrupt Enable When this is 1 and the *Host USBINT* bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.

Table 7.7 – USBINTR – HC USB Interrupt Enable Register

7.2.7 FRINDEX – HC Frame Index Register (address offset: 0x1C)

This register is used by the host controller to index the periodic frame. The register is updated every 125 microseconds.

Bit	Name	Type	Default Value	Description
31:14	Reserved	-	-	-
13:0	FRINDEX	RW	14'h0000	Frame Index This is used by the host controller to index the frame into the Periodic Frame List. It is updated every 125us. It cannot be written unless the host controller is halted.

Table 7.8 – FRINDEX – HC Frame Index Register

7.2.8 PERIODICLISTBASE – HC Periodic Frame List Base Address Register (address offset: 0x24)

This register contains the beginning address of the periodic frame list in the system memory.

Bit	Name	Type	Default Value	Description
31:12	PERI_BASE_ADR	RW	Undefined	Periodic Frame List Base Address This 32-bit register contains the start address of the Periodic Frame List in the system memory. These form the upper 20 bits of the address.
11:0	Reserved	-	-	-

Table 7.9 – PERIODICLISTBASE – HC Periodic Frame List Base Address Register

Copyright © Bridgetek Pte Ltd

7.2.9 ASYNCLISTADDR – HC Current Asynchronous List Address Register (address offset: 0x28)

This register contains the address of the next asynchronous queue head to be executed.

Bit	Name	Type	Default Value	Description
31:5	ASYNC_LADR	RW	Undefined	Current Asynchronous List Address This 32-bit register contains the address of the next asynchronous queue head to be executed. These form the upper 27 bits of the address.
4:0	Reserved	-	-	-

Table 7.10 – ASYNCLISTADDR – HC Current Asynchronous List Address Register

7.2.10 PORTSC – HC Port Status and Control Register (address offset: 0x30)

This register is reset only by hardware when the power is initially applied, or in response to a host controller reset.

Bit	Name	Type	Default Value	Description
31:17	Reserved	-	-	-
16	TST_FORCEEN	RW	1'b0	Test Force Enable When this is 1, the downstream facing port will be enabled in the high-speed mode. Then the Run/Stop bit must be transitioned to 1 in order to enable the transmission of the SOFs out of the port under test. This enables testing of the disconnect detection.
15:12	Reserved	-	-	-
11:10	LINE_STS	RO	Undefined	Line Status These reflect the current logical levels of the D+ and D- signal lines.
9	Reserved	-	-	-
8	PO_RESET	RW	1'b0	Port Reset 1: Port is in the reset state 0: Port is not in the reset state When the software writes a 1 to this bit, the bus reset sequence as defined in the USB specification will start. Software writes a 0 to this bit to terminate the bus reset sequence. Software must keep this bit at 1 long enough to ensure the reset sequence is completed. **Note:** Before setting this bit, Run/Stop bit should be set to 0.
7	PO_SUSP	RW	1'b0	Port Suspend 1: Port is in the suspend state 0: Port is not in the suspend state The Port Enable Bit and Port Suspend Bit of this register define the port state as follows. **Enable/Suspend (Bit)** / **Port State** 0/X – Disabled 1/0 – Enabled 1/1 – Suspended At the suspended status, the downstream propagation of data is blocked on this port except for the port reset. While at the suspended state, the port is sensitive to resume detection. Writing a 0 to this bit is ignored. The host controller

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
				will unconditionally set this to 0 a) when the software sets Force Port Resume bit to 0 (from 1), Or b) when the software sets Port Reset bit to 1 (from 0) **Note:** Before setting this bit, Run/Stop bit should be set to 0.
6	F_PO_RESM	RW	1'b0	Force Port Resume 1: Resume detection/driven on port. 0: No resume detected/driven on port. Software sets this to 1 to resume signaling. The host controller sets this to 1 if a J-to-K transition is detected while the port is in the suspended state. When this transits to 1 for the detection of a J-to-K transition, the Port Change Detect bit in USBSTS register is also set to 1.
5:4	Reserved	-	-	-
3	PO_EN_CHG	RW1C	1'b0	Port Enable/Disable Change 1: Port enable/disable status has changed 0: No change
2	PO_EN	RW	1'b0	Port Enable/Disable 1: Enable 0: Disable Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a '1' to this bit. Writing a '0' to this bit to (forcefully) disable the port is possible however.
1	CONN_CHG	RW1C	1'b0	Connect Status Change 1: Change in current connect status 0: No change This indicates a change has occurred in the current connect status of the port.
0	CONN_STS	RO	1'b0	Current Connect Status 1: Device is presented on the port 0: No device is presented This reflects the current state of the port, and may not correspond directly to cause the Connect Status Change bit to be set.

Table 7.11 – PORTSC – HC Port Status and Control Register

7.3 Configuration Registers

7.3.1 EOF Time & Asynchronous Schedule Sleep Timer Register (address offset: 0x34)

Bit	Name	Type	Default Value	Description
31:7	Reserved	-	-	-
6	U_SUSP_N	RW	1'b1	Transceiver Suspend Mode Active low. Places the transceiver in the suspend mode that draws the minimal power from the power supplies. This is part of the power management.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
5:4	EOF2_Time	RW	2'h0	EOF2 Timing Points Controls EOF2 timing point before the next SOF. **High-Speed EOF2 Time** 0x0: 2 clocks (30MHz): 66 ns 0x1: 4 clocks (30MHz): 133 ns 0x2: 8 clocks (30MHz): 266 ns 0x3: 16 clocks (30MHz): 533 ns **Full-Speed EOF2 Time** 0x0: 20 clocks (30MHz): 666 ns 0x1: 40 clocks (30MHz): 1.333 us 0x2: 80 clocks (30MHz): 2.666 us 0x3: 160 clocks (30MHz): 5.333 us **Low-Speed EOF2 Time** 0x0: 40 clocks (30MHz): 1.333 us 0x1: 80 clocks (30MHz): 2.666 us 0x2: 160 clocks (30MHz): 5.333 us 0x3: 320 clocks (30MHz): 10.66 us
3:2	EOF1_Time	RW	2'h0	EOF1 Timing Points Controls EOF1 timing point before next SOF. This value should be adjusted according to the maximum packet size. **High-Speed EOF1 Time** 0x0: 540 clocks (30MHz): 18 us 0x1: 360 clocks (30MHz): 12 us 0x2: 180 clocks (30MHz): 6 us 0x3: 720 clocks (30MHz): 24 us **Full-Speed EOF1 Time** 0x0: 1600 clocks (30MHz): 53.3 us 0x1: 1400 clocks (30MHz): 46.6 us 0x2: 1200 clocks (30MHz): 40 us 0x3: 21000 clocks (30MHz): 700 us **Low-Speed EOF1 Time** 0x0: 3750 clocks (30MHz): 125 us 0x1: 3500 clocks (30MHz): 116 us 0x2: 3250 clocks (30MHz): 108 us 0x3: 4000 clocks (30MHz): 133 us
1:0	ASYN_SCH_SLPT	RW	2'h1	Asynchronous Schedule Sleep Timer Controls the Asynchronous Schedule sleep timer. 0x0: 5 us 0x1: 10 us 0x2: 15 us 0x3: 20 us

Table 7.12 – EOF Time & Asynchronous Schedule Sleep Timer Register

7.3.2 Bus Monitor Control / Status Register (address offset: 0x40)

Bit	Name	Type	Default Value	Description
31:11	Reserved	RO	21'b0	-
10:9	HOST_SPD_TYP	RO	2'h0	Host Speed Type 0x2: HS 0x0: FS 0x1: LS 0x3: Reserved

Copyright © Bridgetek Pte Ltd

8	VBUS_VLD	RO	1'b0	VBUS Valid When the voltage on the VBUS is above the valid VBUS threshold, this signal is valid.
7:5	Reserved	RW	3'b0	These bits should remain '0'.
4	VBUS_OFF	RW	1'b1	VBUS OFF This controls the voltage on the VBUS_ON/OFF. 0: VBUS on 1: VBUS off
3:2	Reserved	RW	2'b0	These bits should remain '0'.
1	HDISCON_FLT_SEL	RW	1'b0	Select a timer to filter out noise on HDISCON from the UTMI+ 0: Approximately 135 us 1: Approximately 270 us
0	VBUS_FLT_SEL	RW	1'b0	Select a timer to filter out noise on VBUS_VLD from the UTMI+ 0: Approximately 135 us 1: Approximately 472 us

Table 7.13 – Bus Monitor Control / Status Register

7.3.3 HPROT – Master Protection Information Setting Register (address offset: 0x78)

Bit	Name	Type	Default Value	Description
31:4	Reserved	-	-	-
3:0	HPROT	RW	4'h3	Master Protection Information For programmable HPROT, software can use this to implement some level of protection.

Table 7.14 – HPROT – Master Protection Information Setting Register

7.4 USB Host Testing Registers

7.4.1 Vendor Specific IO Control Register (address offset: 0x54)

Bit	Name	Type	Default Value	Description
31:6	Reserved	-	-	-
5	VCTLOAD_N	RW	1'b1	Vendor-Specific Test Mode Control Load. This controls the active low output U_VCTLOAD_N to the PHY. Setting this to 1 makes U_VCTLOAD_N output a 1. Setting this to 0 makes U_VCTLOAD_N output a 0.
4:0	VCTL	RW	5'h00	Vendor-Specific Test Mode Control The programmed value is delivered to the PHY via the U_VCTL output.

Table 7.15 – Vendor Specific IO Control Register

7.4.2 Vendor Specific Status Register (address offset: 0x58)

Bit	Name	Type	Default Value	Description
31:8	Reserved	-	-	-
7:0	VSTS	RO	Depends on the reset values of the PHY.	Vendor-Specific Test Mode Status

Table 7.16 – Vendor Specific Status Register

Copyright © Bridgetek Pte Ltd

7.4.3 Test Register (address offset: 0x50)

Bit	Name	Type	Default Value	Description
31:5	Reserved	-	-	-
4	TST_LOOPBK	RW	1'b0	FIFO Loop Back Mode When this is set to 1, the host controller will enter the loop-back mode.
3	TST_MOD	RW	1'b0	Test Mode When this is set to 1, the host controller will enter the test mode. This test mode can save the simulation time. In the normal mode, the host controller uses a counter for 10ms detection of the USB reset. This is reduced in test mode.
2	TST_PKT	RW	1'b0	Test Packet Upon entering HS mode and setting this bit to 1, the test packet data defined in the USB specification has to be written. The host controller will repeatedly send the packet defined in the UTMI specification to the transceiver.
1	TST_KSTA	RW	1'b0	When this is set to 1, D+/D- are set to HS K state.
0	TST_JSTA	RW	1'b0	When this is set to 1, D+/D- are set to HS J state.

Table 7.17 – Test Register

Copyright © Bridgetek Pte Ltd

8 USB Device

The USB device controller is fully compliant with the USB 2.0 specification. It supports a control endpoint (EP0) and up to 7 other endpoints (EP1-7).

The buffer size of EP0 ranges from 8 to 64 bytes, configurable via software. EP1-7 support optional double buffering. The number of endpoints supported can be set by software, as well as their individual direction, type (interrupt, bulk, and isochronous) and buffer size (8-1024 bytes). The total buffer size to be shared by EP1-7 is 6 kB. However, if EP0 is configured to use less than 64 bytes for its buffer, the difference will be added to the total memory available to the other endpoints. For example, if EP0 buffer size is set to 16 bytes, the total memory available to EP1-7 will be 6 kB + 48 bytes.

One of the data endpoints (EP1-7) can be configured as High Bandwidth (HBW) isochronous. For more information on the HBW isochronous endpoint, please refer to section 8.6.

8.1 Register Summary

Listed below are the registers with their offset from the base address (0x10180). The registers for High Bandwidth Isochronous support have a base address of 0x10A1C. All registers and buffer locations can only be accessed via Byte (8-bit) mode, except the endpoint buffer registers, which can be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit) mode.

Address Offset	Register	Default value	References
	Initialization Registers		
0x18	DC_ADDRESS_ENABLE – Address Register	0x00	Section 8.2.1
0x10	DC_MODE – Mode Register	0x00	Section 8.2.2
0x08	DC_INT_ENABLE – Interrupt Enable Register	0x00	Section 8.2.3
0x0C	DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register	0x00	Section 8.2.4
	Control Endpoint Data Flow Registers		
0x1C	DC_EP0_CONTROL – Endpoint 0 Control Register	0x00	Section 8.3.1
0x20	DC_EP0_STATUS – Endpoint 0 Status Register	0x00	Section 8.3.2
0x24	DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register	0x00	Section 8.3.3
0x28	DC_EP0_BUFFER – Endpoint 0 Buffer Register	0x00	Section 8.3.4
	Other Endpoints Data Flow Registers		
0x2C	DC_EP1_CONTROL – Endpoint 1 Control Register	0x00	Section 8.4.1
0x30	DC_EP1_STATUS – Endpoint 1 Status Register	0x00	Section 8.4.2
0x34	DC_EP1_BUFFER_LENGTH_LSB – Endpoint 1 Buffer Length LSB Register	0x00	Section 8.4.3
0x35	DC_EP1_BUFFER_LENGTH_MSB – Endpoint 1 Buffer Length MSB Register	0x00	Section 8.4.4

Copyright © Bridgetek Pte Ltd

0x38	DC_EP1_BUFFER – Endpoint 1 Buffer Register	0x00	Section 8.4.5
0x3C	DC_EP2_CONTROL – Endpoint 2 Control Register	0x00	Section 8.4.1
0x40	DC_EP2_STATUS – Endpoint 2 Status Register	0x00	Section 8.4.2
0x44	DC_EP2_BUFFER_LENGTH_LSB – Endpoint 2 Buffer Length LSB Register	0x00	Section 8.4.3
0x45	DC_EP2_BUFFER_LENGTH_MSB – Endpoint 2 Buffer Length MSB Register	0x00	Section 8.4.4
0x48	DC_EP2_BUFFER – Endpoint 2 Buffer Register	0x00	Section 8.4.5
0x4C	DC_EP3_CONTROL – Endpoint 3 Control Register	0x00	Section 8.4.1
0x50	DC_EP3_STATUS – Endpoint 3 Status Register	0x00	Section 8.4.2
0x54	DC_EP3_BUFFER_LENGTH_LSB – Endpoint 3 Buffer Length LSB Register	0x00	Section 8.4.3
0x55	DC_EP3_BUFFER_LENGTH_MSB – Endpoint 3 Buffer Length MSB Register	0x00	Section 8.4.4
0x58	DC_EP3_BUFFER – Endpoint 3 Buffer Register	0x00	Section 8.4.5
0x5C	DC_EP4_CONTROL – Endpoint 4 Control Register	0x00	Section 8.4.1
0x60	DC_EP4_STATUS – Endpoint 4 Status Register	0x00	Section 8.4.2
0x64	DC_EP4_BUFFER_LENGTH_LSB – Endpoint 4 Buffer Length LSB Register	0x00	Section 8.4.3
0x65	DC_EP4_BUFFER_LENGTH_MSB – Endpoint 4 Buffer Length MSB Register	0x00	Section 8.4.4
0x68	DC_EP4_BUFFER – Endpoint 4 Buffer Register	0x00	Section 8.4.5
0x6C	DC_EP5_CONTROL – Endpoint 5 Control Register	0x00	Section 8.4.1
0x70	DC_EP5_STATUS – Endpoint 5 Status Register	0x00	Section 8.4.2
0x74	DC_EP5_BUFFER_LENGTH_LSB – Endpoint 5 Buffer Length LSB Register	0x00	Section 8.4.3
0x75	DC_EP5_BUFFER_LENGTH_MSB – Endpoint 5 Buffer Length MSB Register	0x00	Section 8.4.4
0x78	DC_EP5_BUFFER – Endpoint 5 Buffer Register	0x00	Section 8.4.5
0x7C	DC_EP6_CONTROL – Endpoint 6 Control Register	0x00	Section 8.4.1
0x80	DC_EP6_STATUS – Endpoint 6 Status Register	0x00	Section 8.4.2
0x84	DC_EP6_BUFFER_LENGTH_LSB – Endpoint 6 Buffer Length LSB Register	0x00	Section 8.4.3
0x85	DC_EP6_BUFFER_LENGTH_MSB – Endpoint 6 Buffer Length MSB Register	0x00	Section 8.4.4

Copyright © Bridgetek Pte Ltd

0x88	DC_EP6_BUFFER – Endpoint 6 Buffer Register	0x00	Section 8.4.5
0x8C	DC_EP7_CONTROL – Endpoint 7 Control Register	0x00	Section 8.4.1
0x90	DC_EP7_STATUS – Endpoint 7 Status Register	0x00	Section 8.4.2
0x94	DC_EP7_BUFFER_LENGTH_LSB – Endpoint 7 Buffer Length LSB Register	0x00	Section 8.4.3
0x95	DC_EP7_BUFFER_LENGTH_MSB – Endpoint 7 Buffer Length MSB Register	0x00	Section 8.4.4
0x98	DC_EP7_BUFFER – Endpoint 7 Buffer Register	0x00	Section 8.4.5
	General Registers		
0x00	DC_INT_STATUS – Interrupt Status Register	0x00	Section 8.5.1
0x04	DC_EP_INT_STATUS – Endpoints Interrupt Status Register	0x00	Section 8.5.2
0x14	DC_FRAME_NUMBER_LSB – Frame Number LSB Register	0x00	Section 8.5.3
0x15	DC_FRAME_NUMBER_MSB – Frame Number MSB Register	0x00	Section 8.5.4
	High Bandwidth Isochronous Registers (Base Address: 0x10A1C)		
0x00	HBWCTRL0 – High-bandwidth Isochronous Control Register 0	0x00	Section 8.6.1
0x04	HBWCTRL1 – High-bandwidth Isochronous Control Register 1	0x3D	Section 8.6.2
0x08	HBWCTRL2 – High-bandwidth Isochronous Control Register 2	0x0C	Section 8.6.3
0x0C	HBWCTRL3 – High-bandwidth Isochronous Control Register 3	0x80	Section 8.6.4
0x10	HBW_INT_ENABLE – High-bandwidth Interrupt Enable Register	0x00	Section 8.6.5
0x14	HBW_INT_STATUS – High-bandwidth Interrupt Status Register	0x00	Section 8.6.6

Table 8.1 – Overview of USB Device Registers

8.2 Initialization Registers

8.2.1 DC_ADDRESS_ENABLE – Address Register (address offset: 0x18)

This register sets the USB assigned address sent from the USB host and enables the USB device. In response to the standard USB request SET_ADDRESS, the firmware must write the peripheral address to this register.

Bit	Name	Type	Default Value	Description
7	ENABLE	RO	1'b0	Hardware sets this to 1 when software writes a new address to this register. It is cleared by hardware at the end of the current transfer when the new address will take effect.
6:0	ADDR	RW	7'h00	Function Address

Table 8.2 – DC_ADDRESS_ENABLE – Address Register

Copyright © Bridgetek Pte Ltd

8.2.2 DC_MODE – Mode Register (address offset: 0x10)

This register allows the firmware to select the different test modes and enables the USB device function.

Bit	Name	Type	Default Value	Description
7	TST_MODE_ENABLE	RW	1'b0	Test Mode Enable. Setting this to 1 to enter the test mode. It can only be cleared by hardware reset.
6:5	TST_MODE_SELECT	RW	2'h0	Test Mode select (writeable only if MODE_ENABLE is 0) 2'h0: SE0_NAK 2'h1: J 2'h2: K 2'h3: Packet
4:2	Reserved	-	-	-
1	FS_ONLY	RW	1'b0	Setting this to 1 disables HS detection handshake.
0	USB_DEV_EN	RW	1'b0	USB function enables. Setting this to 1 enables the USB device.

Table 8.3 – DC_MODE – Mode Register

8.2.3 DC_INT_ENABLE – Interrupt Enable Register (address offset: 0x08)

This register enables the different interrupt sources by writing a 1 to the corresponding bit.

Bit	Name	Type	Default Value	Description
7	PHY_IE	RW	1'b0	PHY receive error interrupt enable
6	PID_IE	RW	1'b0	Package ID error interrupt enable
5	CRC16_IE	RW	1'b0	CRC16 error interrupt enable
4	CRC5_IE	RW	1'b0	CRC5 error interrupt enable
3	RESM_IE	RW	1'b0	Resume interrupt enable
2	SUS_IE	RW	1'b0	Suspend interrupt enable
1	RST_IE	RW	1'b0	Reset interrupt enable
0	SOF_IE	RW	1'b0	Start of Frame interrupt enable

Table 8.4 – DC_INT_ENABLE – Interrupt Enable Register

8.2.4 DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register (address offset: 0x0C)

This register enables the different interrupt sources based on the specific endpoints by writing 1 to the corresponding bit.

Bit	Name	Type	Default Value	Description
7	EP7_IE	RW	1'b0	Endpoint 7 interrupt enable
6	EP6_IE	RW	1'b0	Endpoint 6 interrupt enable
5	EP5_IE	RW	1'b0	Endpoint 5 interrupt enable
4	EP4_IE	RW	1'b0	Endpoint 4 interrupt enable
3	EP3_IE	RW	1'b0	Endpoint 3 interrupt enable
2	EP2_IE	RW	1'b0	Endpoint 2 interrupt enable
1	EP1_IE	RW	1'b0	Endpoint 1 interrupt enable
0	EP0_IE	RW	1'b0	Endpoint 0 interrupt enable

Table 8.5 – DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register

8.3 Control Endpoint Data Flow Registers

Control Endpoint Data Flow registers are used for configuring the control endpoint, and handle the sending and receiving of data to the control endpoint.

Copyright © Bridgetek Pte Ltd

8.3.1 DC_EP0_CONTROL – Endpoint 0 Control Register (address offset: 0x1C)

This register configures the maximum packet size of the control endpoint 0. It is also used to stall the control endpoint.

Bit	Name	Type	Default Value	Description
7:3	Reserved	-	-	-
2:1	EP_SIZE	RW	2'h0	Endpoint Maximum packet size 2'h0: 8 bytes 2'h1: 16 bytes 2'h2: 32 bytes 2'h3: 64 bytes
0	STALL	RW	1'b0	Send STALL. Software writes 1 to send a STALL handshake in response to an IN token. Software writes 0 to terminate the STALL signalling.

Table 8.6 – DC_EP0_CONTROL – Endpoint 0 Control Register

8.3.2 DC_EP0_STATUS – Endpoint 0 Status Register (address offset: 0x20)

This register is used by the hardware to report the status of the control endpoint 0. Software writes 1 to the corresponding register bit to clear the status.

Bit	Name	Type	Default Value	Description
7:5	Reserved	-	-	-
4	DATA_END	RW1S	1'b0	Data End. Software should update this bit to 1 when writing: • 1 to IN_PKT_RDY for last outgoing data packet • 1 to IN_PKT_RDY for a zero-length data packet • 1 to OUT_PKT_RDY after servicing the last incoming data packet
3	STALL	RW1C	1'b0	Sent STALL. Hardware sets this to 1 when the STALL handshake has been transmitted. Software writes 1 to clear it.
2	SETUP	RW1C	1'b0	SETUP token received. Hardware sets this to 1 and interrupts when the SETUP token has been received. Software writes 1 to clear it.
1	IN_PKT_RDY	RW1S	1'b0	IN packet ready. Software should write 1 to it after loading a data packet into the endpoint 0 IN FIFO. Hardware clears it and generates an interrupt when the data packet has been successfully transmitted.
0	OUT_PKT_RDY	RW1C	1'b0	OUT packet ready. Hardware sets this bit to 1 and generates an interrupt when a data packet has been received. Software writes 1 to clear it after unloading the data packet from the endpoint 0 OUT FIFO.

Table 8.7 – DC_EP0_STATUS – Endpoint 0 Status Register

8.3.3 DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register (address offset: 0x24)

This register is used by the hardware to report the length of the packet received in the endpoint 0 OUT buffer.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	BUF_LEN	RO	7'h00	Indicates the number of data bytes received. Valid only if OUT_PKT_RDY is 1.

Table 8.8 – DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register

8.3.4 DC_EP0_BUFFER – Endpoint 0 Buffer Register (address offset: 0x28)

This register is used to access the endpoint 0 FIFO.

Bit	Name	Type	Default Value	Description
7:0	Data	R/W	8'h00	Endpoint 0 FIFO window register. A read unloads one data byte from the Endpoint 0 OUT FIFO. A write loads one data byte into the Endpoint 0 IN FIFO.

Table 8.9 – DC_EP0_BUFFER – Endpoint 0 Buffer Register

8.4 Other Endpoint Data Flow Registers

This is a set of registers used to access the other endpoints from endpoint 1 to endpoint 7. The register definitions are the same for the registers from endpoint 1 to endpoint 7 but they are located at different offset addresses.

8.4.1 DC_EP(x)_CONTROL – Endpoint Control Registers (address offset: 0x2C/0x3C/0x4C/0x5C/0x6C/0x7C/0x8C)

The different endpoints are configured by writing to the corresponding endpoint control register selected based on the corresponding offset address. This register is used to configure the endpoint type, direction, maximum packet size and double buffering. It is also used to stall the corresponding data endpoint.

Bit	Name	Type	Default Value	Description
7	DBL_BUF	RW	1'b0	Endpoint double buffering Enable. 0: double buffering disabled 1: double buffering enabled
6:4	EP_SIZE	RW	3'h0	Endpoint Maximum packet size. This parameter has to be fixed during the Set Configuration request. 0x0: 8 bytes 0x1: 16 bytes 0x2: 32 bytes 0x3: 64 bytes 0x4: 128 bytes 0x5: 256 bytes 0x6: 512 bytes 0x7: 1024 bytes
3	STALL	RW	1'b0	Send STALL. Valid only when the endpoint is in bulk or interrupt mode. Software should write a 1 to this bit to send a STALL handshake in response to IN token, PING token and data phase of OUT transaction. Software should write a 0 to this bit to terminate the STALL signalling.
2:1	EP_MODE	RW	2'h0	Endpoint Mode. This parameter has to be fixed during the Set Configuration request. 0x0: EP disabled 0x1: EP configured for bulk transfers

Copyright © Bridgetek Pte Ltd

				0x2: EP configured for interrupt transfers 0x3: EP configured for isochronous transfers
0	EP_DIR	RW	1'b0	Endpoint Direction. This parameter has to be fixed during the Set Configuration request. 0x0: EP direction selected as OUT 0x1: EP direction selected as IN

Table 8.10 – DC_EP(x)_CONTROL – Endpoint Control Registers

8.4.2 DC_EP(x)_STATUS – Endpoint Status Registers (address offset: 0x30/0x40/0x50/0x60/0x70/0x80/0x90)

The different endpoint statuses are read by accessing the corresponding status register. The hardware reports the endpoint status in this register. Write a 1 to the corresponding register bit to clear the bit.

Bit	Name	Type	Default Value	Description
7	CLR_TOGGLE	RW1S	1'b0	Clear data toggle. Software can write a 1 to this bit to reset the Endpoint data toggle to 0. This bit is always read as 0.
6	FIFO_FLUSH	RW1S	1'b0	FIFO Flush. Valid only when the EP direction is IN. Writing 1 to this bit flushes the next packet to be transmitted from the Endpoint IN FIFO. The FIFO pointer is reset and the IN_PKT_RDY bit is cleared. Hardware resets the FLUSH bit to 0 when the FIFO flush is complete.
5	DATA_ERR	RW1C	1'b0	Data error. Valid only when the Endpoint is in isochronous mode and the direction is OUT. This flag is set to 1 by hardware if a received packet has a CRC-16 error. It is automatically cleared when software clears the OUT_PKT_RDY bit.
4	STALL	RW1C	1'b0	Sent STALL. Valid only when the Endpoint is in bulk or interrupt mode. Hardware sets this bit to 1 when the STALL handshake is transmitted. Software can clear this bit by writing a 1 to this bit.
3	UNDER_RUN	RW1C	1'b0	Data underrun. Valid only when the Endpoint direction is IN. Its function is dependent upon the Endpoint mode: Isochronous: Hardware sets this to 1 when a zero-length packet is sent in response to an IN token while IN_PKT_RDY is 0. Bulk/Interrupt: Hardware sets this to 1 when a NAK packet is sent in response to an IN token while IN_PKT_RDY is 0. Software can clear this bit by writing a 1 to this bit.
2	OVER_RUN	RW1C	1'b0	Data overrun. Valid only when the Endpoint is in isochronous mode and the direction is OUT. Hardware sets this bit to 1 if a received packet cannot be loaded into the Endpoint FIFO. Software can clear this bit by writing a 1 to this bit.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
1	IN_PKT_RDY	RW1S	1'b0	IN packet ready. Valid only when the Endpoint direction is IN. Software should write a 1 to this bit after loading a data packet into the Endpoint IN FIFO. Hardware clears this bit and generates an interrupt when the data packet has been successfully transmitted
0	OUT_PKT_RDY	RW1C	1'b0	OUT packet ready. Valid only when the Endpoint direction is OUT. Hardware sets this bit to 1 and generates an interrupt when a data packet has been received. Software writes a 1 to clear it after unloading the data packet from the Endpoint OUT FIFO.

Table 8.11 – DC_EP(x)_STATUS – Endpoint Status Registers

8.4.3 DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers (address offset: 0x34/0x44/0x54/0x64/0x74/0x84/0x94)

The different endpoint buffer length is read by accessing the corresponding buffer length register. This register reports the LSB of the OUT-buffer length.

Bit	Name	Type	Default Value	Description
7:0	BUF_LEN_LSB	RO	8'h00	Indicates the low byte of the number of received data bytes in the Endpoint FIFO. Valid only if OUT_PKT_RDY is 1.

Table 8.12 – DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers

8.4.4 DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers (address offset: 0x35/0x45/0x55/0x65/0x75/0x85/0x95)

The different endpoint buffer length is read by accessing the corresponding buffer length register. This register reports the MSB of the OUT-buffer length.

Bit	Name	Type	Default Value	Description
7:3	Reserved	RO	5'h00	-
2:0	BUF_LEN_MSB	RO	3'h0	Indicates the high 3-bit of the number of received data bytes in the Endpoint FIFO. Valid only if OUT_PKT_RDY is 1.

Table 8.13 – DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers

8.4.5 DC_EP(x)_BUFFER – Endpoint Buffer Registers (address offset: 0x38/0x48/0x58/0x68/0x78/0x88/0x98)

The different endpoint buffer is accessed by accessing the corresponding buffer register.

Bit	Name	Type	Default Value	Description
7:0	Buffer	RW	8'h00	Endpoint FIFO window register. A read unloads one data byte from the Endpoint OUT FIFO. A write loads one data byte into the Endpoint IN FIFO.

Table 8.14 – DC_EP(x)_BUFFER – Endpoint Buffer Registers

Copyright © Bridgetek Pte Ltd

8.5 General Registers

8.5.1 DC_INT_STATUS – Interrupt Status Register (address offset: 0x00)

This register indicates that the hardware condition of the corresponding interrupt has occurred. Write a 1 to clear the corresponding bit.

Bit	Name	Type	Default Value	Description
7	PHY	RW1C	1'b0	PHY receive error interrupt status
6	PID	RW1C	1'b0	Package ID error interrupt status
5	CRC16	RW1C	1'b0	CRC16 error interrupt status
4	CRC5	RW1C	1'b0	CRC5 error interrupt status
3	RESM	RW1C	1'b0	Resume interrupt status
2	SUS	RW1C	1'b0	Suspend interrupt status
1	RST	RW1C	1'b0	Reset interrupt status
0	SOF	RW1C	1'b0	Start of Frame interrupt status

Table 8.15 – DC_INT_STATUS – Interrupt Status Register

8.5.2 DC_EP_INT_STATUS – Endpoints Interrupt Status Register (address offset: 0x04)

This register indicates the corresponding endpoint hardware condition has occurred. Write a 1 to clear the corresponding bit.

Bit	Name	Type	Default Value	Description
7	EP7	RW1C	1'b0	Endpoint 7 interrupt status
6	EP6	RW1C	1'b0	Endpoint 6 interrupt status
5	EP5	RW1C	1'b0	Endpoint 5 interrupt status
4	EP4	RW1C	1'b0	Endpoint 4 interrupt status
3	EP3	RW1C	1'b0	Endpoint 3 interrupt status
2	EP2	RW1C	1'b0	Endpoint 2 interrupt status
1	EP1	RW1C	1'b0	Endpoint 1 interrupt status
0	EP0	RW1C	1'b0	Endpoint 0 interrupt status

Table 8.16 – DC_EP_INT_STATUS – Endpoints Interrupt Status Register

8.5.3 DC_FRAME_NUMBER_LSB – Frame Number LSB Register (address offset: 0x14)

This register has the LSB of the last successfully received SOF.

Bit	Name	Type	Default Value	Description
7:0	FRAME_LSB	RO	8'h00	Frame Number for last received SOF, least significant byte.

Table 8.17 – DC_FRAME_NUMBER_LSB – Frame Number LSB Register

8.5.4 DC_FRAME_NUMBER_MSB – Frame Number MSB Register (address offset: 0x15)

This register has the MSB of the last successfully received SOF.

Bit	Name	Type	Default Value	Description
7:3	Reserved	RO	5'h00	-
2:0	FRAME_MSB	RO	3'h0	Frame Number for last received SOF, Most significant byte.

Table 8.18 – DC_FRAME_NUMBER_MSB – Frame Number MSB Register

8.6 High-bandwidth Isochronous IN Support & Registers

Note: The registers for high-bandwidth isochronous IN support has a base address of **0x10A1C**.

Copyright © Bridgetek Pte Ltd

8.6.1 HBWCTRL0 – High-bandwidth Isochronous Control Register 0 (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7	HBW_INI	RWAC	1'b0	Set this bit to trigger a reset to HBW isochronous IN pipe. Only RO bits will be reset by HBW_INI. Those are OVERRUN, HAVE_SPACE and FULL. After the reset, the bit is automatically cleared.
6	OVERRUN	RO	1'b0	Indicates when the FIFO is overrun. The overrun condition will be cleared by HBW_INI. 1: overrun 0: normal
5	Reserved	-	-	-
4	PTSON	RW	1'b0	Enables the PTS engine for the UVC header. 1: enabled 0: disabled
3	SCRON	RW	1'b0	Enables the SCR engine for the UVC header 1: enabled 0: disabled
2	SET_STILL	RW	1'b0	Sets the transferring video frame as still image. Should be set before the image is fed into the HBW ISO-IN FIFO. 1: still image 0: video stream
1	SET_ERROR	RW	1'b0	Sets the transferring video frame as error image. The setting will be correlated to the field in UVC header. 1: error 0: normal video frame
0	AUTO_HEADER	RW	1'b0	Automatically inserts the UVC header and transfer the existing FIFO data on the USB bus. 1: enabled 0: disabled In this mode, the FIFO size can be set up to 16 kB (note that the actual memory available is limited to 6 kB). The HBW engine will automatically decide how much data is sent on the bus based on the amount of data available in the FIFO.

Table 8.19 – HBWCTRL0 – High-bandwidth Isochronous Control Register 0

8.6.2 HBWCTRL1 – High-bandwidth Isochronous Control Register 1 (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	FIFO_SIZE	RW	7'h3D	Defines the FIFO size for the HBW isochronous IN pipe. The FIFO size = FIFO_SIZE * 64 (bytes), and is located at the top of the 6kB shared memory of the USB Device.

Table 8.20 – HBWCTRL1 – High-bandwidth Isochronous Control Register 1

Copyright © Bridgetek Pte Ltd

8.6.3 HBWCTRL2 – High-bandwidth Isochronous Control Register 2 (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:4	MAXPKTSIZE	RWAC	4'h0	Defines the maximum packet size for the HBW isochronous IN pipe. Max packet size = (MAXPKTSIZE + 1) * 64 (bytes).
3:2	HBWMODE	RO	2'b11	00: Reserved 01: Expects 1 isochronous IN. (DATA0) 10: Expects 2 isochronous INs. (DATA1/0) 11: Expects 3 isochronous INs. (DATA2/1/0)
1:0	THSD_SEL	RW	2'b00	Sets the threshold level for the HAVESPACE indicator. 00: 1024 bytes 01: 64 bytes 10: 128 bytes 11: 256 bytes

Table 8.21 – HBWCTRL2 – High-bandwidth Isochronous Control Register 2

8.6.4 HBWCTRL3 – High-bandwidth Isochronous Control Register 3 (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7	HAVE_SPACE	RO	1'b1	Indicates that there is still at least one burst space correlated to THSD_SEL. 1: at least 1 burst space for data is available 0: not enough space
6	FULL	RO	1'b0	Indicates the status of the HBW FIFO 1: full 0: not full
5	HBWINPRDY	RWAC	1'b0	Set this bit to indicate the data in the FIFO is ready to be transferred on the USB bus. 1: data is ready In this mode, only up to (3 * maximum packet size) of the FIFO will be used. The software needs to check the HAVE_SPACE bit to continue feeding in data. **Note:** HBWINPRDY should not be set when AUTO_HEADER is being enabled. Otherwise, unexpected behavior may occur.
4	SEQDATA_END	RWAC	1'b0	Set this bit to indicate the end of sequential data. 1: triggers an end flag to terminate a sequential data
3:0	HBW_ENDP_NUM	RW	4'h0	Hooks up the HBW isochronous IN pipe to an endpoint number. 4'h0: disable this pipe Others: specify an endpoint number

Table 8.22 – HBWCTRL3 – High-bandwidth Isochronous Control Register 3

Copyright © Bridgetek Pte Ltd

8.6.5 HBW_INT_ENABLE – High-bandwidth Interrupt Enable Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7:1	Reserved	RO	7'h00	-
0	HBW_INTEN	RW	1'b0	Interrupt Enable for the HBW pipe 1: enabled 0: disabled Note that after the HBW pipe is hooked up to certain endpoint number, the normal (i.e., non-HBW) interrupts for that endpoint will be disabled.

Table 8.23 – HBW_INT_ENABLE – High-bandwidth Interrupt Enable Register

8.6.6 HBW_INT_STATUS – High-bandwidth Interrupt Status Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7:1	Reserved	RO	7'h00	-
0	HBW_INTR	RW1C	1'b0	Interrupt Status for HBW pipe. 1: An interrupt event has occurred (caused by HAVE_SPACE) 0: No interrupt event has occurred

Table 8.24 – HBW_INT_STATUS – High-bandwidth Interrupt Status Register

8.6.7 Behavior of the HAVE_SPACE Bit

8.6.7.1 Initial status

- HAVE_SPACE will be functionally active after the pipe is hooked up to an endpoint.
- HAVE_SPACE will be set to '1' after the first SOF occurs.

8.6.7.2 The behavior when data is written into the FIFO

- When data is being written into the HBW FIFO, HAVESPACE is temporarily cleared.
- After all data has been written to the FIFO, if the FIFO still has more than 1 kB of space, HAVE_SPACE will set back to '1'.
- Otherwise, if the remaining FIFO space is less than 1 kB, HAVE_SPACE will remain '0'.

8.6.7.3 The behavior when SEQDATA_END is set

- HAVE_SPACE will be cleared after SEQDATA_END is set.
- After the zero-length packet has been transferred on the bus, HAVE_SPACE will be set back to '1'.

8.6.7.4 The behavior when HBWINPRDY is set

- HAVE_SPACE will be cleared after HBWINPRDY is set.
- After the data has been transferred on the bus and more than 1 kB of space is available in the FIFO, HAVE_SPACE will be set back to '1'.

8.6.8 High-bandwidth Isochronous Pipe in UVC

8.6.8.1 The UVC Payload Header

The UVC payload header should be inserted in the isochronous IN pipe after the first SOF.

Copyright © Bridgetek Pte Ltd

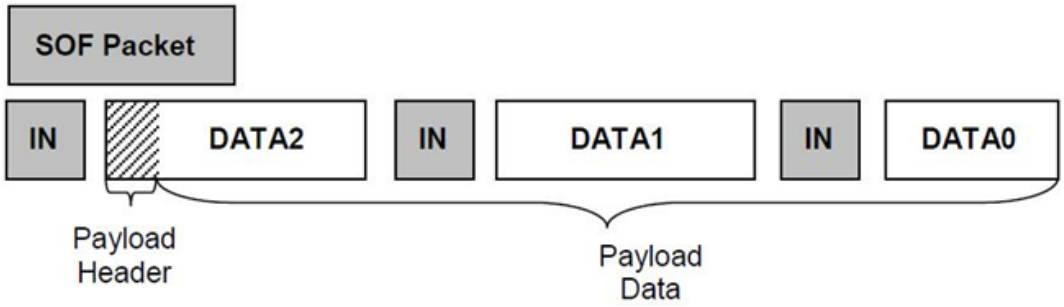


Figure 8.1 - UVC Payload Header transferred over an IN endpoint

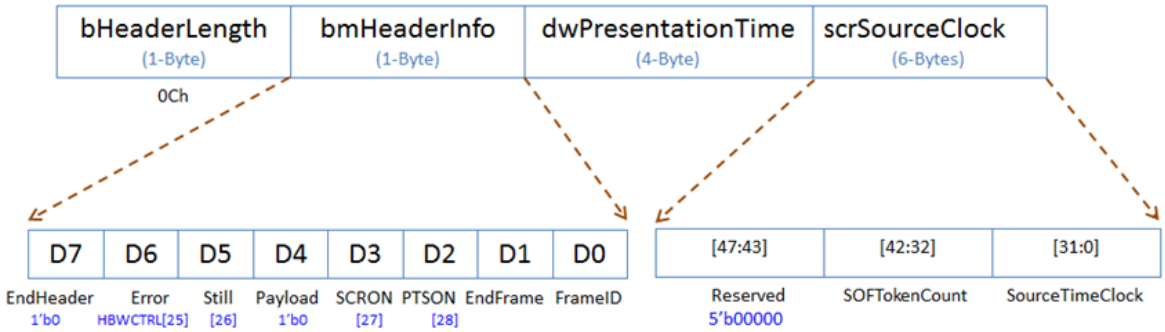
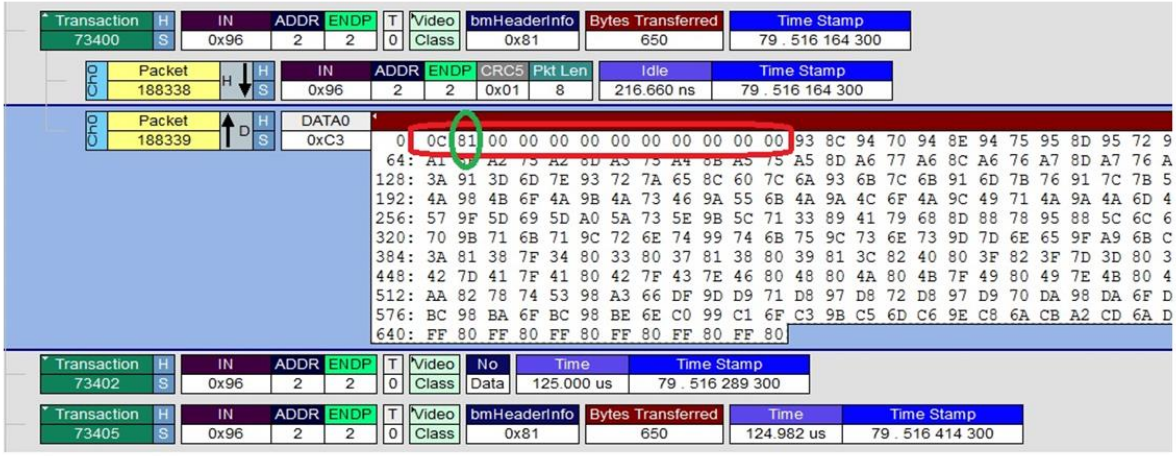


Figure 8.2 - Format of the UVC Payload Header



UVC payload header: 12 bytes

PTSON & SCRON are not enabled

Figure 8.3 - Example of an UVC Payload Header on the USB Bus

8.6.8.2 Enabling the UVC Payload Header Generator

- Set **AUTO_HEADER** (HBWCTRL0[0]) to enable the automatic header generator.
- The Header length is fixed at 12 bytes.
- The EndHeader bit in bmHeaderInfo (refer to figure 8.2) is set when **PTSON** (HBWCTRL0[4]) and **SCRON** (HBWCTRL0[3]) are cleared.
- **PTSON/SCRON/SET_STILL/SET_ERROR** bits in HBWCTRL0 will directly affect the D2/D3/D5/D6 bits of bmHeaderInfo.
- The Payload bit in bmHeaderInfo is always 0.
- The EndFrame bit in bmHeaderInfo will be set after a video frame is transmitted.

Copyright © Bridgetek Pte Ltd

8.6.9 Programming Guide for Using HBW Isochronous Support

8.6.9.1 Automatic Header Mode

- **Initialize the HBW isochronous pipe**
  1. Configure the endpoint that will be hooked up to the HBW isochronous pipe as a single buffer, isochronous endpoint. For example, if endpoint 2 is used, set **DC_EP2_CONTROL** to 0x77 (remember that only HBW isochronous IN transaction is supported).
  2. Set **AUTO_HEADER** bit (HBWCTRL0[0]).
  3. Set **FIFO_SIZE** bits (HBWCTRL1[6:0]), e.g., 0x40 (4 kB)

- **Set the packet size and hook up the endpoint number according to SET_INTERFACE from the USB host**
  1. Set **HBWMODE** (HBWCTRL2[3:2]) according to the appropriate alternate setting.
  2. Set **MAXPKTSIZE** (HBWCTRL2 [7:4]) according to the appropriate alternate setting, e.g., 0x0F (1024 bytes).

- **Hook up the HBW isochronous pipe to the logical endpoint number via HBW_ENDP_NUM**
  Set **HBW_ENDP_NUM** (HBWCTR3[3:0]) as endpoint 2, i.e., HBW_ENDP_NUM = 0x02

- **Feed data into the HBW isochronous pipe**
  1. Check **HAVE_SPACE** (HBWCTRL3[7]). If **HAVE_SPACE** = 1, it means the HBW pipe can be continuously written to.
  1. Feed 1024 bytes data into the buffer register of endpoint 2 **(DC_EP2_BUFFER)**
  2. Keep on checking **HAVE_SPACE** before feeding new data
  3. Set **SEQEND** (HBWCTRL3[4]) to terminate a video frame. The frame end payload for UVC will be automatically generated.

8.6.9.2 UVC Header is managed by firmware

- **Initialize the HBW pipe**
  1. Set the endpoint that will be hooked up to the HBW isochronous pipe as a single buffer, isochronous endpoint. For example, if endpoint 2 is used, set **DC_EP2_CONTROL** to 0x77 (remember that only HBW isochronous IN transaction is supported).
  2. Clear **AUTO_HEADER** bit (HBWCTRL0[0]).
  3. Set **FIFO_SIZE** bits (HBWCTRL1[6:0]), e.g., 0x30 (3 kB)

- **Set the packet size and hook up Endpoint number according to SET_INTERFACE from USB host**
  1. Set **HBWMODE** (HBWCTRL2[3:2]) according to the appropriate alternate setting.
  2. Set **MAXPKTSIZE** (HBWCTRL2 [7:4]) according to the appropriate alternate setting, e.g., 0x0F (1024 bytes).

- **Hook up the HBW isochronous pipe to the logical endpoint number via HBW_ENDP_NUM**
  Set **HBW_ENDP_NUM** (HBWCTR3[3:0]) as endpoint 2, i.e., HBW_ENDP_NUM = 0x02.

- **Feed in data to USB HBW pipe**
  1. Check **HAVE_SPACE** (HBWCTRL3[7]). If **HAVE_SPACE** = 1, it means the HBW pipe can be continuously written to.
  2. Feed 1024 bytes data into the buffer register of endpoint 2 **(DC_EP2_BUFFER).**
  3. Check **HAVE_SPACE** before feeding new data.
  4. Set **HBWINPRDY** (HBWCTRL3[5]) to start the data transfer on the USB bus.
  5. Keep on checking **HAVE_SPACE** before feeding new data.

Copyright © Bridgetek Pte Ltd

9 Ethernet

This is a MAC core that conforms to the IEEE 802.3-2002 specification with the following features:

- Supports 10BASE-T and 100BASE-TX/FX modes
- Supports full and half duplex operation at 10Mbps or 100Mbps
- CRC-32 algorithm calculates the FCS value one nibble at a time, automatic FCS generation and checking, able to capture frames with CRC errors if required
- Programmable MAC address
- Supports promiscuous mode
- Station Management (STA) entity included
- 2 kB TX buffer (arranged as 512x32)
- 4 kB RX buffer (arranged as 1024x32)

Memory organization of the TX/RX RAM is listed below.

Offset	Data Bits	TX RAM Content	RX RAM Content
0x0000	7:0	Data Length LSB	Frame Length LSB
	15:8	Data Length MSB	Frame Length MSB
	23:16	Destination Address Octet 1	Destination Address Octet 1
	31:24	Destination Address Octet 2	Destination Address Octet 2
0x0001	7:0	Destination Address Octet 3	Destination Address Octet 3
	15:8	Destination Address Octet 4	Destination Address Octet 4
	23:16	Destination Address Octet 5	Destination Address Octet 5
	31:24	Destination Address Octet 6	Destination Address Octet 6
0x0002	7:0	Source Address Octet 1	Source Address Octet 1
	15:8	Source Address Octet 2	Source Address Octet 2
	23:16	Source Address Octet 3	Source Address Octet 3
	31:24	Source Address Octet 4	Source Address Octet 4
0x0003	7:0	Source Address Octet 5	Source Address Octet 5
	15:8	Source Address Octet 6	Source Address Octet 6
	23:16	Type/Len MSB	Type/Len MSB
	31:24	Type/Len LSB	Type/Len LSB
0x0004 – 0x01FF	7:0	Data octet n	Data octet n
	15:8	Data octet n+1	Data octet n+1
	23:16	Data octet n+2	Data octet n+2
	31:24	Data octet n+3	Data octet n+3

Table 9.1 – Memory Organization of TX/RX RAM

The memory is accessed indirectly using the data register. Data bits from the table above directly correspond to data bits of the Data Register. The TX RAM offset in the table above corresponds directly to the RAM address. The RX RAM offset in the table above defines the distance from the beginning of the frame in the FIFO.

9.1 Register Summary

A list of registers with their offset from the base address (0x10220) is given below. All registers and buffer locations can only be accessed via Byte (8-bit), Word (16-bit) or Double-Word (32-bit). However, the FIFO must be accessed in Double-Word mode only.

Copyright © Bridgetek Pte Ltd

Address Offset	Register	Default Value	References
0x00	ETH_INT_STATUS – Interrupt Status Register	0x00	Section 9.2.1
0x01	ETH_INT_ENABLE – Interrupt Enable Register	0x00	Section 9.2.2
0x02	ETH_RX_CNTL – Receive Control Register	0x00	Section 9.2.3
0x03	ETH_TX_CNTL – Transmit Control Register	0x00	Section 9.2.4
0x04	ETH_DATA – Data Register (octet n)	0x00	Section 9.2.5
0x05	ETH_DATA – Data Register (octet n+1)	0x00	Section 9.2.6
0x06	ETH_DATA – Data Register (octet n+2)	0x00	Section 9.2.7
0x07	ETH_DATA – Data Register (octet n+3)	0x00	Section 9.2.8
0x08	ETH_ADDR – Address Register (octet 1)	0x00	Section 9.2.9
0x09	ETH_ADDR – Address Register (octet 2)	0x00	Section 9.2.10
0x0A	ETH_ADDR – Address Register (octet 3)	0x00	Section 9.2.11
0x0B	ETH_ADDR – Address Register (octet 4)	0x00	Section 9.2.12
0x0C	ETH_ADDR – Address Register (octet 5)	0x00	Section 9.2.13
0x0D	ETH_ADDR – Address Register (octet 6)	0x00	Section 9.2.14
0x0E	ETH_THRESHOLD – Threshold Register	0x00	Section 9.2.15
0x0F	ETH_MNG_CNTL – Management Control Register	0x00	Section 9.2.16
0x10	ETH_MNG_DIV – Management Divider Register	0x00	Section 9.2.17
0x11	ETH_MNG_ADDR – Management Address Register	0x00	Section 9.2.18
0x12	ETH_MNG_TX0 – Management Transmit Data 0 Register	0x00	Section 9.2.19
0x13	ETH_MNG_TX1 – Management Transmit Data 1 Register	0x00	Section 9.2.20
0x14	ETH_MNG_RX0 – Management Receive Data 0 Register	0x00	Section 9.2.21
0x15	ETH_MNG_RX1 – Management Receive Data 1 Register	0x00	Section 9.2.22
0x16	ETH_NUM_PKT – Number of Packets Register	0x00	Section 9.2.23
0x17	ETH_TR_REQ – Transmission Request Register	0x00	Section 9.2.24

Table 9.2 – Overview of Ethernet Registers

Copyright © Bridgetek Pte Ltd

9.2 Register Details

9.2.1 ETH_INT_STATUS – Interrupt Status Register (address offset: 0x0)

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5	MD_INT	RW1C	1'b0	Set when a transaction on the MII management interface has completed successfully (either read or write). Write a 1 to clear the status flag.
4	RX_ERR	RW1C	1'b0	Set when an error on RX has been encountered. This occurs when the Ethernet PHY asserts RXER during frame reception (100 Mbps only), or the frame is not an integer number of octets and the FCS check failed (dribble bits in frame) – alignment error, or the frame has a wrong CRC, or the length/type field is inconsistent with the client data size. Write a 1 to clear the status flag.
3	FIFO_OV	RW1C	1'b0	Set when RX FIFO overrun is encountered. Write a 1 to clear the status flag.
2	TX_EMPTY	RW1C	1'b0	Set when a packet has been sent. Write a 1 to clear the status flag.
1	TX_ERR	RW1C	1'b0	Set when an error on TX has been encountered. This occurs when the Data Length field value stored in the TX RAM exceeds 2032 in which case the frame will not be sent when this condition is encountered, or the retransmission attempt limit (16) has failed during a truncated binary exponential back off process. Write a 1 to clear the status flag. The write data pointer is also reset in this case.
0	RX_INT	RAC	1'b0	Set when at least one packet is in the receiver's FIFO. This status flag will be cleared by hardware when there is no packet in the receiver FIFO

Table 9.3 – ETH_INT_STATUS – Interrupt Status Register

Note: The individual status will still be reflected even if the individual interrupt has been disabled. This allows polling by software.

9.2.2 ETH_INT_ENABLE – Interrupt Enable Register (address offset: 0x1)

Setting a bit in this register enables the interrupt. Clearing a bit in this register disables the interrupt.

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5	MD_INT_MASK	RW	1'b0	Mask for MD_INT interrupt
4	RX_ERR_MASK	RW	1'b0	Mask for RX_ERR interrupt
3	FIFO_OV_MASK	RW	1'b0	Mask for FIFO_OV interrupt
2	TX_EMPTY_MASK	RW	1'b0	Mask for TX_EMPTY interrupt
1	TX_ERR_MASK	RW	1'b0	Mask for TX_ERR interrupt
0	RX_MASK	RW	1'b0	Mask for RX_INT interrupt

Table 9.4 – ETH_INT_ENABLE – Interrupt Enable Register

Copyright © Bridgetek Pte Ltd

9.2.3 ETH_RX_CNTL – Receive Control Register (address offset: 0x02)

This register configures the receiver.

Bit	Name	Type	Default Value	Description
7:6	RX_MEM_SIZE	RO	2'b01	Memory size – 4096 Bytes
5	Reserved	-	-	-
4	RESET_FIFO	WO	1'b0	1: clears the receiver FIFO; should be done when software initialisation of MAC is needed. It is recommended to set RX_ENABLE 0 first. This bit is automatically cleared.
3	BAD_CRC	RW	1'b1	1: all frames with wrong CRC will be discarded; all valid frames with broadcast address FF-FF-FF-FF-FF-FF in the Destination Address field are captured. 0: Do not drop frames with the wrong CRC. RX_ERR will still be flagged to indicate an erroneous packet has been received. The erroneous packet should be read out to make place for subsequent packets.
2	PRMS_MODE	RW	1'b0	1: Promiscuous mode enabled; all valid frames regardless of destination address will be captured, including multicast and broadcast frames. 0: Promiscuous mode disabled.
1	ACC_MULTI	RW	1'b0	1: Accept multicast; accepts all frames which have first bit of Destination Address set.
0	RX_ENABLE	RW	1'b0	1: enable frame receiver 0: disable frame receiver

Table 9.5 – ETH_RX_CNTL – Receive Control Register

9.2.4 ETH_TX_CNTL – Transmit Control Register (address offset: 0x03)

This register configures the transmitter.

Bit	Name	Type	Default Value	Description
7:6	TX_MEM_SIZE	RO	2'h0	Memory size – 2048 Bytes
5	Reserved	-	-	-
4	DUPLEX_MODE	RW	1'b0	1: enable duplex mode for Ethernet transmitter 0: disable duplex mode for Ethernet transmitter
3	Reserved	-	-	-
2	CRC_ENABLE	RW	1'b0	1: frames will be sent with CRC appended 0: frames will be sent without CRC appended
1	PAD_ENABLE	RW	1'b0	1: padding will be appended to frames shorter than the minimum frame size 0: padding will not be appended to frames shorter than the minimum frame size
0	TX_ENABLE	RW	1'b0	1: enable transmitter 0: disable transmitter

Table 9.6 – ETH_TX_CNTL – Transmit Control Register

9.2.5 ETH_DATA_N0 – Data Register (octet n) (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	For read/write from/to data buffer (RX/TX RAM)

Table 9.7 – ETH_DATA_N0 – Data Register (octet n)

Copyright © Bridgetek Pte Ltd

9.2.6 ETH_DATA_N1 – Data Register (octet n+1) (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	For read/write from/to data buffer (RX/TX RAM)

Table 9.8 – ETH_DATA_N1 – Data Register (octet n+1)

9.2.7 ETH_DATA_N2 – Data Register (octet n+2) (address offset: 0x06)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	For read/write from/to data buffer (RX/TX RAM)

Table 9.9 – ETH_DATA_N2 – Data Register (octet n+2)

9.2.8 ETH_DATA_N3 – Data Register (octet n+3) (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	For read/write from/to data buffer (RX/TX RAM)

Table 9.10 – ETH_DATA_N3 – Data Register (octet n+3)

9.2.9 ETH_ADDR_1 – Address Register (octet 1) (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:0	ADDRESS	RW	8'h00	MAC hardware address octet

Table 9.11 – ETH_ADDR_1 – Address Register (octet 1)

9.2.10 ETH_ADDR_2 – Address Register (octet 2) (address offset: 0x09)

Bit	Name	Type	Default Value	Description
7:0	ADDRESS	RW	8'h00	MAC hardware address octet

Table 9.12 – ETH_ADDR_2 – Address Register (octet 2)

9.2.11 ETH_ADDR_3 – Address Register (octet 3) (address offset: 0x0A)

Bit	Name	Type	Default Value	Description
7:0	ADDRESS	RW	8'h00	MAC hardware address octet

Table 9.13 – ETH_ADDR_3 – Address Register (octet 3)

9.2.12 ETH_ADDR_4 – Address Register (octet 4) (address offset: 0x0B)

Bit	Name	Type	Default Value	Description
7:0	ADDRESS	RW	8'h00	MAC hardware address octet

Table 9.14 – ETH_ADDR_4 – Address Register (octet 4)

9.2.13 ETH_ADDR_5 – Address Register (octet 5) (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	ADDRESS	RW	8'h00	MAC hardware address octet

Table 9.15 – ETH_ADDR_5 – Address Register (octet 5)

9.2.14 ETH_ADDR_6 – Address Register (octet 6) (address offset: 0x0D)

Bit	Name	Type	Default Value	Description
7:0	ADDRESS	RW	8'h00	MAC hardware address octet

Table 9.16 – ETH_ADDR_6 – Address Register (octet 6)

Copyright © Bridgetek Pte Ltd

9.2.15 ETH_THRESHOLD – Threshold Register (address offset: 0x0E)

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5:0	THRESHOLD	RW	6'h3F	This threshold specifies the threshold level for the TX RAM to begin transmission. When the byte count of the data in the TX RAM reaches this level, the transmission will start. Transmission starts when: Number of bytes written >= 4 * (THRESHOLD * 8 + 1)

Table 9.17 – ETH_THRESHOLD – Threshold Register

9.2.16 ETH_MNG_CNTL – Management Control Register (address offset: 0x0F)

This register is used to send management frames from the STA entity across the MII management interface.

Bit	Name	Type	Default Value	Description
7:3	REG_ADDRESS	RW	5'h00	These set the MII register address for the next transaction.
2	Reserved	-	-	-
1	WRITE	RW	1'b0	This bit should be updated together with the START bit. 1: perform write transaction 0: perform read transaction
0	START	RW	1'b0	Setting this bit to 1 will initiate the transaction. Hardware clears this bit after the transaction is complete.

Table 9.18 – ETH_MNG_CNTL – Management Control Register

9.2.17 ETH_MNG_DIV – Management Divider Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7:0	DIV	RW	8'h80	This is used to set the clock divider for the MDC clock used by the STA to clock transactions between PHY and MAC across the serial MII interface. The MDC clock is derived as follows: $Fmdc = \frac{Fclk}{2 * (MDVR + 1)}$

Table 9.19 – ETH_MNG_DIV – Management Divider Register

9.2.18 ETH_MNG_ADDR – Management Address Register (address offset: 0x11)

Bit	Name	Type	Default Value	Description
7:5	Reserved	-	-	-
4:0	PHY_ADDRESS	RW	5'h00	This register should be updated with the PHY address.

Table 9.20 – ETH_MNG_ADDR – Management Address Register

Copyright © Bridgetek Pte Ltd

9.2.19 ETH_MNG_TX0 – Management Transmit Data 0 Register (address offset: 0x12)

Bit	Name	Type	Default Value	Description
7:0	TX_LSB	RW	8'h00	This is the lower byte of a word of data to be sent across the MII management interface to the PHY during the next data transmission.

Table 9.21 – ETH_MNG_TX0 – Management Transmit Data 0 Register

9.2.20 ETH_MNG_TX1 – Management Transmit Data 1 Register (address offset: 0x13)

Bit	Name	Type	Default Value	Description
7:0	TX_MSB	RW	8'h00	This is the upper byte of a word of data to be sent across the MII management interface to the PHY during the next data transmission.

Table 9.22 – ETH_MNG_TX1 – Management Transmit Data 1 Register

9.2.21 ETH_MNG_RX0 – Management Receive Data 0 Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7:0	RX_LSB	RW	8'h00	This is the lower byte of a word of data read by the STA management entity from the PHY during the last transaction.

Table 9.23 – ETH_MNG_RX0 – Management Receive Data 0 Register

9.2.22 ETH_MNG_RX1 – Management Receive Data 1 Register (address offset: 0x15)

Bit	Name	Type	Default Value	Description
7:0	RX_MSB	RW	8'h00	This is the upper byte of a word of data read by the STA management entity from the PHY during the last transaction.

Table 9.24 – ETH_MNG_RX1 – Management Receive Data 1 Register

9.2.23 ETH_NUM_PKT – Number of Packets Register (address offset: 0x16)

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5:0	NUM_PKT	RO	6'h00	This is the number of packets in the receive FIFO. When NPR is greater than 0, RXINT interrupt will be active

Table 9.25 – ETH_NUM_PKT – Number of Packets Register

9.2.24 ETH_TR_REQ – Transmission Request Register (address offset: 0x17)

Bit	Name	Type	Default Value	Description
7:1	Reserved	-	-	-
0	NEW_TX	RW1S	1'b0	1: Send the new frame in transmit memory. Hardware clears this bit when the transmission is complete, or an error on TX is encountered.

Table 9.26 – ETH_TR_REQ – Transmission Request Register

Copyright © Bridgetek Pte Ltd

10 CAN Bus Controller

Two CAN bus controllers are supported by this device. The controllers have the following features.

- Conforms to Bosch CAN 2.0B specification
- Data rate up to 1Mbps
- Hardware message filtering (dual/single filters)
- 64-byte receive FIFO
- 16-byte transmit buffer
- No overload frames are generated
- Normal & Listen Only modes supported
- Single Shot transmission
- Ability to abort transmission
- Readable error counters
- Last Error Code

Listed below are the symbols used in the CAN Frame buffer.

Symbol	Description
FF	Frame Format
RTR	Remote Request bit 1: Remote Frames 0: Data Frames
X	Don't care
DLC	Data Length Code
ID	CAN Message identifier
DATA1 – DATA8	Data bytes

Table 10.1 – Symbols used in the CAN Frame buffer

The memory buffer layout for standard frames is:

Offset	Data Bits	Standard Frame buffer content for TX/RX RAM							
0x00	7:0	FF	RTR	X/0	X/0	DLC3	DLC2	DLC1	DLC0
	15:8	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3
	23:16	ID2	ID1	ID0	X/RTR	X/0	X/0	X/0	X/0
	31:24	DATA1							
0x01	7:0	DATA2							
	15:8	DATA3							
	23:16	DATA4							
	31:24	DATA5							
0x02	7:0	DATA6							
	15:8	DATA7							
	23:16	DATA8							
	31:24	UNUSED							
0x03	7:0	UNUSED							
	15:8	UNUSED							
	23:16	UNUSED							
	31:24	UNUSED							

Table 10.2 – Standard Frames Memory Buffer Layout

Copyright © Bridgetek Pte Ltd

The memory buffer layout for extended frames is:

Offset	Data Bits	Extended Frame buffer content for TX/RX RAM							
0x00	7:0	FF	RTR	X/0	X/0	DLC3	DLC2	DLC1	DLC0
	15:8	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21
	23:16	ID20	ID19	ID18	ID17	ID16	ID15	ID14	ID13
	31:24	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5
0x01	7:0	ID4	ID3	ID2	ID1	ID0	X/RTR	X/0	X/0
	15:8	DATA1							
	23:16	DATA2							
	31:24	DATA3							
0x02	7:0	DATA4							
	15:8	DATA5							
	23:16	DATA6							
	31:24	DATA7							
0x03	7:0	DATA8							
	15:8	UNUSED							
	23:16	UNUSED							
	31:24	UNUSED							

Table 10.3 – Extended Frames Memory Buffer Layout

10.1 Register Summary

Listed below are the registers with their offset from the base addresses (0x10240 for CAN0 and 0X10260 for CAN1). All registers and buffer locations can only be accessed via Byte (8-bit) mode.

Address Offset	Register	Default value	References
0x00	CAN_MODE – Mode Register	0x04	Section 10.2.1
0x01	CAN_CMD – Command Register	0x00	Section 10.2.2
0x02	CAN_STATUS – Status Register	0x20	Section 10.2.3
0x03	CAN_INT_STATUS – Interrupt Status Register	0x00	Section 10.2.4
0x04	CAN_INT_ENABLE – Interrupt Enable Register	0x00	Section 10.2.5
0x05	CAN_RX_MSG – Receive Message Register	0x00	Section 10.2.6
0x06	CAN_BUS_TIM_0 – Bus Timing 0 Register	0x00	Section 10.2.7
0x07	CAN_BUS_TIM_1 – Bus Timing 1 Register	0x00	Section 10.2.8
0x08	CAN_TX_BUF_0 – Transmit Buffer 0 Register	0x00	Section 10.2.9.1
0x09	CAN_TX_BUF_1 – Transmit Buffer 1 Register	0x00	Section 10.2.9.2
0x0A	CAN_TX_BUF_2 – Transmit Buffer 2 Register	0x00	Section 10.2.9.3
0x0B	CAN_TX_BUF_3 – Transmit Buffer 3 Register	0x00	Section 10.2.9.4
0x0C	CAN_RX_BUF_0 – Receive Buffer 0 Register	0x00	Section 10.2.10.1
0x0D	CAN_RX_BUF_1 – Receive Buffer 1 Register	0x00	Section 10.2.10.2
0x0E	CAN_RX_BUF_2 – Receive Buffer 2 Register	0x00	Section 10.2.10.3

Copyright © Bridgetek Pte Ltd

0x0F	CAN_RX_BUF_3 – Receive Buffer 3 Register	0x00	Section 10.2.10.4
0x10	CAN_ACC_CODE_0 – Acceptance Code 0 Register	0x00	Section 10.2.11.1
0x11	CAN_ACC_CODE_1 – Acceptance Code 1 Register	0x00	Section 10.2.11.2
0x12	CAN_ACC_CODE_2 – Acceptance Code 2 Register	0x00	Section 10.2.11.3
0x13	CAN_ACC_CODE_3 – Acceptance Code 3 Register	0x00	Section 10.2.11.4
0x14	CAN_ACC_MASK_0 – Acceptance Mask 0 Register	0x00	Section 10.2.11.5
0x15	CAN_ACC_MASK_1 – Acceptance Mask 1 Register	0x00	Section 10.2.11.6
0x16	CAN_ACC_MASK_2 – Acceptance Mask 2 Register	0x00	Section 10.2.11.7
0x17	CAN_ACC_MASK_3 – Acceptance Mask 3 Register	0x00	Section 10.2.11.8
0x18	CAN_ERR_CODE – Error Code Capture Register	0x00	Section 10.2.12
0x19	CAN_RX_ERR_CNTR – Receive Error Counter Register	0x00	Section 10.2.13
0x1A	CAN_TX_ERR_CNTR – Transmit Error Counter Register	0x00	Section 10.2.14
0x1B	CAN_ARB_LOST_CODE – Arbitration Lost Code Capture Register	0x00	Section 10.2.15

Table 10.4 – Overview of CAN Registers

10.2 Register Details

10.2.1 CAN_MODE – Mode Register (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7:3	Reserved	-	-	-
2	RST	RW	1'b1	Reset Mode 1: Controller in reset mode; no frame reception and transmission are possible. This mode is used to configure the controller hardware.
1	LSTN_ONLY	RW	1'b0	Listen Only Mode. Updatable only when RST is 1. 0: Normal Mode 1: Listen only mode
0	ACC_FLTR	RW	1'b0	Hardware Acceptance Filter Scheme Updatable only when RST is 1 0: dual filter is used for hardware acceptance filter scheme 1: single filter is used for hardware acceptance filter scheme

Table 10.5 – CAN_MODE – Mode Register

Copyright © Bridgetek Pte Ltd

10.2.2 CAN_CMD – Command Register (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7:3	Reserved	-	-	-
2	TX_REQ	W1S	1'b0	Transmit Request. 1: Initiates frames transmission by Bit Stream Processor 0: No effect
1	ABORT_TX	WO	1'b0	Abort Transmission. When writing 1 to this bit simultaneously with TX_REQ bit, 1-shot transmission is performed. If TX_REQ has been set to 1, and the transmission has not started, writing a 1 to ABORT_TX will abort the request.
0	Reserved	-	-	-

Table 10.6 – CAN_CMD – Command Register

10.2.3 CAN_STATUS – Status Register (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7	RX_BUF_STS	RO	1'b0	Receive Buffer Status: 1: at least one message is in RX FIFO. 0: no message is in RX FIFO
6	OVRN_STS	RO	1'b0	Data Overrun Status 1: when RX FIFO encounters overrun. 0: no overrun occurred since the last clear data overrun command. This bit is cleared when a 1 is written to DATA_OVRN bit.
5	TX_BUF_STS	RO	1'b1	Transmit Buffer Status 1: transmit buffer can be written. 0: transmission in progress and transmit buffer is locked such that no data write can be accepted.
4	DOVLD_STS	RO	1'b0	Data Overload Status 1: the overload condition occurs due to the internal conditions of the receiver, which requires a delay of the next frame
3	RX_STS	RO	1'b0	Receive Status 1: when the CAN core is receiving a message.
2	TX_STS	RO	1'b0	Transmit Status 1: when the CAN core is transmitting a message.
1	ERR_STS	RO	1'b0	Error Status 1: when at least one of CAN error counters has reached error warning limit (96). This bit is automatically cleared when the error counter is below the limit.
0	BUS_OFF_STS	RO	1'b0	Bus Off Status 1: Node is in bus off state and cannot transmit and receive frames. When the transmit error counter exceeds the limit of 255, this bit will be set to 1, the CAN controller sets reset mode, and if enabled, an error warning interrupt is generated. The transmit error counter is then set to 127 and

Copyright © Bridgetek Pte Ltd

				receive error counter is cleared. The CAN controller stays in reset mode until the CPU clears the reset mode bit. Once this is completed, the CAN controller waits for 128 occurrences of the bus free signal (11 consecutive recessive bits) counting down the transmit error counter. After that, this bit is cleared (bus on), the error counters are reset and the error warning interrupt is generated if enabled.

Table 10.7 – CAN_STATUS – Status Register

10.2.4 CAN_INT_STATUS – Interrupt Status Register (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7	DOVLDI	RW1C	1'b0	Data Overload Interrupt Set when the overload condition occurs. Write a 1 to clear this interrupt
6	ARB_LOST	RW1C	1'b0	Arbitration Lost Interrupt Set when the CAN core has lost arbitration during transmission of its own message and become a receiver. Write a 1 to clear this interrupt.
5	ERR_WRNG	RW1C	1'b0	Error Warning Interrupt Set when there is a change in ERR_STS or BUS_OFF_STS bits of Status register. Write a 1 to clear this interrupt.
4	ERR_PSV	RW1C	1'b0	Error Passive Interrupt Set when CAN core has reached or exceeded error passive level. Write a 1 to clear this interrupt.
3	RX	RW1C	1'b0	Receive Interrupt Set when there is at least one message in the RX FIFO. Write a 1 to decrement the RX message counter (NUM_FRM). NUM_FRM is not decremented automatically. Note that this flag is cleared after the first message is read. The interrupt handler should check how many messages there are (NUM_FRM) and read all of them out.
2	TX	RW1C	1'b0	Transmission Interrupt Set after a successful transmission. Write a 1 to reset the write pointer to TX RAM before writing the next frame of data.
1	BUS_ERR	RW1C	1'b0	Bus Error Interrupt Set when the CAN core encounters a bus error while transmitting or receiving a message. Write a 1 to clear this interrupt.
0	DATA_OVRN	RW1C	1'b0	Data Overrun Interrupt Set when the RX FIFO overrun has occurred. Write a 1 to clear this interrupt.

Table 10.8 – CAN_INT_STATUS – Interrupt Status Register

Copyright © Bridgetek Pte Ltd

10.2.5 CAN_INT_ENABLE – Interrupt Enable Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7	DOVLDI_EN	RW	1'b0	Mask for DOVLD interrupt.
6	ARB_LOST_EN	RW	1'b0	Mask for ARB_LOST interrupt.
5	ERR_WRNG_EN	RW	1'b0	Mask for ERR_WRNG interrupt.
4	ERR_PSV_EN	RW	1'b0	Mask for ERR_PSV interrupt.
3	RX_EN	RW	1'b0	Mask for RX interrupt.
2	TX_EN	RW	1'b0	Mask for TX interrupt.
1	BUS_ERR_EN	RW	1'b0	Mask for BUS_ERR interrupt.
0	DATA_OVRN_EN	RW	1'b0	Mask for DATA_OVRN interrupt.

Table 10.9 – CAN_INT_ENABLE – Interrupt Enable Register

10.2.6 CAN_RX_MSG – Receive Message Register (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7:5	Reserved	-	-	-
4:0	NUM_FRM	RO	5'h00	Number of stored message frames This shows the number of frames stored in the RX FIFO. The value is incremented on each successful frame reception and decremented by clearing the RX interrupt. Up to 21 messages can be stored. This equation shows the calculation: $n = \frac{64}{3 + data_length_code}$

Table 10.10 – CAN_RX_MSG – Receive Message Register

10.2.7 CAN_BUS_TIM_0 – Bus Timing 0 Register (address offset: 0x06)

Bit	Name	Type	Default Value	Description
7:6	SYNC_JMP_WDT	RW	2'h0	Synchronisation Jump Width This allows compensation for phase shifts between clocks of different bus controllers. The maximum number of clock cycles a bit period may be changed by one resynchronisation is defined by SYNC_JMP_WDT as: $T_{SYNC_JMP_WDT}: t_{sclk} \, x \, (2 \, x \, SYNC_JMP_WDT[1] + SYNC_JMP_WDT[0] + 1)$
5:0	BAUD_PSCL	RW	6'h00	Baud Rate Pre-scaler Baud rate can be set using this equation: $BAUD_PSCL = (32 \, x \, BAUL_PSCL[5] + 16 \, x \, BAUD_PSCL[4] + 8 \, x \, BAUD_PSCL[3] + 4 \, x \, BAUD_PSCL[2] + 2 \, x \, BAUD_PSCL[1] + BAUD_PSCL[0])$ The period of CAN system clock tsclk is thus: $t_{sclk} = 2 \, x \, t_{clk} \, x \, BAUD_PSCL$ where t_{clk} is 10ns based on a 100MHz system clock.

Table 10.11 – CAN_BUS_TIM_0 – Bus Timing 0 Register

Copyright © Bridgetek Pte Ltd

10.2.8 CAN_BUS_TIM_1 – Bus Timing 1 Register (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7	NUM_SAM	RW	1'b0	Number of bus level samples 0: bus level is sampled once (recommended for high-speed buses) 1: bus level is sampled three times (recommended for low/medium speed buses where there is a benefit from filtering spikes)
6:4	TIM_SEG2	RW	3'h0	Number of clock cycles per Time Segment 2 $t_{TIM_SEG2} = t_{sclk} \times (4 \times TIM_SEG1[2] + 2 \times TIM_SEG1[1] + TIM_SEG1[0] + 1)$
3:0	TIM_SEG1	RW	4'h0	Number of clock cycles per Time Segment 1 $t_{TIM_SEG1} = t_{sclk} \times (8 \times TIM_SEG1[3] + 4 \times TIM_SEG1[2] + 2 \times TIM_SEG1[1] + TIM_SEG1[0] + 1)$

Table 10.12 – CAN_BUS_TIM_1 – Bus Timing 1 Register

10.2.9 CAN_TX_BUF – Transmit Buffer Register

The Transmit Buffer Register (CAN_TX_BUF) is used to write a CAN frame which will be sent over the CAN network. It is a write-only register. This register is mapped into four consecutive byte locations starting at offset 0x08. Users can access these locations through registers CAN_TX_BUF_0 to CAN_TX_BUF_3. The byte at location CAN_TX_BUF_0 is least significant, while the byte at location CAN_TX_BUF_3 is most significant.

Writing to the Transmit Buffer Register performs auto increment of the internal write pointer. This pointer is the actual address to the CAN TX RAM. Auto increment is executed only when location CAN_TX_BUF_3 is accessed.

The write pointer can be reset by writing a 1 to bit TX of the Interrupt Status Register. Refer to section 11.2.4 for further information.

10.2.9.1 CAN_TX_BUF_0 – Transmit Buffer 0 Register (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	This is used to write a CAN frame for transmission. When write is performed on CAN_TX_BUF_3, the internal write pointer will be automatically incremented. This pointer can be reset by writing a 1 to TX of the Interrupt Status register.

Table 10.13 – CAN_TX_BUF_0 – Transmit Buffer 0 Register

10.2.9.2 CAN_TX_BUF_1 – Transmit Buffer 1 Register (address offset: 0x09)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	This is used to write a CAN frame for transmission. When write is performed on CAN_TX_BUF_3, the internal write pointer will be automatically incremented. This pointer can be reset by writing a 1 to TX of the Interrupt Status register.

Table 10.14 – CAN_TX_BUF_1 – Transmit Buffer 1 Register

Copyright © Bridgetek Pte Ltd

10.2.9.3 CAN_TX_BUF_2 – Transmit Buffer 2 Register (address offset: 0x0A)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	This is used to write a CAN frame for transmission. When write is performed on CAN_TX_BUF_3, the internal write pointer will be automatically incremented. This pointer can be reset by writing a 1 to TX of the Interrupt Status register.

Table 10.15 – CAN_TX_BUF_2 – Transmit Buffer 2 Register

10.2.9.4 CAN_TX_BUF_3 – Transmit Buffer 3 Register (address offset: 0x0B)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	This is used to write a CAN frame for transmission. When write is performed on CAN_TX_BUF_3, the internal write pointer will be automatically incremented. This pointer can be reset by writing a 1 to TX of the Interrupt Status register.

Table 10.16 – CAN_TX_BUF_3 – Transmit Buffer 3 Register

10.2.10 CAN_RX_BUF – Receive Buffer Register

The Receive Buffer Register (CAN_RX_BUF) is used to read CAN frames received from the CAN network. It is a read-only register. This register is mapped into four consecutive byte locations starting at offset 0x0C. The users can access these locations through registers CAN_RX_BUF_0 to CAN_RX_BUF_3. The byte at location CAN_RX_BUF_0 is least significant, while the byte at location CAN_RX_BUF_3 is most significant.

Reading the Receive Buffer Register performs auto increment of the internal read pointer. This pointer is the actual address to the CAN RX FIFO. Auto increment is executed only when location CAN_RX_BUF_3 is accessed.

10.2.10.1 CAN_RX_BUF_0 – Receive Buffer 0 Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	This is used to read a received CAN frame. When read is performed on CAN_RX_BUF_3, the internal read pointer will be automatically incremented.

Table 10.17 – CAN_RX_BUF_0 – Receive Buffer 0 Register

10.2.10.2 CAN_RX_BUF_1 – Receive Buffer 1 Register (address offset: 0x0D)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	This is used to read a received CAN frame. When read is performed on CAN_RX_BUF_3, the internal read pointer will be automatically incremented.

Table 10.18 – CAN_RX_BUF_1 – Receive Buffer 1 Register

10.2.10.3 CAN_RX_BUF_2 – Receive Buffer 2 Register (address offset: 0x0E)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	This is used to read a received CAN frame. When read is performed on CAN_RX_BUF_3, the internal read pointer will be automatically incremented.

Table 10.19 – CAN_RX_BUF_2 – Receive Buffer 2 Register

Copyright © Bridgetek Pte Ltd

10.2.10.4 CAN_RX_BUF_3 – Receive Buffer 3 Register (address offset: 0x0F)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	This is used to read a received CAN frame. When read is performed on CAN_RX_BUF_3, the internal read pointer will be automatically incremented.

Table 10.20 – CAN_RX_BUF_3 – Receive Buffer 3 Register

10.2.11 CAN Acceptance Filter

The acceptance filter makes it possible to pass received messages to the RX FIFO only when the identifier bits of the received message are equal to the predefined ones within the acceptance filter registers.

The acceptance filter is defined by the acceptance code registers (CAN_ACC_CODE_3:CAN_ACC_CODE_0) and acceptance mask registers (CAN_ACC_MASK_3:CAN_ACC_MASK_0).

The acceptance code registers contain bit patterns of messages to be received while the corresponding acceptance mask registers define which bit positions will be compared and which ones are don't care. Writing a 1 to a certain bit in CAN_ACC_MASK_x defines the corresponding bit in CAN_ACC_CODE_x as 'don't care'.

The figure below demonstrates the process. An "X" means that a bit in the CAN Acceptance Filter is used for filtering, whereas a minus "-" means that a bit in the CAN Acceptance Filter is ignored:

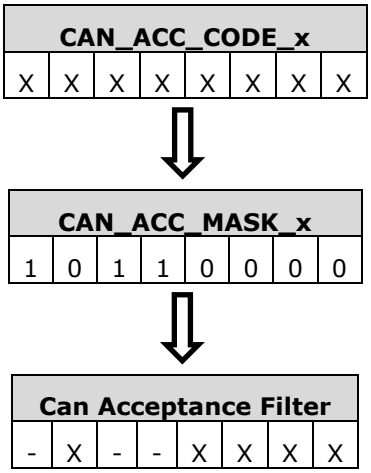


Figure 10.1 – CAN Acceptance Filter

The ACC_FLTR bit of the Mode Register (CAN_MODE) sets one of two available filter configurations: single or dual filter.

Single Filter Configuration:

In single filter configuration, one long filter can be defined (four bytes). If a standard frame message is received, the complete identifier including RTR bit and the first two data bytes (if received) are used for acceptance filtering. Messages may also be accepted if there is no data byte. If only one data byte is received, then only bits up to this data byte are compared with the filter. All single bit comparisons have to signal acceptance for successful reception of message.

If an extended frame message is received while in single filter configuration, the complete identifier including the RTR bit is used for acceptance filtering. Again, for successful reception, all single bit comparisons have to signal acceptance.

Dual Filter Configuration:

Copyright © Bridgetek Pte Ltd

In dual filter configuration, two short filters can be defined. The received message is compared with both filters to decide whether this message should be stored in the RX FIFO. If at least one of the messages filters signals acceptance, the message is accepted.

If a standard frame message is received, the first filter compares the complete standard identifier including the RTR bit and the first data byte of the message. The second filter only compares the complete standard identifier including the RTR bit. For successful reception of a message, all single bit comparisons of at least one filter must signal acceptance. If no data byte is required for filter 1, the four least significant bits of AMR1 and AMR3 have to be set to 1 (don't care).

If an extended frame message is received while in dual filter mode, both filters are comparing the first two bytes of the extended identifier range only. For a successful reception, all single bit comparisons of at least one filter have to signal acceptance.

The contents of the acceptance code registers (CAN_ACC_CODE_3:CAN_ACC_CODE_0), depending on the filter configuration and the frame type, is presented below. Refer to Table 10.2 and Table 10.3 for the bits in a CAN message. For simplicity, the tables refer to the different settings as:

- Setting 1: Single filter configuration, standard frame message
- Setting 2: Single filter configuration, extended frame message
- Setting 3: Dual filter configuration, standard frame message
- Setting 4: Dual filter configuration, extended frame message

A particular bit in CAN_ACC_CODE_x can always be ignored by setting the corresponding bit in CAN_ACC_MASK_x to 1. It is recommended to set the mask to 1 for the unused bits in CAN_ACC_CODE_x.

10.2.11.1 CAN_ACC_CODE_0 – Acceptance Code 0 Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description			
				Setting 1	Setting 2	Setting 3	Setting 4
7	ACC_CODE	RW	1'b0	ID10	ID28	ID10	ID28
6	ACC_CODE	RW	1'b0	ID9	ID27	ID9	ID27
5	ACC_CODE	RW	1'b0	ID8	ID26	ID8	ID26
4	ACC_CODE	RW	1'b0	ID7	ID25	ID7	ID25
3	ACC_CODE	RW	1'b0	ID6	ID24	ID6	ID24
2	ACC_CODE	RW	1'b0	ID5	ID23	ID5	ID23
1	ACC_CODE	RW	1'b0	ID4	ID22	ID4	ID22
0	ACC_CODE	RW	1'b0	ID3	ID21	ID3	ID21

Table 10.21- CAN_ACC_CODE_0 – Acceptance Code 0 Register

10.2.11.2 CAN_ACC_CODE_1 – Acceptance Code 1 Register (address offset: 0x11)

Bit	Name	Type	Default Value	Description			
				Setting 1	Setting 2	Setting 3	Setting 4
7	ACC_CODE	RW	1'b0	ID2	ID20	ID2	ID20
6	ACC_CODE	RW	1'b0	ID1	ID19	ID1	ID19
5	ACC_CODE	RW	1'b0	ID0	ID18	ID0	ID18
4	ACC_CODE	RW	1'b0	RTR	ID17	RTR	ID17
3	ACC_CODE	RW	1'b0	Unused	ID16	DATA1[7]	ID16
2	ACC_CODE	RW	1'b0	Unused	ID15	DATA1[6]	ID15
1	ACC_CODE	RW	1'b0	Unused	ID14	DATA1[5]	ID14
0	ACC_CODE	RW	1'b0	Unused	ID13	DATA1[4]	ID13

Table 10.22 – CAN_ACC_CODE_1 – Acceptance Code 1 Register

Copyright © Bridgetek Pte Ltd

10.2.11.3 CAN_ACC_CODE_2 – Acceptance Code 2 Register (address offset: 0x12)

Bit	Name	Type	Default Value	Description			
				Setting 1	Setting 2	Setting 3	Setting 4
7	ACC_CODE	RW	1'b0	DATA1[7]	ID12	ID10	ID28
6	ACC_CODE	RW	1'b0	DATA1[6]	ID11	ID9	ID27
5	ACC_CODE	RW	1'b0	DATA1[5]	ID10	ID8	ID26
4	ACC_CODE	RW	1'b0	DATA1[4]	ID9	ID7	ID25
3	ACC_CODE	RW	1'b0	DATA1[3]	ID8	ID6	ID24
2	ACC_CODE	RW	1'b0	DATA1[2]	ID7	ID5	ID23
1	ACC_CODE	RW	1'b0	DATA1[1]	ID6	ID4	ID22
0	ACC_CODE	RW	1'b0	DATA1[0]	ID5	ID3	ID21

Table 10.23 - CAN_ACC_CODE_2 – Acceptance Code 2 Register

10.2.11.4 CAN_ACC_CODE_3 – Acceptance Code 3 Register (address offset: 0x13)

Bit	Name	Type	Default Value	Description			
				Setting 1	Setting 2	Setting 3	Setting 4
7	ACC_CODE	RW	1'b0	DATA2[7]	ID4	ID2	ID20
6	ACC_CODE	RW	1'b0	DATA2[6]	ID3	ID1	ID19
5	ACC_CODE	RW	1'b0	DATA2[5]	ID2	ID0	ID18
4	ACC_CODE	RW	1'b0	DATA2[4]	ID1	RTR	ID17
3	ACC_CODE	RW	1'b0	DATA2[3]	ID0	DATA1[3]	ID16
2	ACC_CODE	RW	1'b0	DATA2[2]	RTR	DATA1[2]	ID15
1	ACC_CODE	RW	1'b0	DATA2[1]	Unused	DATA1[1]	ID14
0	ACC_CODE	RW	1'b0	DATA2[0]	Unused	DATA1[0]	ID13

Table 10.24 – CAN_ACC_CODE_3 – Acceptance Code 3 Register

10.2.11.5 CAN_ACC_MASK_0 – Acceptance Mask 0 Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	This register determines which bits in CAN_ACC_CODE_0 is used for the acceptance filter. A 1 at a particular bit means that the corresponding bit in CAN_ACC_CODE_0 will not be compared.

Table 10.25 – CAN_ACC_MASK_0 – Acceptance Mask 0 Register

10.2.11.6 CAN_ACC_MASK_1 – Acceptance Mask 1 Register (address offset: 0x15)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	This register determines which bits in CAN_ACC_CODE_1 is used for the acceptance filter. A 1 at a particular bit means that the corresponding bit in CAN_ACC_CODE_1 will not be compared.

Table 10.26 – CAN_ACC_MASK_1 – Acceptance Mask 1 Register

10.2.11.7 CAN_ACC_MASK_2 – Acceptance Mask 2 Register (address offset: 0x16)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	This register determines which bits in CAN_ACC_CODE_2 is used for the acceptance filter. A 1 at a particular bit means that the corresponding bit in CAN_ACC_CODE_2 will not be compared.

Table 10.27 – CAN_ACC_MASK_2 – Acceptance Mask 2 Register

Copyright © Bridgetek Pte Ltd

10.2.11.8 CAN_ACC_MASK_3 – Acceptance Mask 3 Register (address offset: 0x17)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	This register determines which bits in CAN_ACC_CODE_3 is used for the acceptance filter. A 1 at a particular bit means that the corresponding bit in CAN_ACC_CODE_3 will not be compared.

Table 10.28 – CAN_ACC_MASK_3 – Acceptance Mask 3 Register

10.2.12 CAN_ERR_CODE – Error Code Capture Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
7	RX_WRN	RO	1'b0	Set when RX_ERR counter is greater than or equal to 96.
6	TX_WRN	RO	1'b0	Set when TX_ERR counter is greater than or equal to 96.
5	ERR_DIR	RO	1'b0	Direction of transfer when error occurred. 0: transmission 1: reception
4	ACK_ERR	RO	1'b0	ACK error occurred.
3	FRM_ERR	RO	1'b0	Form error occurred.
2	CRC_ERR	RO	1'b0	CRC error occurred.
1	STF_	RO	1'b0	Stuff error occurred.

Copyright © Bridgetek Pte Ltd

	ERR			
0	BIT_ERR	RO	1'b0	Bit error occurred.

Table 10.29 – CAN_ERR_CODE – Error Code Capture Register

10.2.13 CAN_RX_ERR_CNTR – Receive Error Counter Register (address offset: 0x19)

Bit	Name	Type	Default Value	Description
7:0	RX_ERR	RO	8'h00	This is the current receive error counter. If a bus off event occurs, it is initialized to 0.

Table 10.30 – CAN_RX_ERR_CNTR – Receive Error Counter Register

10.2.14 CAN_TX_ERR_CNTR – Transmit Error Counter Register (address offset: 0x1A)

Bit	Name	Type	Default Value	Description
7:0	TX_ERR	RO	8'h00	This is the low-byte of the current transmit error counter as the width of the transmit error counter is 9-bit. If a bus off event occurs, it is initialized to 127 to count the minimum protocol defined time (128 occurrences of bus free signal). Reading TX_ERR during this time gives information about the status of the bus off recovery.

Table 10.31 – CAN_TX_ERR_CNTR – Transmit Error Counter Register

10.2.15 CAN_ARB_LOST_CODE – Arbitration Lost Code Capture Register (address offset: 0x1B)

Bit	Name	Type	Default Value	Description
7:5	Reserved	-	-	-
4:0	ARB_CODE	RO	5'h00	This contains the bit position at which arbitration was lost during transmission of a message. This register is not updated until the previous arbitration lost interrupt has been acknowledged.

Table 10.32 – CAN_ARB_LOST_CODE – Arbitration Lost Code Capture Register

Copyright © Bridgetek Pte Ltd

11 SD Host

The device supports a SD Host with the following features.

- Compliant with SD host controller standard specification, version 3.0
- Compliant with SD physical layer specification, version 3.0
- Compliant with SDIO card specification, version 2.0
- Compliant with MMC card specification, version 4.3
- Supports PIO data transfers
- 4kB data FIFO
- 4-bit data bus, which supports 1-bit/4-bit data transfer
- Supports CPRM function for security
- Built-in 7-bit/16-bit CRC generation and checking
- Card detection (Insertion/Removal)
- Supports Read Wait mechanism for SDIO function

11.1 Register Summary

Listed below are the registers with their offset from the base address (0x10400). All registers can only be accessed via Double-Word (32-bit) mode. Note that some registers are not 32-bit long. In that case, several registers are combined into one 32-bit location. If one such register is accessed, all other registers in the same 32-bit location will also be affected. The users need to take care not to modify the content of the other registers.

Address Offset	Register	Default Value	References
0x00	SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register	0x00000000	Section 11.2.1
0x04	SDH_BLK_SIZE – Block Size Register	0x0000	Section 11.2.2
0x06	SDH_BLK_COUNT – Block Count Register	0x0000	Section 11.2.3
0x08	SDH_ARG_1 – Argument 1 Register	0x00000000	Section 11.2.4
0x0C	SDH_TNSFER_MODE – Transfer Mode Register	0x0000	Section 11.2.5
0x0E	SDH_CMD – Command Register	0x0000	Section 11.2.6
0x10 -0x1C	SDH_RESPONSE – Response Register	0x00000000	Section 11.2.7
0x20	SDH_BUF_DATA – Buffer Data Port Register	0x00000000	Section 11.2.8
0x24	SDH_PRESENT_STATE – Present State Register	0x00000000	Section 11.2.9
0x28	SDH_HST_CNTL_1 – Host Control 1 Register	0x00	Section 11.2.10
0x29	SDH_PWR_CNTL – Power Control Register	0x00	Section 11.2.11
0x2A	SDH_BLK_GAP_CNTL – Block Gap Control Register	0x0000	Section 11.2.12
0x2C	SDH_CLK_GAP_CNTL – Clock Control Register	0x0000	Section 11.2.13
0x2E	SDH_TIMEOUT_CNTL – Timeout Control Register	0x00	Section 11.2.14
0x2F	SDH_SW_RST – Software Reset Register	0x00	Section 11.2.15

Copyright © Bridgetek Pte Ltd

0x30	SDH_NRML_INT_STATUS – Normal Interrupt Status Register	0x0000	Section 11.2.16
0x32	SDH_ERR_INT_STATUS – Error Interrupt Status Register	0x0000	Section 11.2.17
0x34	SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register	0x0000	Section 11.2.18
0x36	SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register	0x0000	Section 11.2.19
0x38	SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register	0x0000	Section 11.2.20
0x3A	SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register	0x0000	Section 11.2.21
0x3C	SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register	0x0000	Section 11.2.22
0x3E	SDH_HOST_CNTL_2 – Host Control 2 Register	0x0000	Section 11.2.23
0x40	SDH_CAP_1 – Capabilities Register 1	0x016A0080	Section 11.2.24
0x44	SDH_CAP_2 – Capabilities Register 2	0x00000F75	Section 11.2.25
0x48	SDH_RSRV_1 – Reserved 1 Register	0x00000000	Section 11.2.26
0x4C	SDH_RSRV_2 – Reserved 2 Register	0x00000000	Section 11.2.27
0x50	SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event for Auto CMD Error Status Register	0x0000	Section 11.2.28
0x52	SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register	0x0000	Section 11.2.29
0x54	SDH_RSRV_3 – Reserved 3 Register	0x00000000	Section 11.2.30
0x58	SDH_RSRV_4 – Reserved 4 Register	0x00000000	Section 11.2.31
0x60	SDH_PRST_INIT – Preset value for initialization Register	0x0000	Section 11.2.32
0x62	SDH_PRST_DFLT_SPD – Preset value for default speed Register	0x0000	Section 11.2.33
0x64	SDH_PRST_HIGH_SPD – Preset value for the high-speed Register	0x0000	Section 11.2.34
0x66	SDH_PRST_SDR12 – Preset value for SDR12 Register	0x0000	Section 11.2.35
0x68	SDH_PRST_SDR25 – Preset value for SDR25 Register	0x0000	Section 11.2.36
0x6A	SDH_PRST_SDR50 – Preset value for SDR50 Register	0x0000	Section 11.2.37
0x6C	SDH_PRST_SDR104 – Preset value for SDR104 Register	0x0000	Section 11.2.38
0x6E	SDH_PRST_DDR50 – Preset value for DDR50 Register	0x0000	Section 11.2.39
0xFC	SDH_RSRV_5 – Reserved 5 Register	0x0000	Section 11.2.40
0xFE	SDH_HC_VER – Host Controller Version Register	0x0002	Section 11.2.41
0x100	SDH_VNDR_0 – Vendor defined 0 Register	0x00000001	Section 11.2.42
0x104	SDH_VNDR_1 – Vendor defined 1 Register	0x00000000	Section 11.2.43

Copyright © Bridgetek Pte Ltd

0x108	SDH_VNDR_2 – Vendor defined 2 Register	0x00000000	Section 11.2.44
0x10C	SDH_VNDR_3 – Vendor defined 3 Register	0x1F000000	Section 11.2.45
0x110	SDH_VNDR_4 – Vendor defined 4 Register	0x00000000	Section 11.2.46
0x114	SDH_VNDR_5 – Vendor defined 5 Register	0x00000000	Section 11.2.47
0x118	SDH_VNDR_6 – Vendor defined 6 Register	0x00000000	Section 11.2.48
0x11C	SDH_VNDR_7 – Vendor defined 7 Register	0x00000000	Section 11.2.49
0x120	SDH_VNDR_8 – Vendor defined 8 Register	0x00000000	Section 11.2.50
0x124	SDH_VNDR_9 – Vendor defined 9 Register	0x00000000	Section 11.2.51
0x128	SDH_RSRV_6 – Reserved 6 Register	0x00000000	Section 11.2.52
0x178	SDH_HW_ATTR – Hardware Attributes Register	0x00000150	Section 11.2.53
0x180	SDH_CPR_MOD_CNTL – Cipher Mode Control Register	0x00000000	Section 11.2.54
0x184	SDH_CPR_MOD_STATUS – Cipher Mode Status Register	0x00000000	Section 11.2.55
0x188	SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register	0x0000	Section 11.2.56
0x18A	SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register	0x0000	Section 11.2.57
0x18C	SDH_IN_DATA_LSB –Input Data LSB Register	0x00000000	Section 11.2.58
0x190	SDH_IN_DATA_MSB –Input Data MSB Register	0x00000000	Section 11.2.59
0x194	SDH_IN_KEY_LSB – Input Key LSB Register	0x00000000	Section 11.2.60
0x198	SDH_IN_KEY_MSB – Input Key MSB Register	0x00000000	Section 11.2.61
0x19C	SDH_OUT_DATA_LSB – Output Data LSB Register	0x00000000	Section 11.2.62
0x1A0	SDH_OUT_DATA_MSB – Output Data MSB Register	0x00000000	Section 11.2.63
0x1A4	SDH_SCRT_CONS_DATA – Secret Constant Table Data Port	0x00000000	Section 11.2.64

Table 11.1 – Overview of SD Host Registers

11.2 Register Details

11.2.1 SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register (address offset: 0x00)

This register sets a 32-bit block count to the argument of CMD23 while executing Auto CMD23. The available block count will be limited by BLK_CNT. In this case, the maximum value is 65535 blocks.

Bit	Name	Type	Default Value	Description
31:0	ARG_2	RW	32'h00000000	Auto CMD23 argument 2

Table 11.2 – SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register

Copyright © Bridgetek Pte Ltd

11.2.2 SDH_BLK_SIZE – Block Size Register (address offset: 0x04)

This register is used to configure the number of bytes in a data block.

Bit	Name	Type	Default Value	Description
15:12	Reserved	-	-	-
11:0	BLK_SIZE	RW	12'h000	Specifies the block size of data transfers for CMD17/18/24/25/53 and can be set with values ranging from 1 up to the maximum buffer size. It can be accessed only when no transaction is executed (i.e., after a transaction has stopped). 12'h0800: 2048 bytes … 12'h0002: 2 bytes 12'h0001: 1 byte 12'h0000: No transfer

Table 11.3 – SDH_BLK_SIZE – Block Size Register

11.2.3 SDH_BLK_COUNT – Block Count Register (address offset: 0x06)

The block count register is set when the BLK_CNT_EN bit is set to 1. This register is used only for the multi-block transfers. The host controller will decrease the counting number during the data transfer and stop counting when it counts down to zero. When a suspend command is completed, the remaining block counts can be determined by reading this register. Before issuing a resume command to start a re-transfer, the host driver should restore the block counts that are previously saved.

Bit	Name	Type	Default Value	Description
31:16	BLK_CNT	RW	16'h000	Block count of the current transfer. 16'hFFFF: 65535 blocks … 16'h0002: 2 blocks 16'h0001: 1 block **16'h0000: Stop counting**

Table 11.4 – SDH_BLK_COUNT – Block Count Register

11.2.4 SDH_ARG_1 – Argument 1 Register (address offset: 0x08)

This register is assigned to bits[39:8] of the command field.

Bit	Name	Type	Default Value	Description
31:0	ARG1	RW	32'h00000000	Command argument

Table 11.5 – SDH_ARG_1 – Argument 1 Register

11.2.5 SDH_TNSFER_MODE – Transfer Mode Register (address offset: 0x0C)

The host driver should set this register before issuing the data transfer command or the resume command. When in the SDIO transfer, the values of this register should be preserved after the suspend command and should be restored before the resume command.

Bit	Name	Type	Default Value	Description
15:6	Reserved	-	-	-
5	MULTI_BLK	RW	1'b0	1: Multiple blocks selection 0: Single block selection

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
4	TRAN_DIR_SEL	RW	1'b0	1: Read from the card to host 0: Write from the host to card
3:2	AUTO_CMD_EN	RW	2'h0	Auto CMD enable There are two methods to stop the read and write operations of multiple blocks: 01: Auto CMD12 Enable When this field is set to 01, the host controller will issue a CMD12 when the last block transfer is completed. 10: Auto CMD23 Enable When this bit field is set to 10, the host controller will issue a CMD23 before issuing a command specified in the Command Register. 11: Reserved 00: Auto Command Disabled
1	BLK_CNT_EN	RW	1'b0	Block count enable. This bit is only valid for a multi-block transfer. When set to 0, the BLK_CNT register will be disabled. The multi-block transfer will be an infinite transfer.
0	Reserved	RW	1'b0	Write 0 to this bit

Table 11.6 – SDH_TNSFER_MODE – Transfer Mode Register

11.2.6 SDH_CMD – Command Register (address offset: 0x0E)

The host driver should check the Command Inhibit (CMD) and Command Inhibit (DAT) bits in the present state register to determine whether the SD bus is free to transfer.

Bit	Name	Type	Default Value	Description		
31:30	Reserved	-	-	-		
29:24	CMD_IDX	RW	6'h00	Command Index. These bits should be assigned to bits [45:40] of the command field		
23:22	CMD_TYPE	RW	2'b00	11	Abort	CMD12/52 for writing I/O Abort in CCCR
				10	Resume	CMD52 for writing Function Select in CCCR
				01	Suspend	CMD52 for writing Bus Suspend in CCCR
				00	Normal	Other commands
21	DATA_PRE_SEL	RW	1'b0	Data Present Select. 1: Indicates that data is present and data transfer is enabled for the command. 0: Commands only using the CMD line, commands with no data transfer but set the busy signal on DAT[0], the resume command.		
20	CMD_IDX_CHK_EN	RW	1'b0	Command Index Check Enable 1: the host controller will check the index field response to determine if the values are CMD_IDX. If they are not the same, CMD_IDX_ERR will be triggered.		

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
19	CMD_CRC_CHK_EN	RW	1'b0	Command CRC Check Enable 1: the host controller will check the CRC field response to determine whether the CRC is correct. CMD_CRC_ERR will be triggered if an error is detected.
18	Reserved	-	-	-
17:16	RSP_TYPE_SEL	RW	2'h0	Response Type Select. 11: Response length 48 with busy check after response 10: Response length 48 01: Response length 136 00: No response

Table 11.7 – SDH_CMD – Command Register

11.2.7 SDH_RESPONSE – Response Register (address offset: 0x10-0x1C)

The host controller preserves the Auto CMD12 response in the upper word of the response register to prevent the value from being overwritten by responses to subsequent command.

Bit	Name	Type	Default Value	Description
127:0	RSP	RO	0	Command Response

Table 11.8 – SDH_RESPONSE – Response Register

11.2.8 SDH_BUF_DATA – Buffer Data Port Register (address offset: 0x20)

This register uses the 32-bit Data Port Register to access the internal buffer.

Bit	Name	Type	Default Value	Description
31:0	DATA_PORT	RW	32'h00000000	Buffer Data Port Register

Table 11.9 – 11.2.8 SDH_BUF_DATA – Buffer Data Port Register

11.2.9 SDH_PRESENT_STATE – Present State Register (address offset: 0x24)

The host driver can access the status from this read-only register. (Note: There is no default. The SD card you plug in will provide the values.)

Bit	Name	Type	Default Value	Description
31:25	Reserved	-	-	-
24	CMD_LIN_LV	RO	X	Command Line Signal Level
23:20	DATA_LIN_LV	RO	X	Data[3:0] Line Signal Level
19	WR_PROP_LV	RO	X	Write Protect Pin Level 1: Write enabled 0: Write protected
18	CD_PIN_LV	RO	X	Card Detect Pin Level 1: Card is detected 0: Card is not detected
17	SYS_CARD_STABLE	RO	X	Card State Stable 1: No card or card is inserted 0: Reset or de-bounce
16	SYS_CARD_INSERT	RO	X	Card Inserted 1: Card inserted 0: Reset, de-bouncing, or no card is detected
15:12	Reserved	-	-	-

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
11	BUF_RD_EN	ROC	X	Buffer Read Enable 1: Read Enable 0: Read Disable
10	BUF_WR_EN	ROC	X	Buffer Write Enable 1: Write Enable 0: Write Disable
9	RD_TRAN_ACT	ROC	X	Read Transfer Active 1: under the following conditions: (1) After the end bit of a read command (2) When CONT_REQ in the block gap control register is set to restart a transfer. 0: under the following conditions: (1) When all data blocks specified by the block length are transferred to the system. (2) When SP_BLK_GAP in the block gap control register is set to 1, and the host controller has transferred all the valid data blocks to the system. The TRAN_CMPLT interrupt is generated when this bit changes from 1 to 0.
8	WR_TRAN_ACT	ROC	X	Write Transfer Active 1: under the following conditions: (1) After the end bit of a write command (2) When CONT_REQ in the block gap control register is set to restart a transfer. 0: under the following conditions: (1) After getting the CRC status of the last data block specified by the transfer count. (2) After getting the CRC status of any block where data transmission is stopped by SP_BLK_GAP. A BLK_GAP_EVT interrupt will be generated when SP_BLK_GAP is set to 1 and this bit changes to 0. This bit is useful in the command with a busy data line.
7:3	Reserved	-	-	-
2	DATA_LIN_ACT	ROC	X	Data Line Active. In a read transfer, this status bit is used to check whether a read transfer is executing on the bus. Changing this bit from 1 to 0 will generate a BLK_GAP_EVT interrupt when SP_BLK_GAP is set to 1. 1: under the following conditions: (1) After the end bit of a read command (2) When CONT_REQ in the block gap control register is set to restart a transfer. 0: under the following conditions: (1) When the end bit of the last data block is sent from the SD bus to the host controller. (2) When SP_BLK_GAP is set to 1 and a read transfer is stopped at the block gap. In a write transfer, this status bit is used to check whether a write transfer is executing on the bus. Changing this bit from 1 to 0 will generate a TRAN_CMPLT interrupt in the normal interrupt status register. 1: under the following conditions:

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
				(1) After the end bit of a read command (2) When CONT_REQ in the block gap control register is set to restart a transfer. 0: under the following conditions: (1) When the card releases the busy signal of the last data block. (2) When SP_BLK_GAP is set to 1 and the card releases the write busy at the block gap. In the command with busy data line, this bit indicates whether a command with busy is executing on the bus. This bit will be set after the end bit of the command with busy and will be cleared when busy is de-asserted or busy is not detected after the end of a response.
1	CMD_INHIBIT_DAT	ROC	X	Command Inhibit (DAT) 1: Cannot issue new commands to use the data line 0: Issue new commands to use the data line
0	CMD_INHIBIT_CMD	ROC	X	Command Inhibit (CMD) 1: Cannot issue command 0: Issue command only with the command line

Table 11.10 – SDH_PRESENT_STATE – Present State Register

11.2.10 SDH_HST_CNTL_1 – Host Control 1 Register (address offset: 0x28)

Bit	Name	Type	Default Value	Description
7	CD_SEL	RW	1'b0	Card Detect Signal Selection 1: The test level for the card detection 0: The card detect pin is selected
6	CD_TEST_LV	RW	1'b0	Card Detect Test Level 1: Card is inserted 0: Card cannot be found
5	EXT_DATA_WIDTH	RW	1'b0	Extended Data Transfer Width 1: 8-bit bus width 0: Bus width is selected by the data transfer width
4:2	Reserved	-	-	-
1	DATA_WIDTH	RW	1'b0	Data Width 1: 4-bit mode 0: 1-bit mode
0	Reserved	-	-	-

Table 11.11 – SDH_HST_CNTL_1 – Host Control 1 Register

11.2.11 SDH_PWR_CNTL – Power Control Register (address offset: 0x29)

Bit	Name	Type	Default Value	Description
15:12	Reserved	-	-	-
11:9	SD_BUS_VOL	RW	3'h0	SD Bus Voltage Select 111: 3.3V (Typ) 110: 3.0V (Typ) 101: 1.8V (Typ) Others: Reserved

Copyright © Bridgetek Pte Ltd

8	Reserved	-	-	-

Table 11.12 – SDH_PWR_CNTL – Power Control Register

11.2.12 SDH_BLK_GAP_CNTL – Block Gap Control Register (address offset: 0x2A)

Bit	Name	Type	Default Value	Description
23:20	Reserved	-	-	-
19	INT_BLK_GAP	RW	1'b0	1: Checks the interrupt at block gap enabled
18	READ_WAIT	RW	1'b0	1: Enables Read Wait
17	CONT_REQ	RWAC	1'b0	1: Restarts a transaction (SP_BLK_GAP must also be 1; if 0, this will be aborted) It is cleared automatically by the host controller when: a. In a read transfer, DATA_LIN_ACT changes from 0 to 1 to start a read transfer b. In a write transfer, WR_TRAN_ACT changes from 0 to 1 to start a write transfer
16	SP_BLK_GAP	RW	1'b0	1: Stop at block gap request; the host controller will stop at the block gap by using READ_WAIT, or stop the SD clock output to the card in a read transaction 0: the host controller will not write data to DATA_PORT

Table 11.13 – SDH_BLK_GAP_CNTL – Block Gap Control Register

Copyright © Bridgetek Pte Ltd

11.2.13 SDH_CLK_CNTL – Clock Control Register (address offset: 0x2C)

Bit	Name	Type	Default Value	Description
15:8	LOW_BIT_SD_CLK_SEL	RW	8'h00	Bits 7:0 of the 10-bit SD clock divider. The divider is used to select the frequency of the SD clock output to the card. The base clock is ½ of the chip system clock (i.e., base clock = 50 MHz). N: chip system clock * (1 / 2N) … 0x002: base clock / 4 0x001: base clock / 2 0x000: base clock
7:6	UPPER_BIT_SD_CLK_SEL	RW	2'h0	SD clock frequency bits 9:8 for 10-bit divided clock mode.
5	CLK_GEN_SEL	ROC	1'b0	This bit is always set to zero 0: 111.20-bit divided clock mode
4:3	Reserved	-	-	-
2	SD_CLK_EN	RW	1'b0	1: IO_SD_CLK will be output
1	CLK_STABLE	ROC	1'b0	This bit is set to 1 when the internal clock is stable
0	INTER_CLK_EN	RW	1'b0	1: Internal clock will start oscillating

Table 11.14 – SDH_CLK_CNTL – Clock Control Register

11.2.14 SDH_TIMEOUT_CNTL – Timeout Control Register (address offset: 0x2E)

The value of DATA_TIMER indicates the timeout times of data line.

Bit	Name	Type	Default Value	Description
23:20	Reserved	-	-	-
19:16	DATA_TIMER	RW	4'hE	1111: Reserved 1110: System clock x 2^27 1101: System clock x 2^26 … 0000: System clock x 2^13

Table 11.15 – SDH_TIMEOUT_CNTL – Timeout Control Register

11.2.15 SDH_SW_RST – Software Reset Register (address offset: 0x2F)

A reset pulse will be generated when this bit is set to 1. This bit will be automatically cleared once the reset pulse is issued.

Bit	Name	Type	Default Value	Description
31:27	Reserved	-	-	-
26	SOFT_RST_DAT	RWAC	1'b0	1: Software reset for data line
25	SOFT_RST_CMD	RWAC	1'b0	1: Software reset for command line
24	SOFT_RST_ALL	RWAC	1'b0	1: Software reset for all

Table 11.16 – SDH_SW_RST – Software Reset Register

Copyright © Bridgetek Pte Ltd

11.2.16 SDH_NRML_INT_STATUS – Normal Interrupt Status Register (address offset: 0x30)

The interrupt status can be latched by setting the Normal Interrupt Status Enable Register corresponding bit to 1.

Bit	Name	Type	Default Value	Description
15	ERR_INT	ROC	1'b0	1: Error Interrupt
14:9	Reserved	-	-	-
8	CARD_INT	ROC	1'b0	1: Card Interrupt
7	CARD_REMOVE	RW1C	1'b0	1: Card Remove
6	CARD_INSERT	RW1C	1'b0	1: Card Inserted
5	BUF_RD_RDY	RW1C	1'b0	1: Buffer Read Ready
4	BUF_WR_RDY	RW1C	1'b0	1: Buffer Write Ready
3	Reserved	RW1C	1'b0	Write 0 to this bit
2	BLK_GAP_EVT	RW1C	1'b0	1: Block Gap Event
1	TRAN_CMPLT	RW1C	1'b0	1: Transfer Complete
0	CMD_CMPLT	RW1C	1'b0	1: Command Complete

Table 11.17 – SDH_NRML_INT_STATUS – Normal Interrupt Status Register

11.2.17 SDH_ERR_INT_STATUS – Error Interrupt Status Register (address offset: 0x32)

The interrupt status can be latched by setting the corresponding bit in the Error Interrupt Status Enable Register.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
31:27	Reserved	-	-	-
26	Reserved	RW1C	1'b0	Write 0 to this bit
25	Reserved	RW1C	1'b0	Write 0 to this bit
24	AUTO_CMD12_ERR	RW1C	1'b0	Auto CMD12 error
23	CUR_LIM_ERR	RW1C	1'b0	Current limit error
22	DATA_END_BIT_ERR	RW1C	1'b0	Data End Bit error
21	DATA_CRC_ERR	RW1C	1'b0	Data CRC error
20	DATA_TIMEOUT_ERR	RW1C	1'b0	Data Timeout error
19	CMD_IDX_ERR	RW1C	1'b0	Command Index error
18	CMD_ERR_BIT_ERR	RW1C	1'b0	Command End Bit error
17	CMD_CRC_ERR	RW1C	1'b0	Command CRC error
16	CMD_TIMEOUT_ERR	RW1C	1'b0	Command Timeout error

Table 11.18 – SDH_ERR_INT_STATUS – Error Interrupt Status Register

11.2.18 SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register (address offset: 0x34)

If the corresponding bit of the interrupt source in the Normal Interrupt Status Enable register is set to 1, the interrupt becomes active, which is latched and available for the host driver in the Normal Interrupt Status Register.

Bit	Name	Type	Default Value	Description
15	Reserved	-	-	-
14:9	Reserved	-	-	-
8	CARD_INT_ST_EN	RW	1'b0	Card Interrupt status enable
7	CARD_REMOVE_ST_EN	RW	1'b0	Card Remove status enable
6	CARD_INSERT_ST_EN	RW	1'b0	Card Insert status enable
5	BUF_RD_RDY_ST_EN	RW	1'b0	Buffer Read Ready status enable
4	BUF_WR_RDY_ST_EN	RW	1'b0	Buffer Write Ready status enable
3	Reserved	RW	1'b0	Write 0 to this bit
2	BLK_GAP_EVT_ST_EN	RW	1'b0	Block Gap Event status enable
1	TRAN_CMPLT_ST_EN	RW	1'b0	Transfer Complete status enable
0	CMD_CMPLT_ST_EN	RW	1'b0	Command Complete status enable

Table 11.19 – SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register

11.2.19 SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register (address offset: 0x36)

If the corresponding bit of the interrupt source in the Error Interrupt Status Enable Register is set to 1, and if the interrupt becomes active, the active state will be latched and will be available for the host driver in this register.

Bit	Name	Type	Default Value	Description
31:26	Reserved	-	-	-
25	Reserved	RW	1'b0	Write 0 to this bit
24	AUTO_CMD12_ERR_ST_EN	RW	1'b0	Auto CMD12 error status enable

Copyright © Bridgetek Pte Ltd

23	CUR_LIM_ERR_ST_EN	RW	1'b0	Current limit error status enable
22	DATA_END_BIT_ERR_ST_EN	RW	1'b0	Data End Bit error status enable
21	DATA_CRC_ERR_ST_EN	RW	1'b0	Data CRC error status enable
20	DATA_TIMEOUT_ERR_ST_EN	RW	1'b0	Data Timeout error status enable
19	CMD_IDX_ERR_ST_EN	RW	1'b0	Command Index error status enable
18	CMD_END_BIT_ERR_ST_EN	RW	1'b0	Command End Bit error status enable
17	CMD_CRC_ERR_ST_EN	RW	1'b0	Command CRC error status enable
16	CMD_TIMEOUT_ERR_ST_EN	RW	1'b0	Command Timeout error status enable

Table 11.20 – SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register

11.2.20 SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register (address offset: 0x38)

This register is used to select which interrupts are notified to the host system.

Bit	Name	Type	Default Value	Description
15	Reserved	-	-	-
14:9	Reserved	-	-	-
8	CARD_INT_SIG_EN	RW	1'b0	Card Interrupt signal enable
7	CARD_REMOVE_SIG_EN	RW	1'b0	Card Remove signal enable
6	CARD_INSERT_SIG_EN	RW	1'b0	Card Insert signal enable
5	BUF_RD_RDY_SIG_EN	RW	1'b0	Buffer Read Ready signal enable
4	BUF_WR_RDY_SIG_EN	RW	1'b0	Buffer Write Ready signal enable
3	Reserved	RW	1'b0	Write 0 to this bit
2	BLK_GAP_EVT_SIG_EN	RW	1'b0	Block Gap Event signal enable
1	TRAN_CMPLT_SIG_EN	RW	1'b0	Transfer Complete signal enable
0	CMD_CMPLT_SIG_EN	RW	1'b0	Command Complete signal enable

Table 11.21 – SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register

11.2.21 SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register (address offset: 0x3A)

This register is used to select which interrupts are notified to the host system.

Bit	Name	Type	Default Value	Description
31:26	Reserved	-	-	-
25	Reserved	RW	1'b0	Write 0 to this bit
24	AUTO_CMD12_ERR_SIG_EN	RW	1'b0	Auto CMD12 error signal enable
23	CUR_LIM_ERR_SIG_EN	RW	1'b0	Current limit error signal enable
22	DATA_END_BIT_ERR_SIG_EN	RW	1'b0	Data End Bit error signal enable
21	DATA_CRC_ERR_SIG_EN	RW	1'b0	Data CRC error signal enable

Copyright © Bridgetek Pte Ltd

20	DATA_TIMEOUT_ERR_SIG_EN	RW	1'b0	Data Timeout error signal enable
19	CMD_IDX_ERR_SIG_EN	RW	1'b0	Command Index error signal enable
18	CMD_END_BIT_ERR_SIG_EN	RW	1'b0	Command End Bit error signal enable
17	CMD_CRC _ERR_SIG_EN	RW	1'b0	Command CRC error signal enable
16	CMD_TIMEOUT_ERR_SIG_EN	RW	1'b0	Command Timeout error signal enable

Table 11.22 – SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register

11.2.22 SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register (address offset: 0x3C)

When the auto_cmd12_en register is set to 1 and the auto cmd12 error status register is set, the host driver will check this register to identify what kind of error happens during executing AUTO CMD12. This register is valid only when the auto_cmd12_err is set to 1.

Bit	Name	Type	Default Value	Description
15:8	Reserved	-	-	-
7	CMD_NO_EX_BY_CMD12	ROC	1'b0	Command not executed by Auto CMD12 error
6:5	Reserved	-	-	-
4	AUTO_CMD_IDX_ERR	ROC	1'b0	Auto CMD index error
3	AUTO_CMD_END_BIT_ERR	ROC	1'b0	Auto CMD end bit error
2	AUTO_CMD_CRC_ERR	ROC	1'b0	Auto CMD CRC error
1	AUTO_CMD_TIMEOUT_ERR	ROC	1'b0	Auto CMD timeout error
0	AUTO_CMD12_NO_EX	ROC	1'b0	Auto CMD12 not executed

Table 11.23 – SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register

11.2.23 SDH_HOST_CNTL_2 – Host Control 2 Register (address offset: 0x3E)

Bit	Name	Type	Default Value	Description
31	PRESET_VAL_EN	RW	1'b0	0: SDCLK and driver strength are controlled by the host driver 1: Automatic selection by the pre-set value
30	ASYN_INT_EN	R/W	1'b0	Asynchronous Interrupt Enable
29:24	Reserved	-	-	-
23	SAMPLE_CLK_SEL	RW	1'b0	Sampling clock select 0: Fixed clock is used to sample data
22:16	Reserved	-	-	-

Table 11.24 – SDH_HOST_CNTL_2 – Host Control 2 Register

11.2.24 SDH_CAP_1 – Capabilities Register 1 (address offset: 0x40)

The host controller may implement these values during initialization.

Bit	Name	Type	Default Value	Description
31:30	SLOT_TYPE	RO	2'b00	2'b00: Removable Card Slot

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
29	ASYNC_INT_SUPPORT	RO	1'b0	0: Asynchronous interrupt (for SDIO) is not supported
28	BUS_64_SUPPORT	RO	1'b0	0: 64-bit system address is not supported
27	Reserved	-	-	-
26	VOLTAGE_1V8_SUPPORT	RO	1'b0	0: 1.8V signal is not supported
25	VOLTAGE_3V0_SUPPORT	RO	1'b0	0: 3.0V signal is not supported
24	VOLTAGE_3V3_SUPPORT	RO	1'b1	1: 3.3V signal is supported
23	SUSPEND_RESUME_SUPPORT	RO	1'b0	0: Suspend/Resume (for SDIO) is not supported
22	SDMA_SUPPORT	RO	1'b0	0: SDMA is not supported
21	HI_SPEED_SUPPORT	RO	1'b1	1: High speed (SD clock from 25 MHz to 50 MHz) is supported
20	ADMA1_SUPPORT	RO	1'b0	0: ADMA1 is not supported
19	ADMA2_SUPPORT	RO	1'b0	0: ADMA2 is not supported
18	8BIT_SUPPORT	RO	1'b0	0: 8-bit data bus is not supported
17:16	MAX_BLK_LEN	R0	2'h2	This value indicates the maximum block size that can be read and written by the host driver to the buffer in the host controller. 2'h2: 2048 bytes
15:8	BASE_CLK_FOR_SD_CLK	RO	8'h00	Base Clock Frequency For SD Clock 8'h00: The information is not available from these bits. For information, the base clock frequency is fixed at 50 MHz in FT90x series.
7	TIMEOUT_CLK_UNIT	RO	1'b1	Shows the unit of the base clock frequency used to detect data timeout error. 1: MHz
6	Reserved	-	-	-
5:0	TIMEOUT_CLK_FREQ	RO	6'h00	The base clock frequency used to detect data timeout error. 6'h00: The information is not available from these bits. For information, the base clock is the same as the system clock, i.e., 100 MHz.

Table 11.25 – SDH_CAP_1 – Capabilities Register 1

Copyright © Bridgetek Pte Ltd

11.2.25 SDH_CAP_2 – Capabilities Register 2 (address offset: 0x44)

Bit	Name	Type	Default Value	Description
31:24	Reserved	-	-	-
23:15	CLK_MULTIPLIER	RO	8'h00	Clock Multiplier is not supported.
15:14	RE_TUNING_MODE	RO	2'h0	Re-Tuning Mode 1. The host controller does not have any internal logic to detect when the re-tuning needs to be performed.
13	TUNING_SDR50	RO	1'b0	SDR50 does not require tuning.
12	Reserved	-	-	-
11:8	TIMER_CNT_FOR_RE_TUNING	RO	4'hF	Re-Tuning Timer is not supported.
7	Reserved	-	-	-
6	DRIVER_D_SUPPORT	RO	1'b0	Driver Type D is not supported for the 1.8-V signalling.
5	DRIVER_C_SUPPORT	RO	1'b0	Driver Type C is not supported for the 1.8-V signalling.
4	DRIVER_A_SUPPORT	RO	1'b0	Driver Type A is not supported for the 1.8-V signalling.
3	Reserved	-	-	-
2	DDR50_SUPPORT	RO	1'b0	DDR50 is not supported.
1	SDR104_SUPPORT	RO	1'b0	SDR104 is not supported.
0	SDR50_SUPPORT	RO	1'b0	SDR50 is supported. **Note:** SDR50 is supported by the FT90x series even though this bit is '0'.

Table 11.26 – SDH_CAP_2 – Capabilities Register 2

11.2.26 SDH_RSRV_1 – Reserved 1 Register (address offset: 0x48)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.27 – SDH_RSRV_1 – Reserved 1 Register

11.2.27 SDH_RSRV_2 – Reserved 2 Register (address offset: 0x4C)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.28 – SDH_RSRV_2 – Reserved 2 Register

11.2.28 SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status (address offset: 0x50)

The Force Event register is not a physical register. It is an address to which the Auto CMD error status register can be written. The force event register is only for debugging.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
15:8	Reserved	-	-	-
7	R_CMD_NO_EX_BY_CMD12	WO	1'b0	Force event for the Command Not Executed by Auto CMD12 Error
6:5	Reserved	-	-	-
4	R_CMD_IDX_ERR	WO	1'b0	Force event for the Auto CMD Index Error
3	R_CMD_END_BIT_ERR	WO	1'b0	Force event for the Auto CMD End Bit Error
2	R_CMD_CRC_ERR	WO	1'b0	Force event for the Auto CMD CRC Error
1	R_CMD_TIMEOUT_ERR	WO	1'b0	Force event for the Auto CMD Timeout Error
0	R_CMD12_NO_EX	WO	1'b0	Force event for the Auto CMD12 Not Executed

Table 11.29 – SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status

11.2.29 SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register (address offset: 0x52

The Force Event register is not a physical register. It is an address to which the Error Interrupt Status Register can be written. This Force Event register is for debugging only. The effect of writing to this address will be reflected in the Error Interrupt Status Register if the corresponding bit of the Error Interrupt Status Enable register is set.

Bit	Name	Type	Default Value	Description
31:25	Reserved	-	-	Write 0 to these bits
24	R_AUTP_CMD_ERR	WO	1'b0	Force Event for the Auto CMD Error
23	R_CUR_LIMIT_ERR	WO	1'b0	Force Event for the Current Limit Error
22	R_DATA_END_BIT_ERR	WO	1'b0	Force Event for the Data End Bit Error
21	R_DATA_CRC_ERR	WO	1'b0	Force Event for the Data CRC Error
20	R_DATA_TIMEOUT_ERR	WO	1'b0	Force Event for the Data Timeout Error
19	R_CMD_IDX_ERR	WO	1'b0	Force Event for the Command Index Error
18	R_CMD_END_BIT_ERR	WO	1'b0	Force Event for the Command End Bit Error
17	R_CMD_CRC_ERR	WO	1'b0	Force Event for the Command CRC Error
16	R_CMD_TIMEOUT_ERR	WO	1'b0	Force Event for the Command Timeout Error

Table 11.30 – SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register

11.2.30 SDH_RSRV_3 – Reserved 3 Register (address offset: 0x54)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.31- SDH_RSRV_3 – Reserved 3 Register

11.2.31 SDH_RSRV_4 – Reserved 4 Register (address offset: 0x58)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Copyright © Bridgetek Pte Ltd

Table 11.32 – SDH_RSRV_4 – Reserved 4 Register

11.2.32 SDH_PRST_INIT – Preset value for initialization (address offset: 0x60)

Bit	Name	Type	Default Value	Description
15:14	DRIVER_STR_SEL	RO		Driver Strength Select Value Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling. 11: Driver type D is selected 10: Driver type C is selected 01: Driver type A is selected 00: Driver type B is selected
13:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value The version does not support the programmable clock generator and is fixed to 0.
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.33 – SDH_PRST_INIT – Preset value for initialization

11.2.33 SDH_PRST_DFLT_SPD – Preset value for default speed (address offset: 0x62)

Bit	Name	Type	Default Value	Description
15:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value. The version does not support the programmable clock generator and is fixed to 0.
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value. The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.34 – SDH_PRST_DFLT_SPD – Preset value for default speed

11.2.34 SDH_PRST_HIGH_SPD – Preset value for the high speed (address offset: 0x64)

Bit	Name	Type	Default Value	Description
15:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO		Clock Generator Select Value. The version does not support the programmable clock generator and is fixed to 0.
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value. The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.35 – SDH_PRST_HIGH_SPD – Preset value for the high speed

Copyright © Bridgetek Pte Ltd

11.2.35 SDH_PRST_SDR12 – Preset value for SDR12 (address offset: 0x66)

Bit	Name	Type	Default Value	Description
15:14	DRIVER_STR_SEL	RO		Driver Strength Select Value Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling. 11: Driver type D is selected 10: Driver type C is selected 01: Driver type A is selected 00: Driver type B is selected
13:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value The version does not support the programmable clock generator and is fixed to 0.
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.36 – SDH_PRST_SDR12 – Preset value for SDR12

11.2.36 SDH_PRST_SDR25 – Preset value for SDR25 (address offset: 0x68)

Bit	Name	Type	Default Value	Description
15:14	DRIVER_STR_SEL	RO		Driver Strength Select Value. Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling. 11: Driver type D is selected 10: Driver type C is selected 01: Driver type A is selected 00: Driver type B is selected
13:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value The version does not support the programmable clock generator and is fixed to 0.

Copyright © Bridgetek Pte Ltd

9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.37 – SDH_PRST_SDR25 – Preset value for SDR25

11.2.37 SDH_PRST_SDR50 – Preset value for SDR50 (address offset: 0x6A)

Bit	Name	Type	Default Value	Description
15:14	DRIVER_STR_SEL	RO		Driver Strength Select Value Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling. 11: Driver type D is selected 10: Driver type C is selected 01: Driver type A is selected 00: Driver type B is selected
13:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value. The version does not support the programmable clock generator and is fixed to 0.
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value. The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.38 – SDH_PRST_SDR50 – Preset value for SDR50

11.2.38 SDH_PRST_SDR104 – Preset value for SDR104 (address offset: 0x6C)

Bit	Name	Type	Default Value	Description
15:14	DRIVER_STR_SEL	RO		Driver Strength Select Value. Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling. 11: Driver type D is selected 10: Driver type C is selected 01: Driver type A is selected 00: Driver type B is selected
13:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value The version does not support the programmable clock generator and is fixed to 0
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.39 – SDH_PRST_SDR104 – Preset value for SDR104

11.2.39 SDH_PRST_DDR50 – Preset value for DDR50 (address offset: 0x6E)

Bit	Name	Type	Default	Description

Copyright © Bridgetek Pte Ltd

			Value	
15:14	DRIVER_STR_SEL	RO		Driver Strength Select Value. Driver Strength is supported by the 1.8-V signalling bus speed modes. This field is meaningless for the 3.3-V signalling. 11: Driver type D is selected 10: Driver type C is selected 01: Driver type A is selected 00: Driver type B is selected
13:11	Reserved	-	-	-
10	CLK_GEN_SEL	RO	1'b0	Clock Generator Select Value. The version does not support the programmable clock generator and is fixed to 0.
9:0	SDCLK_FREQ_SEL	RO		SDCLK Frequency Select Value The 10-bit pre-set value for setting the SDCLK Frequency Select in the Clock Control. The register is described by a host system.

Table 11.40 – SDH_PRST_DDR50 – Preset value for DDR50

11.2.40 SDH_RSRV_5 – Reserved 5 Register (address offset: 0xFC)

Bit	Name	Type	Default Value	Description
15:0	Reserved	-	-	-

Table 11.41 – SDH_RSRV_5 – Reserved 5 Register

11.2.41 SDH_HC_VER – Host Controller Version Register (address offset: 0xFE)

Bit	Name	Type	Default Value	Description
31:24	VNDR_VER_NUM	RO	8'h00	Vendor Version Number
23:16	SPEC_VER_NUM	RO	8'h02	Specification Version Number

Table 11.42 – SDH_HC_VER – Host Controller Version Register

11.2.42 SDH_VNDR_0 – Vendor-defined 0 Register (address offset: 0x100)

Bit	Name	Type	Default Value	Description
31:28	Reserved	-	-	-
27:24	N_CRC	RW	4'h0	Write CRC Status Wait Cycle. The host controller is used to set 5 SCLK clock cycles for the specifications and round-chip effect. Users can add the wait cycle for other factors.
23:17	Reserved	-	-	-
16	INT_EDGE_SEL	RW	1'b0	1: The CMD and DAT line output at the rising edge of SCLK 0: The CMD and DAT line output at the falling edge of SCLK
15:14	Reserved	-	-	-
13:8	P_LAT_OFF	RW	6'h00	Pulse latch offset. When the host controller uses the pulse latch to sample the read data and response, users need to set the latch offset to correctly sample the value. The values set should be smaller than

Copyright © Bridgetek Pte Ltd

				the SDCLK Frequency Select. 0x3F: Latch value at the 63rd ½ chip frequency clock rising edge after the SCLK edge …. 0x01: Latch value at the 1st 0x00: Latch value at SCLK edge
7:1	Reserved	-	-	-
0	P_LAT_EN	RW	1'b1	1: Use the pulse latching function for the read data and response. Should always be set to 1.

Table 11.43 – SDH_VNDR_0 – Vendor-defined 0 Register

Copyright © Bridgetek Pte Ltd

11.2.43 SDH_VNDR_1 – Vendor-defined 1 Register (address offset: 0x104)

Bit	Name	Type	Default Value	Description
31:25	Reserved	-	-	-
24	CMD_CONFLICT_EN	RW	1'b0	1: Enable host controller to check the CMD line conflict error
23:19	Reserved	-	-	-
18:16	N_SB	RW	3'h0	N_SB Timing: Users can add the busy wait cycle for other factors. The host controller is set to 5 SCLK clock cycles.
15:12	Reserved	-	-	-
11:8	N_CR	RW	3'h0	N_CR Timing: Users can add the response wait cycle for other factors. The host controller is set to 64 SCLK clock cycles.
7:3	Reserv	-	-	-

Copyright © Bridgetek Pte Ltd

	ed			
2	MMC_BOOT_ACK_EN	RW	1'b0	MMC Booting Mode Acknowledge Enable
1:0	MMC_BOOT	RW	2'h0	MMC Booting Mode Selection 11: MMC Bus Test mode 10: MMC Alternative Boot Mode 01: MMC Boot mode 00: Normal mode

Table 11.44 – SDH_VNDR_1 – Vendor-defined 1 Register

11.2.44 SDH_VNDR_2 – Vendor-defined 2 Register (address offset: 0x108)

Bit	Name	Type	Default Value	Description
31:1	Reserved	-	-	-
0	CLK_CTRL_SW_RST	RWAC	1'b0	1: To reset the clock control of the host controller.

Table 11.45 – SDH_VNDR_2 – Vendor-defined 2 Register

Copyright © Bridgetek Pte Ltd

11.2.45 SDH_VNDR_3 – Vendor-defined 3 Register (address offset: 0x10C)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.46 – SDH_VNDR_3 – Vendor-defined 3 Register

11.2.46 SDH_VNDR_4 – Vendor-defined 4 Register (address offset: 0x110)

Bit	Name	Type	Default Value	Description
31:0	Reserved	RO	-	-

Table 11.47 – SDH_VNDR_4 – Vendor-defined 4 Register

Copyright © Bridgetek Pte Ltd

11.2.47 SDH_VNDR_5 – Vendor-defined 5 Register (address offset: 0x114)

Bit	Name	Type	Default Value	Description
31:4	Reserved	-	-	-
3:0	DB_TIMEOUT	RW	4'h0	Card Insertion De-bounce Cycle 0: 2^9 chip system clock cycles 1: 2^{10} chip system clock cycles …. 15: 2^{24} chip system clock cycles

Table 11.48 – SDH_VNDR_5 – Vendor-defined 5 Register

11.2.48 SDH_VNDR_6 – Vendor-defined 6 Register (address offset: 0x118)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.49 – SDH_VNDR_6 – Vendor-defined 6 Register

11.2.49 SDH_VNDR_7 – Vendor-defined 7 Register (address offset: 0x11C)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.50 – SDH_VNDR_7 – Vendor-defined 7 Register

11.2.50 SDH_VNDR_8 – Vendor-defined 8 Register (address offset: 0x120)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.51 – SDH_VNDR_8 – Vendor-defined 8 Register

11.2.51 SDH_VNDR_9 – Vendor-defined 9 Register (address offset: 0x124)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.52 – SDH_VNDR_9 – Vendor-defined 9 Register

11.2.52 SDH_RSRV_6 – Reserved 6 Register (address offset: 0x128)

Bit	Name	Type	Default Value	Description
31:0	Reserved	-	-	-

Table 11.53 – SDH_RSRV_6 – Reserved 6 Register

11.2.53 SDH_HW_ATTR – Hardware Attributes Register (address offset: 0x178)

Bit	Name	Type	Default Value	Description
31:9	Reserved	-	-	-
8:0	HW_CONFIG	RO	9'h150	8: Async 7: 4-bit SD data bus 6: CPRM present 5: DLL absent 4:0 DATA FIFO is 4-kB SRAM

Table 11.54 – SDH_HW_ATTR – Hardware Attributes Register

Copyright © Bridgetek Pte Ltd

11.2.54 SDH_CPR_MOD_CNTL – Cipher Mode Control Register (address offset: 0x180)

This register is the configurable register for the CPRM function. When the CPRM function is used, this register will be used to select the mode of the cipher function to encrypt or decrypt.

Bit	Name	Type	Default Value	Description
31:11	Reserved	-	-	-
10	SWAP_HL	RW	1'b0	1: Swap the high/low word of the encrypted data to TX FIFO The high-word and low-word of the encrypted data will be swapped before being written to TX FIFO. The high-word and low-word of the encrypted data will be swapped before decryption.
9	CH_ENDIAN	RW	1'b0	Change Endianness In this mode, the endianness of the encrypted data will be changed before being written to the TX FIFO. In this mode, the endianness of data from the RX FIFO will be changed before decryption.
8	SEC_ACCESS_EN	RW	1'b0	Secret Constant Table Access Enable This bit must be enabled before writing or reading the secret constant table. Once this bit is enabled, the firmware will always access from the very beginning of the secret constant table.
7	AUTO_C2_DCBC_EN	RW	1'b0	Auto C2 Decryption with C-CBC Mode Enable In this mode, data will be automatically decrypted and sent to the buffer. The data lengths should be multiples of 8 bytes.
6	AUTO_C2_ECBC_EN	RW	1'b0	Auto C2 Encryption with C-CBC Mode Enable In this mode, data written to the buffer will be automatically encrypted and sent to the TX FIFO. The data lengths should be multiples of 8 bytes.
5	RNGC2_G_EN	RW	1'b0	C2 Random Number Generator Enable
4	C2_DCBC_EN	RW	1'b0	C2 Decryption with C-CBC Mode Enable
3	C2_D_EN	RW	1'b0	C2 Decryption with EBC Mode Enable
2	C2_ECBC_EN	RW	1'b0	C2 Encryption with C-CBC Mode Enable
1	C2_E_EN	RW	1'b0	C2 Encryption with EBC Mode Enable
0	C2_G_EN	RW	1'b0	C2 One Way Function Enable

Table 11.55 – SDH_CPR_MOD_CNTL – Cipher Mode Control Register

11.2.55 SDH_CPR_MOD_STATUS – Cipher Mode Status Register (address offset: 0x184)

Bit	Name	Type	Default Value	Description
31:1	Reserved	-	-	-
0	CP_RDY	RW1C	1'b0	Cipher is ready. When this bit is set to 1, reading 0x19C and 0x1A0 will retrieve cipher or plain text.

Table 11.56 – SDH_CPR_MOD_STATUS – Cipher Mode Status Register

Copyright © Bridgetek Pte Ltd

11.2.56 SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register (address offset: 0x188)

Bit	Name	Type	Default Value	Description
31:1	Reserved	-	-	-
0	RDY_SIG_EN	RW	1'b0	Cipher Ready Signal Enable

Table 11.57 – SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register

11.2.57 SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register (address offset: 0x18A)

Bit	Name	Type	Default Value	Description
31:1	Reserved	-	-	-
0	RDY_SIG_EN	RW	1'b0	Cipher Ready Signal Enable

Table 11.58 – SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register

11.2.58 SDH_IN_DATA_LSB – Input Data LSB Register (address offset: 0x18C)

Bit	Name	Type	Default Value	Description
31:0	DATA	RW	32'h00000000	Input port for the input data bits 31:0

Table 11.59 – SDH_IN_DATA_LSB –Input Data LSB Register

11.2.59 SDH_IN_DATA_MSB – Input Data MSB Register (address offset: 0x190)

Bit	Name	Type	Default Value	Description
31:0	DATA	RW	32'h00000000	Input port for the input data bits 63:32

Table 11.60 – SDH_IN_DATA_MSB –Input Data MSB Register

11.2.60 SDH_IN_KEY_LSB – Input Key LSB Register (address offset: 0x194)

Bit	Name	Type	Default Value	Description
31:0	KEY	RW	32'h0000_0000	Input port for the input key bits 31:0

Table 11.61 – SDH_IN_KEY_LSB – Input Key LSB Register

11.2.61 SDH_IN_KEY_MSB – Input Key MSB Register (address offset: 0x198)

Bit	Name	Type	Default Value	Description
31:0	KEY	RW	32'h00000000	Input port for the input key bits 63:32

Table 11.62 – SDH_IN_KEY_MSB – Input Key MSB Register

11.2.62 SDH_OUT_DATA_LSB – Output Data LSB Register (address offset: 0x19C)

Bit	Name	Type	Default Value	Description
31:0	DATA	RW	32'h00000000	Output port for the output data bits 31:0

Table 11.63 – SDH_OUT_DATA_LSB – Output Data LSB Register

Copyright © Bridgetek Pte Ltd

11.2.63 SDH_OUT_DATA_MSB – Output Data MSB Register (address offset: 0x1A0)

Bit	Name	Type	Default Value	Description
31:0	DATA	RW	32'h00000000	Output port for the output data bits 63:32

Table 11.64 – SDH_OUT_DATA_MSB – Output Data MSB Register

11.2.64 SDH_SCRT_CONS_DATA – Secret Constant Table Data Port (address offset: 0x1A4)

Bit	Name	Type	Default Value	Description
31:8	Reserved	-	-	-
7:0	DATA_PORT	RW	8'h00	Secret constant table data port. 256 bytes are needed; this port should be written 256 time

Table 11.65 – SDH_SCRT_CONS_DATA – Secret Constant Table Data Port

Copyright © Bridgetek Pte Ltd

12 UART

The device supports 2 UARTs with the following features:

- Software compatible with 16450, 16550, 16750 and 16950 UARTs
- Separate configurable BAUD clock line
- Configuration capability
- Two modes of operation: UART mode and FIFO mode
- Majority voting logic
- 16 / 128 bytes FIFO for TX and RX in FIFO mode to reduce the interrupt frequency
- Adds or deletes standard asynchronous communication bits (start, stop and parity) to or from the serial data
- Double buffering for both TX and RX in UART mode
- Independently controlled transmit, receive, line status and data set interrupts
- Programmable baud generator
- MODEM control functions (CTS, RTS, DSR, DTR, RI and DCD)
- Programmable automatic out-of-band Flow Control logic through Auto-RTS and Auto-CTS
- Programmable automatic Flow Control logic using DTR and DSR
- Programmable automatic in-band Flow Control logic using XON / XOFF characters
- Programmable special characters detection
- Trigger levels for TX and RX FIFO interrupts, automatic in-band and out-of-band flow control
- RS-485 buffer enable signals
- TX and RX disable capability
- Fully programmable serial interface characteristics:
    - 5-, 6-, 7-, 8- or 9-bit characters
    - Even, odd, or no-parity bit generation and detection
    - 1-, 1.5, or 2-stop bit generation
    - Baud generation
    - Detection of bad data in receive FIFO
- Clock pre-scaler from 1 to 31875
- Enhanced isochronous clock option
- Complete status reporting capabilities
- False start bit detection
- Line-break generation and detection. Internal diagnostic capabilities:
    - Loop-back controls for communications link fault isolation
    - Break, parity, overrun, framing error simulation
- Full prioritised interrupt system controls
- Software reset

Copyright © Bridgetek Pte Ltd

12.1 Register Summary

The registers offset from the base addresses are 0x10320 for UART0 and 0x10330 for UART1. All registers can only be accessed via Byte (8-bit) mode.

Offset	Register	Default Value	References
	STANDARD 16550 COMPATIBLE REGISTERS		
0x00	UART_RBR – Receiver Buffer Register	0x00	Section 12.5.1
0x00	UART_THR – Transmitter Holding Register	0x00	Section 12.5.2
0x00	UART_DLL – Divisor Latch LSB Register	0x00	Section 12.5.3
0x01	UART_DLM – Divisor Latch MSB Register	0x00	Section 12.5.4
0x01	UART_IER – Interrupt Enable Register	0x00	Section 12.5.5
0x02	UART_ISR – Interrupt Status Register	0x00	Section 12.5.6
0x02	UART_FCR – FIFO Control Register	0x00	Section 12.5.7
0x03	UART_LCR – Line Control Register	0x00	Section 12.5.8
0x04	UART_MCR – Modem Control Register	0x00	Section 12.5.9
0x05	UART_LSR – Line Status Register	0x00	Section 12.5.10
0x06	UART_MSR – Modem Status Register	0x00	Section 12.5.11
0x07	UART_SPR – SPR Register	0x00	Section 12.5.12
	16650 COMPATIBLE REGISTERS		
0x02	UART_EFR – Enhanced Feature Register	0x00	Section 12.6.1
0x04	UART_XON1 – XON1 Register	0x00	Section 12.6.2
0x05	UART_XON2 – XON2 Register	0x00	Section 12.6.3
0x06	UART_XOFF1 – XOFF1 Register	0x00	Section 12.6.4
0x07	UART_XOFF2 – XOFF2 Register	0x00	Section 12.6.5
	16950 COMPATIBLE REGISTERS		
0x01	UART_ASR – Additional Status Register	0x00	Section 12.7.1
0x03	UART_RFL – Receiver FIFO Level Register	0x00	Section 12.7.2
0x04	UART_TFL – Transmitter FIFO Level Register	0x00	Section 12.7.3
0x05	UART_ICR – ICR Register	0x00	Section 12.7.4
	INDEXED CONTROL REGISTERS		

Copyright © Bridgetek Pte Ltd

0x00	UART_ACR – Additional Control Register	0x00	Section 12.8.1
0x01	UART_CPR – Clock Pre-scaler Register	0x00	Section 12.8.2
0x02	UART_TCR – Times Clock Register	0x00	Section 12.8.3
0x03	UART_CKS – Clock Select Register	0x00	Section 12.8.4
0x04	UART_TTL – Transmitter Trigger Level Register	0x00	Section 12.8.5
0x05	UART_RTL – Receiver Trigger Level Register	0x00	Section 12.8.6
0x06	UART_FCL – Flow Control Level LSB Register	0x00	Section 12.8.7
0x07	UART_FCH – Flow Control Level Register MSB	0x00	Section 12.8.8
0x08	UART_ID1 – Identification 1 Register	0x00	Section 12.8.9
0x09	UART_ID2 – Identification 2 Register	0x00	Section 12.8.10
0x0A	UART_ID3 – Identification 3 Register	0x00	Section 12.8.11
0x0B	UART_REV – Revision Register	0x00	Section 12.8.12
0x0C	UART_CSR – Channel Software Reset Register	0x00	Section 12.8.13
0x0D	UART_NMR – Nine Bit Mode Register	0x00	Section 12.8.14
0x0E	UART_MDM – Modem Disable Mask Register	0x00	Section 12.8.15
0x0F	UART_RFC – Readable FCR Register	0x00	Section 12.8.16
0x10	UART_GDS – Good Data Status Register	0x00	Section 12.8.17
0x11	UART_RSRV_1 – Reserved Register 1	0x00	Section 12.8.18
0x12	UART_PIDX – Port Index Register	0x00	Section 12.8.19
0x13	UART_RSRV_2 – Reserved Register 2	0x00	Section 12.8.20

Table 12.1 – Overview of UART Registers

Copyright © Bridgetek Pte Ltd

12.2 UART Mode Selection

The operation of the UART depends on a number of standard mode settings. These modes are referred to throughout this section. The modes, and how to enable them, are listed below.

UART Mode	FIFO Size in Bytes	UART_FCR[0]	Enhanced Mode UART_EFR[4] = 1	UART_FCR[5] (guarded with UART_LCR[7] = 1)	Bit UARTx_FIFOSEL in MSC0CFG
16450	1	0	X	X	X
16550	16	1	0	0	0
Extended 16550	128	1	0	X	1
16650	128	1	1	X	X
16750	128	1	0	1	0
16950	128	1	1	X	X

Table 12.2 – UART Mode Selection

16450 Mode

The 16450 mode is the default mode set after a hardware reset. In the 16450 Mode, the FIFO is disabled, and the UART operates in BYTE mode. With UART_FCR[0] cleared, (FIFO disabled) all other mode setting are ignored.

16550 Mode

In the 16550 mode, the FIFO's are enabled, and can accept up to 16 bytes of data (reception and transmission directions). To put the UART into 16550 mode, the UART_FCR[0] should be set high. In this mode, the FIFOSEL pin should be tied low.

Extended 16550 Mode

The extended 16550 mode is enabled by connection of the FIFOSEL to a HIGH state. In this mode, the FIFO size is increased to 128 bytes.

16650 Mode

The 16650 mode is active when UART_EFR[4] is set (enhanced mode is enabled). As 16650 software drivers usually put the device into enhanced mode, running 16650 drivers on the UART device will result in 16650 compatibilities with 128 deep FIFO's, as long as UART_FCR[0] is set.

The value of UARTx_FIFOSEL is ignored in the 16650 mode.

The 16650 mode enhancements over 16550 mode are:

- Deeper FIFO size
- Automatic RTS/CTS out-of-band flow control
- Sleep mode
- Automatic in-bank flow control
- Special character detection
- IRDA-format transmit and receive mode
- Transmit trigger levels
- Optional clock prescaler

Copyright © Bridgetek Pte Ltd

16750 Mode

The 16750 mode is enabled by writing the UART_FCR[0] with 1 and UART_FCR[5] with 1. In the 16750 mode, the FIFO size is set to 128 bytes. Please note, that writes to UART_FCR[5] are protected by UART_FCR[7]. To write UART_FCR[5], first set the UART_FCR[7] high, then write UART_FCR[5], and clear UART_FCR[7] back to activate protection. In the 16750 mode, the FIFOSEL pin should be tied low (0).

16750 mode enhancements over 16550 mode are:

- Deeper FIFO size
- Automatic RTS/CTS out-of-band flow control
- Sleep mode

16950 Mode

The additional features of 16950 mode apply only when UART is in Enhanced mode (UART_EFR[4] = 1). UART_FCR[0] set in Enhanced mode enables the 128 Bytes FIFO mode.

Configuration of the UART in 16950 Mode is identical with the 16650 Mode. Additional specific features of 16950 Mode's, are enabled using the Additional Control Register ACR. In addition to larger FIFO's, higher baud rates, the enhancements of 16950 over 16650 mode are:

- Selectable arbitrary trigger levels for the receiver and transmitter FIFO interrupts
- Improved automatic flow control using selectable arbitrary thresholds
- DSR/DTR automatic flow control
- Transmitter and receiver can be optionally disabled
- Software reset of device
- Readable FIFO fill levels
- Optional generation of an RS-485 buffer enable signal
- Four-byte device identification
- Readable status for automatic in-band and out-of-band flow control
- External 1x clock modes
- Flexible M N/8 clock pre-scaler
- 9-bit data mode

The 16950 trigger levels are enabled when UART_ACR[5] is set (UART_FCR[7:4] are ignored). The arbitrary trigger levels can be defined in UART_RTL, UART_TTL, UART_FCL and UART_FCH registers. The Additional Status Register (UART_ASR) offers flow control status for the local and remote transmitters. FIFO levels are readable using UART_RFL and UART_TFL registers.

The user may apply an external 1x (or Nx) clock for the transmitter and receiver to the RI and DSR pin respectively. The transmitter clock may be asserted on the DTR pin. The external clock options are selected through the UART_CKS register.

It is also possible to define the over-sampling rate used by the transmitter and receiver clocks. The 16450/16550/16750 and compatible devices employ 16 times over-sampling. There are 16 clock cycles per bit. The UART can employ over-sampling rate from 4 to 16 by programming the UART_TCR register. This allows the data rates to be increased. Default value after reset for this register is 0x00, which corresponds to a 16-cycle sampling clock. Writing 0x01, 0x02 or 0x03 will also result in 16 cycle sampling clock. To program the value to any value from 4 to 15, it is necessary to write this value into UART_TCR. To set the device to a 13-cycle sampling clock, it would be necessary to write 0x0D to UART_TCR.

The UART also offers 9-bit data frames for multi-drop industrial applications.

12.3 FIFOSEL, CLKSEL and INTSEL Configuration

The bits UARTx_FIFOSEL, UARTx_CLKSEL and UARTx_INTSEL (x is 0 for UART0, and 1 for UART1) in the system register MSC0CFG are used to configure several settings in the two UARTs as follow:

- UARTx_FIFOSEL (together with UART_EFR and UART_FCR registers) is used to select the UART operation mode (see Table 12.2).

Copyright © Bridgetek Pte Ltd

- UARTx_CLKSEL is used to enable/disable the pre-scaler.
- UARTx_INTSEL is used to enable/disable the internal UART interrupt signal to the CPU. When UARTx_INTSEL is 1, the interrupt signal is enabled. When UARTx_INTSEL is 0, the interrupt signal is controlled by UART_MCR[3]. It is enabled when UART_MCR[3] is set, and disabled when UART_MCR[3] is cleared.

UART Mode	FIFOSEL	INTSEL	CLKSEL
450	X	1 enables interrupt	1 disables pre-scaler
550	0	1 enables interrupt	1 disables pre-scaler
Extended 550	1	1 enables interrupt	1 disables pre-scaler
650	X	1 enables interrupt	1 disables pre-scaler
750	0	1 enables interrupt	1 disables pre-scaler
950	X	1 enables interrupt	1 disables pre-scaler

Table 12.3 – FIFOSEL, CLKSEL and INTSEL configuration

12.4 Baud Rate Generation

The FT90X UARTs contains a programmable 16-bit baud rate generator that divides the system input clock (i.e. 100 MHz) by a divisor in the range between 1 and (2^{16} - 1). The 16-bit divisor is written into the UART_DLM (MSB) and UART_DLL (LSB) registers, and must be loaded during the initialization of the UART in order to ensure desired operation of the baud rate generator. In addition to this, a clock pre-scaler is provided which can further divide the clock by values in the range 1 to 31 in steps of 1. Also, a further feature is the Times Clock Register (UART_TCR) which allows the sampling clock (SC) to be set to any value between 4 and 16.

The actual transmitter and receiver baud rate is calculated as follows:

$$Baud\ Rate = \frac{Input\ Clock}{SC * Divisor * Prescaler}$$

Where:

Input Clock = 100 MHz (for FT90x series)

SC = Sample Clock values defined in UART_TCR[3:0]

Divisor = UART_DLL + (256 x UART_DLM)

Pre-scaler = 1 when UART_MCR[7] = 0, else:

= UART_CPR[7:3]

Note: UART_MCR[7] is complement of UARTx_CLKSEL after reset but may be overwritten by software.

The value of SC is decided based on UART_TCR[3:0] as follows.

UART_TCR[3:0]	Clock Cycles
0000 to 0011	16
0100 to 1111	4 – 15

Table 12.4 – Sample Clock Values

12.5 Standard 16550 Compatible Registers

12.5.1 UART_RBR – Receiver Buffer Register (address offset: 0x00, requires UART_LCR[7] = 0)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	FIFO Data Read from RX Buffer

Table 12.5 – UART_RBR – Receiver Buffer Register

Copyright © Bridgetek Pte Ltd

12.5.2 UART_THR – Transmitter Holding Register (address offset: 0x00, requires UART_LCR[7] = 0)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	FIFO Data Write to TX Buffer

Table 12.6 – UART_THR – Transmitter Holding Register

12.5.3 UART_DLL – Divisor Latch LSB Register (address offset: 0x00, requires UART_LCR[7] = 1)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	The 8 least-significant bits (LSB) of the 16-bit divisor for generation of the baud clock in the baud rate generator.

Table 12.7 – UART_DIV_LSB – Divisor LSB Register

12.5.4 UART_DLM – Divisor Latch MSB Register (address offset: 0x01, requires UART_LCR[7] = 1)

Bit	Name	Type	Default Value	Description
7:0	DATA	WO	8'h00	The 8 most-significant bits (MSB) of the 16-bit divisor for generation of the baud clock in the baud rate generator.

Table 12.8 – UART_DIV_MSB – Divisor MSB Register

12.5.5 UART_IER – Interrupt Enable Register (address offset: 0x01, requires UART_LCR[7] = 0 and UART_ACR[7] = 0)

Bit	Name	Type	Default Value	Description
7	CTS_EN	RW	1'h0	Enable CTS interrupt (when UART_EFR[4] = 1)
6	RTS_EN	RW	1'h0	Enable RTS interrupt (when UART_EFR[4] = 1)
5	SC_EN	RW	1'h0	Enable Special Character interrupt Mask or Alternate Sleep Mode
4	SM_EN	RW	1'h0	Enable Sleep Mode
3	MOD_STS_EN	RW	1'h0	Enable Modem Status Interrupt
2	LINE_STS_EN	RW	1'h0	Enable Receiver Line Status Interrupt
1	TX_EMTY_EN	RW	1'h0	Enable Transmitter Holding Register Empty Interrupt
0	RX_AVL_EN	RW	1'h0	Enable Received Data Available Interrupt

Table 12.9 – UART_IER – Interrupt Enable Register

12.5.6 UART_ISR – Interrupt Status Register (address offset: 0x02, requires that the last value written to UART_LCR was not 0xBF)

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	Reserved
5:4	Interrupt priority (Enhanced mode)	RO	2'h0	<Refer to Table 12.11 below>
3:1	Interrupt priority (All modes)	RO	3'h0	<Refer to Table 12.11 below>
0	Interrupt pending	RO	1'h0	<Refer to Table 12.11 below>

Table 12.10 – UART_ ISR – Interrupt Status Register

Copyright © Bridgetek Pte Ltd

In order to provide minimum software overhead during data character transfers, the UART prioritizes interrupts into six levels, with lower level having higher priority, and records these in the Interrupt Status Register. When the CPU accesses the Interrupt Status Register, the UART freezes all interrupts and indicates the highest priority pending interrupt to the CPU. While this CPU access is occurring, the UART records new interrupt, but do not change its current indication until the access is complete. The table below shows the contents of the Interrupt Status Register.

UART_ ISR[5:0][2]	Interrupt Function		
	Interrupt Type	**Interrupt Source**	**Level**
000001	None	No interrupt pending[1]	-
000110	Receiver Line Status or address-bit detected in 9-bit mode	Overrun Error or Parity Error or Framing Error or Break Interrupt or address-bit detected in 9-bit mode	1
000100	Receiver Data Available	The receiver FIFO level is above the interrupt trigger level	2a
001100	Receiver time-out	There has been no read of UART_RBR or a period of time greater than the time-out period. There has been no new data received and written into the UART_RBR for a period of time greater than the time-out period.[4]	2b
000010	Transmitter empty	Transmitter Holding Register Empty	3
000000	Modem status change	Clear to Send or Data Set Ready or Ring Indicator or Data Carrier Detect	4
010000	In-band flow control XOFF **or** special character (XOFF2) **or** special character 1,2,3 or 4 **or** bit 9 set in 9-bit mode	Valid XOFF, valid XOFF2 or matches special character 1, 2, 3 or 4 (only in Enhanced mode)	5[3]
100000	CTS or RTS change of state	When CTS or RTS bits will change	6[3]

Table 12.11 – Interrupt Status Register Software Handling

Notes:
1 - UART_ISR[0] indicates whether any interrupts are pending.
2 - Interrupts of priority levels 5 and 6 cannot occur unless the UART is in Enhanced mode
3 - UART_ISR[5] is only used in 16650 & 16950 modes. In 16750 mode, it is 0 when FIFO size is 16, and 1 when FIFO size is 128. In all other modes, it is permanently set to 0.
4 – The timeout period is four times the character period (including start and stop bits) measured from the center of the first stop bit of the last data received.

12.5.7 UART_FCR – FIFO Control Register (address offset: 0x02, requires that the last value written to UART_LCR was not 0xBF)

16550 AND 16750 MODE

Bit	Name	Type	Default Value	Description
7:6	RCVR_TRIG	WO	2'h0	Receiver FIFO Trigger. See Table 12.12.
5	FIFO_SIZE	WO	1'h0	Enable UART support for 128 Byte deep FIFO's **Note:** This bit is guarded with UART_LCR[7] = 1
4:3	Reserved	WO	2'h0	

Copyright © Bridgetek Pte Ltd

2	TXMT_RST	WO	1'h0	Transmitter FIFO reset. Writing a 1 to this bit clears all bytes in the Transmitter FIFO, and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.
1	RCVR_RST	WO	1'h0	Receiver FIFO reset. Writing a 1 to this bit clears all bytes in the Receiver FIFO, and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.
0	FIFO_EN	WO	1'h0	Receiver and Transmitter FIFO's enable.

Table 12.12 – UART_FCR – FIFO Control Register – 16550/16750 mode

This is a write-only register at the same location as UART_ISR (UART_ISR is a read only register). Readable contents of this register are placed in register UART_ICR at offset 0x0F. This register is used to enable the FIFOs, clear the FIFOs and set the FIFO triggers levels. When changing from the FIFO Mode to UART Mode and vice versa, data is automatically cleared from the FIFOs.

UART_FCR[7:6] (RCVR_TRIG) are used to set the trigger level for the Receiver FIFO interrupt.

UART_FCR[7:6]	RECEIVER FIFO TRIGGER LEVEL	
	Standard FIFO* (16 Bytes)	Extended FIFO* (128 Bytes)
00	1	1
01	4	32
10	8	64
11	14	112

Table 12.13 – UART_RCVR – FIFO Trigger Level – 16550/16750 mode

***** - depends on FCR[5] (FIFO_SIZE) bit value.

In these modes (16550 and 16750), the transmitter trigger level is fixed to 1.

16650 MODE

Bit	Name	Type	Default Value	Description
7:6	RCVR_TRIG	WO	2'h0	Receiver FIFO Trigger
5:4	THR_TRIG	WO	1'h0	Transmitter FIFO Trigger
3	Reserved	-	2'h0	-
2	TXMT_RST	WO	1'h0	Transmitter FIFO reset. Writing a 1 to this bit clears all bytes in the Transmitter FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.
1	RCVR_RST	WO	1'h0	Receiver FIFO reset. Writing a 1 to this bit clears all bytes in the Receiver FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.
0	FIFO_EN	WO	1'h0	Receiver and Transmitter FIFO's enable

Table 12.14 - UART_FCR – FIFO Control Register – 16650 mode

This is a write only register at the same location as the UART_ISR (UART_ISR is a read only register). This register is used to enable the FIFOs, clear the FIFOs, set the RCVR FIFO and THR FIFO triggers level. When changing from the FIFO Mode to UART Mode and vice versa, data is automatically cleared from the FIFOs.

UART_FCR[7:6] (RCVR_TRIG) are used to set the trigger level for the Receiver FIFO interrupt and flow control.

Copyright © Bridgetek Pte Ltd

UART_FCR[7:6]	RECEIVER FIFO TRIGGER LEVEL	
	Lower trigger for flow control	Interrupt trigger/upper trigger for flow control
00	1	16
01	16	32
10	32	112
11	112	120

Table 12.15 - Receiver FIFO Trigger Level in 16650 Mode

UART_FCR[5:4] (THR_TRIG) are used to set the trigger level for the Transmitter FIFO interrupt.

UART_FCR[5:4]	TRANSMITTER FIFO TRIGGER LEVEL
00	16
01	32
10	64
11	112

Table 12.16 - Transmitter FIFO Trigger Level in 16650 Mode

16950 MODE

When UART_ACR[5] = 1, bits UART_FCR[5:4] and UART_FCR[7:6] are ignored, and the transmitter trigger level can be defined by UART_TTL (transmitter) and UART_RTL (receiver) registers. The trigger level determined by UART_TTL and UART_RTL may be from 0 to 127. There are also UART_FCH and UART_FCL registers used for specifying trigger levels for the flow control feature.

Setting 0x00 to UART_TTL causes an interrupt to occur when the FIFO and the transmitter shift register are both empty and the serial output signal is in the idle state.

12.5.8 UART_LCR – Line Control Register (address offset: 0x03, reading requires UART_ACR[7] = 0)

Bit	Name	Type	Default Value	Description
7	DLAB	RW	1'h0	Divisor Latch Access Bit 0: Receiver and Transmitter Buffers enable 1: Divisor Latch Enable
6	SET_BRK	RW	1'h0	When set, the transmitter is switched into break state, the serial output signal is driven into logic 0 state.
5	SET_PARITY	RW	1'h0	Stick Parity (explanation below) 0: Disable Stick Parity 1: Enable Stick Parity When bits 3, 4 and 5 are logic 1, the Parity bit is transmitted and checked as a logic 0. If bits 3 and 5 are 1 and bit 4 is a logic 0, then the Parity bit is transmitted and checked as a logic 1. If bit 5 is a logic 0, Stick Parity is disabled.
4	EVEN_PARITY	RW	1'h0	Even Parity Select 0: An odd number of logic ones is checked for in the transmitted and received character. 1: An even number of logic ones is checked for in the transmitted and received character.
3	PARITY_EN	RW	1'h0	Parity Enable 1: parity enabled 0: parity disabled

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
2	STOP_BITS	RW	1'h0	Number of STOP bits 0: 1 Stop bit generated 1: 1.5 stop bits generated for 5 data bits, or 2 STOP bits generated for 6/7/8 data bits.
1:0	WORD_LEN	RW	1'h0	Word Length select bits 00: 5 data bits 01: 6 data bits 10: 7 data bits 11: 8 data bits

Table 12.17 – UART_LCR – Line Control Register

12.5.9 UART_MCR – Modem Control Register (address offset: 0x04, requires that the last value written to UART_LCR was not 0xBF, reading requires UART_ACR[7] = 0)

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Default Value</th><th>Description</th></tr>
<tr><td>7</td><td>BAUD_PSCL*</td><td>RW</td><td>1'h0</td><td>Baud rate pre-scaler select

Writing a 0 to this bit sets the clock divider in the baud generator to 1, else the divider is equal to UART_CPR[7:3].</td></tr>
<tr><td>6</td><td>IRDA_MODE*</td><td>RW</td><td>1'h0</td><td>IrDA mode

This bit is only available in 16650 or 16950 mode. A '1' on this bit enables IrDA mode, which transfers received and transmitted data in the IrDA format.</td></tr>
<tr><td>5</td><td>AFE_XON*</td><td>RW</td><td>1'h0</td><td>Auto Flow Control enable (AFE) / "XON any" enable.

1: auto flow control enabled
0: disable auto flow control

<table>
<tr><th>UART_MCR [5] (AFE_XON)</th><th>UART_MCR[1] (RTS)</th><th>Flow Control Configuration</th></tr>
<tr><td>1</td><td>1</td><td>Auto-RTS and Auto-CTS are enabled</td></tr>
<tr><td>1</td><td>0</td><td>Only Auto-CTS is enabled</td></tr>
<tr><td>0</td><td>…</td><td>Both Auto-RTS and Auto-CTS are disabled</td></tr>
</table>

In Enhanced mode this bit is used for enabling the "XON any" feature. This feature allows re-enabling transmission in case of receiving any character in Automatic In-band Flow Control mode.</td></tr>
<tr><td>4</td><td>LOOPBK_EN</td><td>RW</td><td>1'h0</td><td>Enable Loopback bit

1: This bit provides a local loop-back feature for diagnostic testing of the UART. In the loop-back mode, the output of the Transmitter Shift Register is "looped back" into the Receiver Shift Register input. As a result, transmitted data is immediately received. This feature allows the CPU to verify the transmit and received data paths of the UART. In the loop-back mode, the receiver and transmitter interrupts are fully operational. The interrupts are still controlled by the UART_IER register.</td></tr>
<tr><td>3:2</td><td>Reserved</td><td>RW</td><td>1'h0</td><td>Reserved. This bit should be '0'.</td></tr>
</table>

Copyright © Bridgetek Pte Ltd

1	RTS	RW	1'h0	Request to send 1: the RTS output is forced to a logic 0 0: the RTS output is forced to a logic 1
0	DTR	RW	1'h0	Data Terminal Ready 1: the DTR output is forced to a logic 0 0: the DTR output is forced to a logic 1

Table 12.18 – UART_MCR – Modem Control Register

Note: * denotes only in 16650/16950 mode

12.5.10 UART_LSR – Line Status Register (address offset: 0x05, requires that the last value written to UART_LCR was not 0xBF, reading requires UART_ACR[6] = 0)

Bit	Name	Type	Default Value	Description
7	RBR_ERR	RO	1'h0	Error in UART_RBR / Error in Receiver FIFO In the UART Mode, this is a 0. In the FIFO mode: 1: when there is at least one parity error, framing error or break indication in the FIFO. 0: when LSR is read. In 16450 mode, this bit is permanently cleared. In 9-bit mode, this bit is not affected by UART_LSR[2].
6	TX_EMTY	RO	1'h0	**Transmitter Empty** 1: whenever the Transmitter Holding Register (THR) and the Transmitter Shift Register are both empty. 0: whenever either the THR or TSR contains a data character. **In the FIFO mode,** 1: whenever the transmitter FIFO and shift register are both empty
5	TXH_EMTY	RO	1'h0	**Transmitter Holding Register Empty** 1: Transmitter Holding Register is Empty 0: Transmitter Holding Register has data **In the FIFO mode**, 1: XMIT FIFO is empty 0: at least 1 byte is written to the XMIT FIFO
4	BRK_INT	RO	1'h0	Break Interrupt 1: whenever the received data input is held at logic 0 for longer than a full word transmission time (that is, the total time of Start bit, data bits, a Parity and Stop bits). 0: whenever the CPU reads the contents of the Line Status Register. In the FIFO mode, this error is associated with the particular character in the FIFO it applies to. This error is revealed to the CPU when its associated character is at the top of the FIFO. When break occurs, only one zero character is loaded into the FIFO. The next character transfer is enabled after the received data input goes to logic 1 and receives the next valid start bit.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
3	FRM_ERR	RO	1'h0	Framing Error Indicates that the received character does not have a valid Stop bit. 1: whenever the Stop bit following the last data bit or parity bit is detected as a logic 0 bit. 0: whenever the CPU reads the contents of the Line Status Register. In the FIFO mode, this error is associated with the particular character in the FIFO it applies to. This error is revealed to the CPU when its associated character is at the top of the FIFO. The UART will try to resynchronize after a framing error.
2	PRTY_ERR_RX_9BIT	RO	1'h0	Parity Error/ 9-bit of received data in UART_RBR 1: upon detection of a parity error 0: whenever the CPU reads the contents of the Line Status Register. In the FIFO mode, this error is associated with the particular character in the FIFO it applies to. This error is revealed to the CPU when its associated character is at the top of the FIFO. In 9-bit mode, this bit is the 9th bit of the received data, in addition to the 8 bits in UART_RBR.
1	OVRN_ERR	RO	1'h0	Overrun Error 1: upon detection of an overrun condition 0: whenever the CPU reads the contents of the Line Status Register. If the FIFO mode data continues to fill the FIFO beyond the trigger level, an overrun error will occur only after the FIFO is full and the next character has been completely received in the shift register. OVRN_ERR is indicated to CPU as soon as it happens. The character in the shift register is overwritten, but it is not transferred to the FIFO.
0	DATA_RDY	RO	1'h0	Data Ready 1: a complete incoming character has been received and transferred into the Receiver Buffer Register or the FIFO. 0: by reading all of the data in the Receiver Buffer Register or the FIFO

Table 12.19 – UART_LSR – Line Status Register

Copyright © Bridgetek Pte Ltd

12.5.11 UART_MSR – Modem Status Register (address offset: 0x06, requires that the last value written to UART_LCR was not 0xBF)

Bit	Name	Type	Default Value	Description
7	DCD	RO	1'h0	Data Carrier Detect This bit is the complement of the Data Carrier Detect (DCD) input. If bit 4 of the Modem Control Register is set to a 1, this bit is equivalent to UART_MCR[3].
6	RI	RO	1'h0	Ring Indicator This bit is the complement of the Ring Indicator (RI) input. If bit 4 of the Modem Control Register is set to a 1, this bit is equivalent to UART_MCR[2].
5	DSR	RO	1'h0	Data Set Ready This bit is the complement of the Data Set Ready (DSR) input. If bit 4 (LOOPBK_EN) of the Modem Control Register is set to a 1, this bit is equivalent to DTR in the Modem Control Register.
4	CTS	RO	1'h0	Clear to Send This bit is the complement of the Clear to Send (CTS) input. If bit 4 (LOOPBK_EN) of the Modem Control Register is set to a 1, this bit is equivalent to RTS in the Modem Control Register.
3	D_DCD	RO	1'h0	Delta Data Carrier Detect 1: indicates that the DCD input to the chip has changed state. **Note:** Whenever bit 0, 1, 2, or 3 is set to logic 1, a MODEM Status Interrupt is generated.
2	TERI	RO	1'h0	Trailing Edge Ring Indicator 1: Indicates that the RI input to the chip has changed from a low to a high state.
1	D_DSR	RO	1'h0	Delta Data Set ready 1: Indicates that the DSR input to the chip has changed state since the last time it was read by the CPU.
0	D_CTS	RO	1'h0	Delta Clear To Send 1: Indicates that the CTS input to the chip has changed state since the last time it was read by the CPU.

Table 12.20 – UART_MSR – Modem Status Register

12.5.12 UART_SPR – SPR Register (address offset: 0x07, requires that the last value written to UART_LCR was not 0xBF)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	This is the command port for the indexed control register. Writing and reading to indexed control register is addressed by offset written to this register

Copyright © Bridgetek Pte Ltd

Table 12.21 – UART_SPR – SPR Register

12.6 16650 COMPATIBLE REGISTERS

Note: To access these registers, UART_LCR must be set to 0xBF.

12.6.1 UART_EFR – Enhanced Feature Register (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7	CTS_FC	RO	1'h0	1: Automatic CTS flow control enables
6	RTS_FC	RO	1'h0	1: Automatic RTS flow control enables
5	SPL_CHAR	RO	1'h0	1: Special Character Detection mode enable
4	EM	RO	1'h0	1: Enhanced mode enable
3:2	IBT_FCM	RO	1'h0	In-band transmit flow control mode 00: Disable in-band flow control 01: Enable single character in-band transmit flow control. Recognizing XON2 as the XON character and XOFF2 as the XOFF character 10: Enable single character in-band transmit flow control. Recognizing XON1 as the XON character and XOFF1 as the XOFF character 11: Reserved
1:0	IBR_FCM	RO	1'h0	In-band receive flow control mode 00: Disable in-band flow control 01: Enable single character in-band receive flow control. Recognizing XON2 as the XON character and XOFF2 as the XOFF character 10: Enable single character in-band receive flow control. Recognizing XON1 as the XON character and XOFF1 as the XOFF character 11: Reserved

Table 12.22 – UART_EFR – Enhanced Feature Register

12.6.2 UART_XON1 – XON1 Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Value of XON1

Table 12.23 – UART_XON1 – XON1 Register

12.6.3 UART_XON2 – XON2 Register (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Value of XON2

Table 12.24 – UART_XON2 – XON2 Register

12.6.4 UART_XOFF1 – XOFF1 Register (address offset: 0x06)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Value of XOFF1

Table 12.25 – UART_XOFF1 – XOFF1 Register

Copyright © Bridgetek Pte Ltd

12.6.5 UART_XOFF2 – XOFF2 Register (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Value of XOFF2

Table 12.26 – UART_XOFF2 – XOFF2 Register

12.7 16950 COMPATIBLE REGISTERS

Note: To access these registers, UART_ACR[7] must be set to 1.

12.7.1 UART_ASR – Additional Status Register (address offset: 0x01, requires UART_LCR[7] = 0)

Bit	Name	Type	Default Value	Description
7	TX_IDLE	RO	1'h0	1: transmitter is idle (transmitter FIFO and shift register are empty) 0: transmitter is transmitting.
6	FIFO_SIZE	RO	1'h0	1: FIFO are 16 deep if UART_FCR[0] = 1 0: FIFO are 128 deep if UART_FCR[0] = 1
5	FIFO_SEL	RO	1'h0	Actual state of the UARTx_FIFOSEL bit
4	SPECIAL_CHAR_DETECT	RO	1'h0	1: special character detects and is stored in UART_RBR 0: no special character detect The flag is cleared by reading UART_ASR.
3	DTR	RO	1'h0	Complement state of DTR pin
2	RTS	RO	1'h0	Complement state of RTS pin
1	REMOTE_TX_DSBL	RW	1'h0	1: transmitter has sent an XOFF character 0: remote transmitter is not disabled by in-band flow control. This bit may be cleared by software to re-enable remote transmitter (XON is sent).
0	TX_DSBL	RW	1'h0	1: transmitter disabled (receiver detect XOFF) 0: transmitter is not disabled by in-band flow control. This bit may be cleared by software to re-enable transmission if it was disabled by in-band flow control.

Table 12.27 - UART_ASR – Additional Status Register

12.7.2 UART_RFL – Receiver FIFO Level Register (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	Number of characters in the Receiver FIFO

Table 12.28 - UART_RFL – Receiver FIFO Level Register

12.7.3 UART_TFL – Transmitter FIFO Level Register (address offset: 0x04, requires that the last value written to UART_LCR was not 0xBF)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	Number of characters in the Transmitter FIFO

Table 12.29 - UART_TFL – Transmitter FIFO Level Register

Copyright © Bridgetek Pte Ltd

12.7.4 UART_ICR – ICR Register (address offset: 0x05, requires that the last value written to UART_LCR was not 0xBF, reading requires UART_ACR[6] = 1)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Data port to the indexed control register. Writing data to the indexed control register and reading data from the indexed control register is done using this register.

Table 12.30 - UART_ICR – ICR Register

12.8 INDEXED CONTROL REGISTERS

Writing to Indexed Control Registers (ICRs) is addressed by the offset in UART_SPR, and data is loaded through the UART_ICR register. Before writing, be sure that UART_LCR was not loaded with '0xBF' (it enables access to 16650 compatible registers).To write ICRs, please follow these steps:

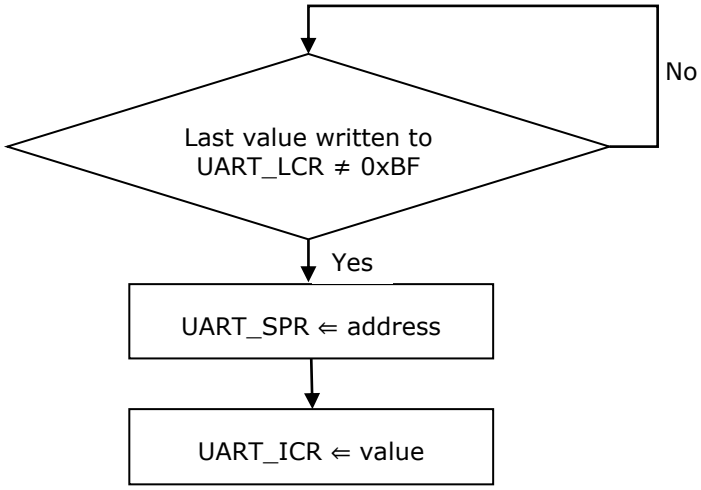


Figure 12.1 – ICR registers write access

Reading from Indexed Control Registers is addressed by the offset in UART_SPR, and data is read through the UART_ICR register. Before reading, be sure that UART_LCR was not loaded with '0xBF' (it enables access to 16650 compatible registers). To read from ICRs, please follow these steps:

Copyright © Bridgetek Pte Ltd

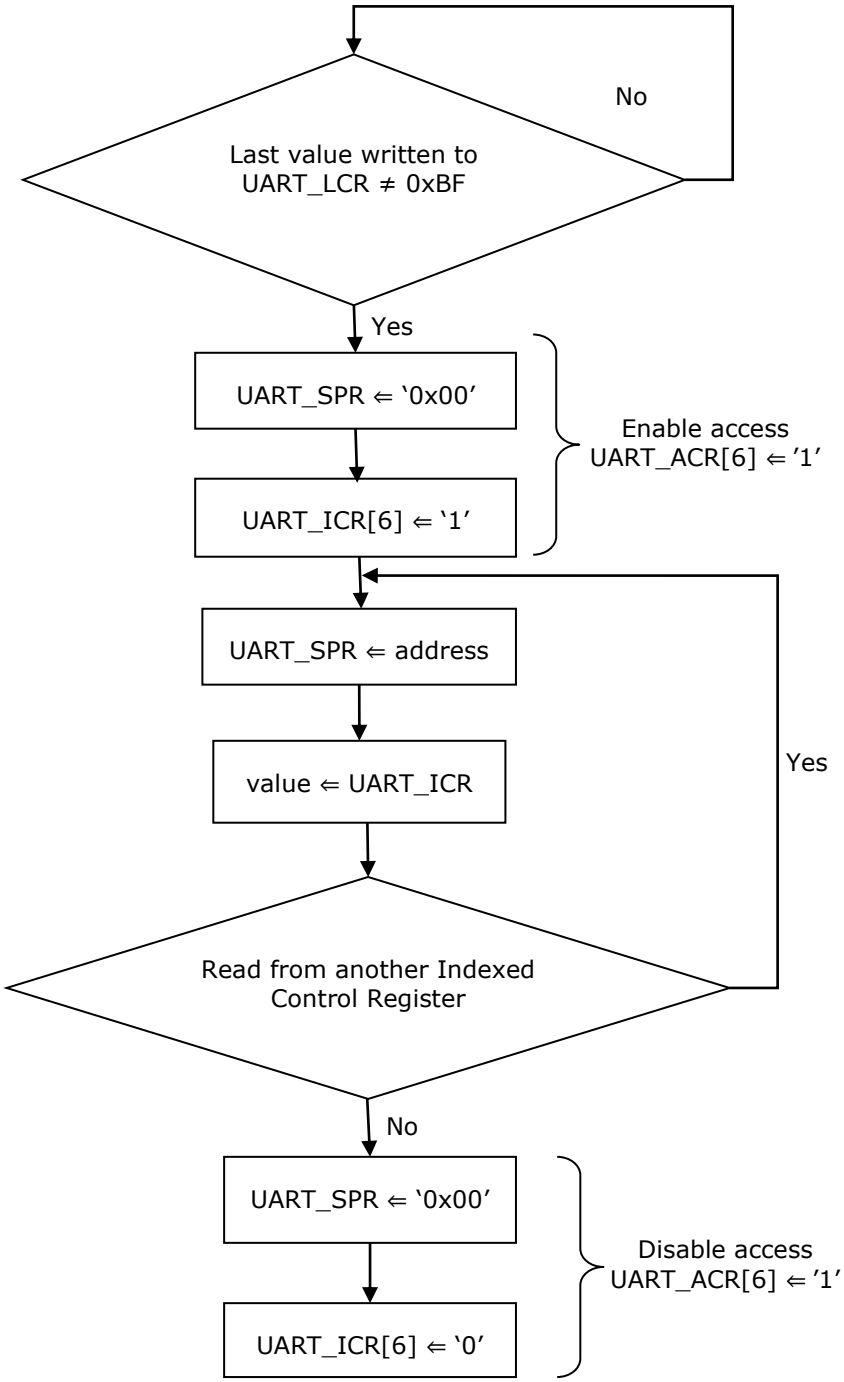


Figure 12.2 – ICR registers read access

Copyright © Bridgetek Pte Ltd

12.8.1 UART_ACR – Additional Control Register (SPR offset: 0x00)

Bit	Name	Type	Default Value	Description
7	ADL_STS_EN	RW	1'h0	Additional status enable 1: UART_ASR, UART_TFL and UART_RFL are enabled 0: disabled
6	ICR_RD_EN	RW	1'h0	ICR read enable 0: UART_LSR is readable 1: ICR registers are readable
5	950_TLE	RW	1'h0	0: Interrupts and flow control trigger levels are described in the UART_FCR register 1: The triggers are set by UART_RTL, UART_TTL, UART_FCL and UART_FCH registers.
4:3	DTR	RW	1'h0	DTR line configuration When UART_CKS[4] or UART_CKS[5] are set, the transmitter 1x clock or the output of the baud rate generator (Nx clock) are asserted on the DTR pin, otherwise the pin is defined as follows: 00: DTR pin is compatible with 16450, 16550, 16650 and 16750 (i.e. normal). 01: DTR pin is used for out-of-band flow control. It will be forced high, when the receiver FIFO level reaches the upper flow control trigger (UART_FCH). It will be forced low when the receiver FIFO level falls below the lower flow control trigger (UART_FCL). 10: DTR is configured to drive the active low enable pin of an external RS485 buffer. The pin will be forced low whenever the transmitter is not empty (UART_LSR[6]=0), otherwise the pin is high. 11: DTR is configured to drive the active high enable pin of the external RS485 buffer. The pin will be forced high whenever the transmitter is not empty (UART_LSR[6]=0), otherwise the pin is low.
2	DSR	RW	1'h0	1: enables automatic out-of-band flow control using the DSR pin.
1	TX	RW	1'h0	Transmitter disable. 0: Transmitter is enabled. 1: Transmitter is disabled. Data in UART_THR are not transmitted. In-band flow control characters may still be transmitted.
0	RX	RW	1'h0	Receiver disable. 0: Receiver is enabled and data are stored in UART_RBR. 1: Receiver is disabled. The receiver continues to operate as normal to maintain frame synchronization but received data are not stored. In-band control characters continue to be detected and acted upon. Special characters will not be detected.

Table 12.31 – UART_ACR- Additional Control Register

Copyright © Bridgetek Pte Ltd

12.8.2 UART_CPR – Clock Pre-scaler Register (SPR offset: 0x01)

Bit	Name	Type	Default Value	Description
7:3	PSCL	RW	5'h00	Clock pre-scaler
2:0	Reserved	-	-	-

Table 12.32 – UART_CPR – Clock Pre-scaler Register

12.8.3 UART_TCR – Times Clock Register (SPR offset: 0x02)

Bit	Name	Type	Default Value	Description
7:4	Reserved	-	-	-
3:0	N_TIMES_CLOCK	RW	4'h00	Sample Clock value for Baud Rate generation

Table 12.33 – UART_TCR – Time Clock Register

12.8.4 UART_CKS – Clock Select Register (SPR offset: 0x03)

Bit	Name	Type	Default Value	Description
7	TX_CLK_MD	RW	1'h0	Transmitter 1x clock mode selector 0: transmitter clock is in Nx clock mode 1: transmitter is in isochronous 1x clock mode
6	TX_CLK_SRC	RW	1'h0	Transmitter clock source selector 0: transmitter clock source is the output of the baud rate generator 1: transmitter uses external clock applied to the RI pin
5:4	TX_CLK_GEN	RW	2'h0	Transmitter 1x clock or baud rate generator output (BAUD_OUT) on the DTR pin. 00: The function of the DTR pin is defined by the setting of UART_ACR[4:3]. 01: The transmitter 1x clock is asserted on the DTR pin and setting of UART_ACR[4:3] is ignored. 10: The output of baud rate generator (Nx clock) is asserted on the DTR pin and the setting of UART_ACR[4:3] is ignored. 11: Reserved
3	RX_CLK_MD	RW	1'h0	Receiver 1x clock mode selector 0: Receiver clock is in Nx clock mode 1: Receiver is in isochronous 1x clock mode
2	BAUD_OUT	RW	1'h0	Disable BAUD_OUT pin 0: BAUD_OUT is enabled and connected to the baud rate generator which is Nx clock. By default it is the 16x clock but using the TCR register it may be configured in range from 4x to 16x clock. 1: BAUD_OUT is disabled and permanently set to logic 0
1:0	RX_CLK_SRC	RW	2'h0	Receiver Clock source selector 00: The RCLK pin is selected for the receiver clock 01: The DSR pin is selected for the receiver clock 10: The baud rate generator output is selected for the receiver clock (internal BAUD_OUT connection) 11: The transmitter clock is selected for the receiver clock

Table 12.34 – UART_CKS Clock Select Register

Copyright © Bridgetek Pte Ltd

12.8.5 UART_TTL – Transmitter Trigger Level Register (SPR offset: 0x04)

This register is located at 0x04 offset of the Indexed Control Register. This register is used for storing the interrupt trigger level for the transmitter in 16950 mode (UART_ACR[5] = 1). The interrupt occurs (if enabled) when the transmitter FIFO level falls below the value of the UART_TTL register. If UART_TTL = 0, then an interrupt will occur when both FIFO and shift register are empty and the SO line is marked to be in the idle state.

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	TRIG_LVL	RW	7'h0	Transmitter Trigger Level The interrupt trigger level for the transmitter in 16950 mode (UART_ACR[5] = 1)

Table 12.35 – UART_TTL – Transmitter Trigger Level Register

12.8.6 UART_RTL – Receiver Trigger Level Register (SPR offset: 0x05)

The UART_RTL register is located at 0x05 offset of the ICR. This register is used for storing the interrupt trigger level for the receiver in 16950 mode (ACR[5] = 1). The interrupt occurs (if enabled) when the receiver FIFO level reaches the value stored in this register.

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	TRIG_LVL	RW	7'h0	Receiver Trigger Level The interrupt trigger level for the receiver in 16950 mode (UART_ACR[5] = 1)

Table 12.36 – UART_RTL – Receiver Trigger Level Register

12.8.7 UART_FCL – Flow Control Level LSB Register (SPR offset: 0x06)

Automatic flow control is supported by UART_FCL and UART_FCH registers. These registers are active only in Enhanced mode, when UART_FCR[6:7] bits are disabled (UART_ACR[5] = 1). The UART_FCL stores the lower trigger level, and UART_FCH stores the upper trigger level. Both registers are able to store level values from 0 to 127.

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	FLW_CNTL	RW	7'h0	Value of Flow Control LSB

Table 12.37 – UART_FCL – Flow Control Level LSB Register

12.8.8 UART_FCH – Flow Control Level Register MSB (SPR offset: 0x07)

Automatic flow control is supported by UART_FCL and UART_FCH registers. These registers are active only in Enhanced mode, when UART_FCR[6:7] bits are disabled (UART_ACR[5] = 1). The UART_FCL register stores the lower trigger level, and UART_FCH stores the upper trigger level. Both registers are able to store level values from 0 to 127.

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	FLW_CNTL	RW	7'h0	Value of Flow Control MSB

Table 12.38 – UART_FCH – Flow Control Level Register MSB

12.8.9 UART_ID1 – Identification 1 Register (SPR offset: 0x08)

To identify the device type, use ID1,ID2,ID3 and REV registers. In ID1,ID2 and ID3 registers, a hexadecimal ID of the device is written. The REV register includes a hardware revision sign.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h0	Identification value 1

Table 12.39 – UART_ID1 – Identification 1 Register

12.8.10 UART_ID2 – Identification 2 Register (SPR offset: 0x09)

To identify the device type, use UART_ID1, UART_ID2, UART_ID3 and UART_REV registers. In UART_ID1, UART_ID2 and UART_ID3 registers, a hexadecimal ID of the device is written.

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h0	Identification value 2

Table 12.40 – UART_ID2 – Identification 2 Register

12.8.11 UART_ID3 – Identification 3 Register (SPR offset: 0x0A)

To identify the device type, use UART_ID1, UART_ID2, UART_ID3 and UART_REV registers. In UART_ID1, UART_ID2 and UART_ID3 registers, a hexadecimal ID of the device is written.

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h0	Identification value 3

Table 12.41 UART_ID3 – Identification 3 Register

12.8.12 UART_REV – Revision Register (SPR offset: 0x0B)

To identify the device type, use UART_ID1, UART_ID2, UART_ID3 and UART_REV registers. In UART_ID1, UART_ID2 and UART_ID3 registers, a hexadecimal ID of the device is written.

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h0	Hardware revision

Table 12.42 – UART_REV – Revision Register

12.8.13 UART_CSR – Channel Software Reset Register (SPR offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h0	00: To reset the UART write 0x00 to the Channel Software Reset register

Table 12.43 – UART_CSR – Channel Software Reset Register

12.8.14 UART_NMR – Nine Bit Mode Register (SPR offset: 0x0D)

To enable 9-bit data mode, UART_NMR[0] bit must be logic 0. In this mode, data are nine bits wide, and the 9th bit is stored in UART_LSR[2] for receiving. In transmission, the 9th bit should be written in UART_SPR[0] bit before writing 8-bit data to THR.

In 9-bit mode, the setting in UART_LCR[1:0] are ignored. Furthermore, as parity is permanently disabled, the setting of UART_LCR[5:3] is also ignored.

In 9-bit mode, in-band flow control is disabled.

When the UART is configured for both Enhanced and 9-bit data mode, setting UART_IER[5] will enable detection of up to four 'address' characters. The eight least significant bits of these characters are stored in UART_XON1, UART_XON2, UART_XOFF1 and UART_XOFF2 registers. The 9th bit of these characters is stored in UART_NMR[5] to UART_NMR[2].

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5	9_SC4	RW	1'h0	9-th bit of special character 4
4	9_SC3	RW	1'h0	9-th bit of special character 3
3	9_SC2	RW	1'h0	9-th bit of special character 2
2	9_SC1	RW	1'h0	9-th bit of special character 1
1	9_INT_EN	RW	1'h0	9-bit data mode interrupt enable 0: interrupt for detection of an 'address' character is disabled 1: interrupt for detection of an 'address' character is enabled
0	9_EN	RW	1'h0	9-bit data mode enable

Table 12.44 – UART_NMR – Nine Bit Mode Register

12.8.15 UART_MDM – Modem Disable Mask Register (SPR offset: 0x0E)

UART_MDM is used to mask selected interrupts of modem lines.

Bit	Name	Type	Default Value	Description
7:4	Reserved	-	-	-
3	DCD_MASK	RW	1'h0	Delta DCD disable. 0: Enables level 4 interrupt from delta DCD when UART_IER[3]=1 1: Disables level 4 interrupt from delta DCD.
2	RI_MASK	RW	1'h0	Trailing edge RI disable. 0: Enables level 4 interrupt from trailing edge RI when UART_IER[3]=1 1: Disables level 4 interrupt from trailing edge RI.
1	DSR_MASK	RW	1'h0	Delta DSR disable. 0: Enables level 4 interrupt from delta DSR when UART_IER[3]=1 1: Disables level 4 interrupt from delta DSR.
0	CTS_MASK	RW	1'h0	Delta CTS disable. 0: Enables level 4 interrupt from delta CTS when UART_IER[3]=1 1: Disables level 4 interrupt from delta CTS.

Table 12.45 – UART_MDM – Modem Disable Mask Register

12.8.16 UART_RFC – Readable FCR Register (SPR offset: 0x0F)

Bit	Name	Type	Default Value	Description
7:0	FCR	RO	8'h00	Read the state of UART_FCR register

Table 12.46 – UART_RFC – Readable FCR Register

12.8.17 UART_GDS – Good Data Status Register (SPR offset: 0x10)

Good data status is set when the following conditions are true:

- ISR reads level 0 (no interrupt), level 2 or 2a (receiver data) or level 3 (THR empty) interrupt.
- UART_LSR[7] is clear.
- UART_LSR[1] is clear.

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	Contains 'good data status' bit on least significant position.

Table 12.47 – UART_GDS – Good Data Status Register

12.8.18 UART_RSRV_1 – Reserved Register 1 (SPR offset: 0x11)

Bit	Name	Type	Default Value	Description
7:0	Reserved	-	-	-

Table 12.48 – UART_RSRV_1 – Reserved Register 1

12.8.19 UART_PIDX – Port Index Register (SPR offset: 0x12)

Bit	Name	Type	Default Value	Description
7:0	DATA	RO	8'h00	The value of this register is always 0.

Table 12.49 – UART_PIDX – Port Index Register

12.8.20 UART_RSRV_2 – Reserved Register 2 (SPR offset: 0x13)

Bit	Name	Type	Default Value	Description
7:0	Reserved	-	-	-

Table 12.50 – UART_RSRV_2 – Reserved Register 2

Copyright © Bridgetek Pte Ltd

13 Timers and Watchdog

FT90X consists of a 32-bit watchdog timer and four 16-bit users' timers.

The watchdog timer is clocked off the main clock and it is started by setting the start_wdg bit in TIMER_CONTROL_2. Once started, the watchdog timer cannot be disabled and shall be periodically restarted by clearing its timer counter. This is called "kicking the watchdog" and is done by setting the clear_wdg bit in TIMER_CONTROL_2. The watchdog timer value is set via TIMER_WDG. TIMER_WDG is a 5-bit register, the value of which represents a 2's exponent. Therefore, the timer is always set as 2^^TIMER_WDG. The watchdog is designed as a two-level watchdog. If the watchdog is not kicked within 2^^TIMER_WDG system clock cycles, the first level timeout occurs and the watchdog interrupt is raised and upon the second expiry of the watchdog timer, a POR occurs.

The first level timeout provides firmware an opportunity to perform any emergency housekeeping and error logging (e.g., the handler may output system state to a debug console or write to storage for post-crash analysis) and the second level timeout recovers the system through a POR. As there are 2 levels of timeouts, then the overall duration to reach POR is twice of 2^^TIMER_WDG.

In order to achieve the 2 levels of timeout leading to POR, the watchdog interrupt handler has to follow a specific sequence. wdt_int_ien bit in TIMER_CONTROL_2 shall be set and a watchdog interrupt handler shall be installed by the firmware. Upon the first level timeout, the watchdog interrupt handler shall clear the wdg_int interrupt status bit in TIMER_CONTROL_2. This will enable the second level timer (duration is same as 2^^TIMER_WDG) to run and lead to POR. The watchdog interrupt handler shall not execute a return from interrupt and instead shall execute a while(1) loop until POR occurs. Executing a return from interrupt from the handler leads to indeterminate system behavior.

The wdt_status bit in WDT_STATUS is a sticky bit and once is set by hardware when the second level timeout occurs. It remains set through a POR and is cleared by hardware when it is read. This bit is used to indicate if a previous POR was caused by a watchdog expiry. Upon every startup, firmware may check this bit to determine if a POR was caused by normal power up or watchdog expiry and take necessary recovery action (e.g., enter failsafe mode).

Four user timers can be clocked off the main clock; each timer has its own 16-bit pre-scaler. These timers can be started, stopped and cleared/initialized. The pre-scalers can be cleared/initialized the same way. Current values of all four user timers can be read from registers (one at a time - common register, multiplexed access). All timers count up or down and signal an interrupt when rolling over. Each of the timers can be configured to be one-shot or in continuous mode. They are initialized from a common register and only one at a time (multiplexed access). If the user timer has already started using its pre-scaler, it cannot be cleared and the command is ignored. Each of the pre-scalers automatically stops after it is cleared individually. It also starts automatically when the corresponding user timer starts using it.

The registers are accessible via 8-bit mode only. To read or write the registers, always start with the lower 8 bits followed by the upper 8 bits. For writing, the actual write occurs only after the upper 8 bits are written. For reading, the upper 8 bits are held in a temporary buffer when the lower 8 bits are being read. Writing or reading the upper 8 bits without accessing the lower 8 bits first may produce erroneous results.

Normal operation:

User timers	Pre-scaler	Watchdog
Select the timer to initialize by writing into the **TIMER_SELECT** register. Write initial/final value first into **TIMER_WRITE_LS** and then into **TIMER_WRITE_MS** registers.	Write initial value first into **TIMER_PRESC_LS** and then into **TIMER_PRESC_MS** registers.	Write initial value into **TIMER_WDG** register to initialize one of the 32 bits of the timer.
Write into **direction** bit in **TIMER_CONTROL_3** register to	N/A	N/A

Copyright © Bridgetek Pte Ltd

select up/down counting.		
Write into **mode** bit in **TIMER_CONTROL_3** register to select mode.	N/A	N/A
Write into **clear_x** bits in **TIMER_CONTROL_4** register to initialize the timer.	Write into **clear_presc** bit in **TIMER_CONTROL_4** register to initialize the pre-scaler.	Write into **clear_wdg** bit in **TIMER_CONTROL_2** register to clear watchdog.
Write into **start_x** bits in **TIMER_CONTROL_1** register to start the timer.	Write into **prescaler_en** bit in **TIMER_CONTROL_2** register to enable pre-scaler and it will automatically start when the timer using it starts.	Write into **start_wdg** bit in **TIMER_CONTROL_2** register to start watchdog.
Select timer you want to read from by writing into **timer_read_sel** bit in **TIMER_SELECT** register. Current value can be read from **TIMER_READ_LS** and **TIMER_READ_MS** registers.	N/A	N/A
Write into **stop_x** bit in **TIMER_CONTROL_1** register to stop the timer.	N/A	N/A

Table 13.1 – Timers/Watchdog Operation

13.1 Register Summary

Listed below are the registers with their offset from the base address (0x10340). All registers can only be accessed via Byte (8-bit) mode.

Offset	Register	Default Value	References
0x00	TIMER_CONTROL_0 – Timers Control Register 0	0x00	Section 13.2.1
0x01	TIMER_CONTROL_1 – Timers Control Register 1	0x00	Section 13.2.2
0x02	TIMER_CONTROL_2 – Timers Control Register 2	0x00	Section 13.2.3
0x03	TIMER_CONTROL_3 – Timers Control Register 3	0x00	Section 13.2.4
0x04	TIMER_CONTROL_4 – Timers Control Register 4	0x00	Section 13.2.5
0x05	TIMER_INT – Timers Interrupt Register	0x00	Section 13.2.6
0x06	TIMER_SELECT – Timers A..D Select Register	0x00	Section 13.2.7
0x07	TIMER_WDG – Watchdog Start Value	0x00	Section 13.2.8
0x08	TIMER_WRITE_LSB – Timer A..D Start Value 7:0	0x00	Section 13.2.9
0x09	TIMER_WRITE_MSB – Timer A..D Start Value 15:8	0x00	Section 13.2.10
0x0A	TIMER_PRESC_LSB – Pre-scaler Start Value 7:0	0x00	Section 13.2.11
0x0B	TIMER_PRESC_MSB – Pre-scaler Start Value 15:8	0x00	Section 13.2.12

Copyright © Bridgetek Pte Ltd

0x0C	TIMER_READ_LSB – Timer A..D Current Value 7:0	0x00	Section 13.2.13
0x0D	TIMER_READ_MSB – Timer A..D Current Value 15:8	0x00	Section 13.2.14
0x0E	TIMER_RESERVED – Reserved Register	-	Section 13.2.15
0x0F	WDT_STATUS – Watchdog Status Register	0x00	Section 13.2.16

Table 13.2 – Overview of Timers/Watchdog Registers

13.2 Register Details

13.2.1 TIMER_CONTROL_0 – Timers Control Register 0 (address offset: 0x00)

Bit	Name	Type	Default Value	Description
1	block_en	RW	1'b0	1: To enable the timer module
0	soft_reset	RWAC	1'b0	1: Write 1 to trigger the reset

Table 13.3 – TIMER_CONTROL_0 – Timers Control Register 0

13.2.2 TIMER_CONTROL_1 – Timers Control Register 1 (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7	stop_d	RWAC	1'b0	1: To trigger stopping timer D
6	stop_c	RWAC	1'b0	1: To trigger stopping timer C
5	stop_b	RWAC	1'b0	1: To trigger stopping timer B
4	stop_a	RWAC	1'b0	1: To trigger stopping timer A
3	start_d	RWAC	1'b0	1: To trigger starting timer D
2	start_c	RWAC	1'b0	1: To trigger starting timer C
1	start_b	RWAC	1'b0	1: To trigger starting timer B
0	start_a	RWAC	1'b0	1: To trigger starting timer A

Table 13.4 – TIMER_CONTROL_1 – Timers Control Register 1

13.2.3 TIMER_CONTROL_2 – Timers Control Register 2 (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7:4	prescaler_en	RW	4'h0	Enables pre-scaler bits for timers D/C/B/A respectively
3	wdg_int_ien	RW	1'b0	1: Enable watch dog interrupt
2	wdg_int	RW1C	1'b0	1: Watch-dog interrupt pending
1	clear_wdg	RWAC	1'b0	1: To trigger clearing watch dog timer
0	start_wdg	RWAC	1'b0	1: To trigger starting watch dog timer

Table 13.5 – TIMER_CONTROL_2 – Timers Control Register 2

Copyright © Bridgetek Pte Ltd

13.2.4 TIMER_CONTROL_3 – Timers Control Register 3 (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7:4	direction	RW	4'h0	Counter direction bits for timers D/C/B/A respectively 1: Up 0: Down
3:0	mode	RW	4'h0	Continuous/1-shot mode bits for timers D/C/B/A respectively 1: 1-shot 0: Continuous

Table 13.6 – TIMER_CONTROL_3 – Timers Control Register 3

13.2.5 TIMER_CONTROL_4 – Timers Control Register 4 (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7	presc_clear_d	RWAC	1'b0	1: To trigger clearing pre-scaler D
6	presc_clear_c	RWAC	1'b0	1: To trigger clearing pre-scaler C
5	presc_clear_b	RWAC	1'b0	1: To trigger clearing pre-scaler B
4	presc_clear_a	RWAC	1'b0	1: To trigger clearing pre-scaler A
3	clear_d	RWAC	1'b0	1: To trigger clearing timer D
2	clear_c	RWAC	1'b0	1: To trigger clearing timer C
1	clear_b	RWAC	1'b0	1: To trigger clearing timer B
0	clear_a	RWAC	1'b0	1: To trigger clearing timer A

Table 13.7 – TIMER_CONTROL_4 – Timers Control Register 4

13.2.6 TIMER_INT – Timers Interrupt Register (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7	timer_int_d_en	RW	1'b0	1: Enables timer D interrupt
6	timer_int_d	RW1C	1'b0	1: Timer D interrupt pending
5	timer_int_c_en	RW	1'b0	1: Enables timer C interrupt
4	timer_int_c	RW1C	1'b0	1: Timer C interrupt pending
3	timer_int_b_en	RW	1'b0	1: Enables timer B interrupt
2	timer_int_b	RW1C	1'b0	1: Timer B interrupt pending
1	timer_int_a_en	RW	1'b0	1: Enables timer A interrupt
0	timer_int_a	RW1C	1'b0	1: Timer A interrupt pending

Table 13.8 – TIMER_INT – Timers Interrupt Register

13.2.7 TIMER_SELECT – Timers A..D Select Register (address offset: 0x06)

Bit	Name	Type	Default Value	Description
3:2	timer_read_sel	RW	2'h0	Selects one of timers or pre-scalers A/B/C/D to read. 0: timer/pre-scaler A 1: timer/pre-scaler B 2: timer/pre-scaler C 3: timer/pre-scaler D
1:0	timer_write_sel	RW	2'h0	Selects one of timers or pre-scalers A/B/C/D to write. 0: timer/pre-scaler A 1: timer/pre-scaler B 2: timer/pre-scaler C 3: timer/pre-scaler D

Table 13.9 – TIMER_SELECT – Timers A..D Select Register

Copyright © Bridgetek Pte Ltd

13.2.8 TIMER_WDG – Watchdog Start Value (address offset: 0x07)

Bit	Name	Type	Default Value	Description
4:0	timer_wdg_write	RW	5'h00	Sets the watchdog duration in the number of system clock cycles. The number of clock cycles is equal to $2^{timer_wdg_write}$. timer_wdg_write: Number of cycles 5'h00: 1 5'h01: 2 …: … 5'h1E: 1,073,741,824 5'h1F: 2,147,483,648

Table 13.10 – TIMER_WDG – Watchdog Start Value

13.2.9 TIMER_WRITE_LSB – Timer A..D Start Value 7:0 (address offset: 0x08)

Note: TIMER_SELECT must be selected first for proper operation.

Bit	Name	Type	Default Value	Description
7:0	timer_write_7_0	RW	8'h00	Writes the LSB of the timer start value

Table 13.11 – TIMER_WRITE_LSB – Timer A..D Start Value 7:0

13.2.10 TIMER_WRITE_MSB – Timer A..D Start Value 15:8 (address offset: 0x09)

Bit	Name	Type	Default Value	Description
7:0	timer_write_15_8	RW	8'h00	Writes the MSB of the timer start value

Table 13.12 – TIMER_WRITE_MSB – Timer A..D Start Value 15:8

13.2.11 TIMER_PRESC_LSB – Pre-scaler Start Value 7:0 (address offset: 0x0A)

Bit	Name	Type	Default Value	Description
7:0	timer_presc_7_0	RW	8'h00	Writes the LSB of the timer pre-scaler start value

Table 13.13 – TIMER_PRESC_LSB – Pre-scaler Start Value 7:0

13.2.12 TIMER_PRESC_MSB – Pre-scaler Start Value 15:8 (address offset: 0x0B)

Bit	Name	Type	Default Value	Description
7:0	timer_presc_15_8	RW	8'h00	Writes MSB of the timer pre-scaler start value

Table 13.14 – TIMER_PRESC_MSB – Pre-scaler Start Value 15:8

13.2.13 TIMER_READ_LSB – Timer A..D Current Value 7:0 (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	timer_read_7_0	RO	8'h00	Reads the MSB of the timer start value

Table 13.15 – TIMER_READ_LSB – Timer A..D Current Value 7:0

Copyright © Bridgetek Pte Ltd

13.2.14 TIMER_READ_MSB – Timer A..D Current Value 15:8 (address offset: 0x0D)

Bit	Name	Type	Default Value	Description
7:0	timer_read_15_8	RO	8'h00	Reads the MSB of the timer start value

Table 13.16 – TIMER_READ_MSB – Timer A..D Current Value 15:8

13.2.15 TIMER_RESERVED – Reserved Register (address offset: 0x0E)

Bit	Name	Type	Default Value	Description
7:0	Reserved	-	-	-

Table 13.17 – TIMER_RESERVED – Reserved Register

13.2.16 WDT_STATUS – Watchdog Status Register (address offset: 0x0F)

Bit	Name	Type	Default Value	Description
7:1	Reserved	RO	7'h00	-
0	WDG_STATUS	RAC	1'h0	This bit is set if there was a second watchdog roll-over that causes a POR-like system reset. It is automatically cleared after being read.

Table 13.18 – WDT_STATUS – Watchdog Status Register

Copyright © Bridgetek Pte Ltd

14 I²S

The I²S interface supports both master and slave modes. The formats supported are I²S, left-justified and right-justified.

In master mode, 2 clock sources are to be provided externally. One is 24.576 MHz and the other is 22.5792 MHz. LRCLK, BCLK and MCLK will be generated by the module based on the sampling rate and bit length.

The table below shows the oversampling rate supported (i.e. the MCLK frequency supported). The MCLK divider factor must be set accordingly. The reference clock is either the 24.576 MHz or 22.5792 MHz incoming clock.

Sampling frequency (Hz)	Oversampling Rate Supported	Reference Clock	MCLK Divider Setting
11,025	32fs	22.5792 MHz	64
	64fs		32
	128fs		16
	256fs		8
22,050	32fs	22.5792 MHz	32
	64fs		16
	128fs		8
	256fs		4
44,100	32fs	22.5792 MHz	16
	64fs		8
	128fs		4
	256fs		2
16,000	32fs	24.576 MHz	48
	64fs		24
	128fs		12
	256fs		6
	384fs		4
	512fs		3
32,000	32fs	24.576 MHz	24
	64fs		12
	128fs		6
	256fs		3
	384fs		2
48,000	32fs	24.576 MHz	16
	64fs		8
	128fs		4
	256fs		2
	512fs		1
96,000	32fs	24.576 MHz	8
	64fs		4

Copyright © Bridgetek Pte Ltd

Sampling frequency (Hz)	Oversampling Rate Supported	Reference Clock	MCLK Divider Setting
	128fs		2
	256fs		1
192,000	32fs	24.576 MHz	4
	64fs		2
	128fs		1
384,000	32fs	24.576 MHz	2
	64fs		1

Table 14.1 – Oversampling rates supported by FT90x Revision C I²S

In this mode (i.e., master mode), the number of BCLK cycles per channel per sampling cycle can only be either 16 or 32.

The table shows the supported bit length with the number of BCLK per sampling cycle in the Master mode. The BCLK divider factor must be set accordingly. The reference clock is either the 24.576 MHz or 22.5792 MHz incoming clock.

Sampling Frequency	Bit length per Channel	# of BCLK cycles per Channel per Sampling Cycle	BCLK Divider Setting
11,025	16	16	64
	16, 20, 24, 32	32	32
22,050	16	16	32
	16, 20, 24, 32	32	16
44,100	16	16	16
	16, 20, 24, 32	32	8
16,000	16	16	48
	16, 20, 24, 32	32	24
32,000	16	16	24
	16, 20, 24, 32	32	12
48,000	16	16	16
	16, 20, 24, 32	32	8
96,000	16	16	8
	16, 20, 24, 32	32	4
192,000	16	16	4
	16, 20, 24, 32	32	2
384,000	16	16	2
	16, 20, 24, 32	32	1

Table 14.2 – FT90x Revision C I²S settings

In slave mode, LRCLK and BCLK are to be input to the FT90X. MCLK is still generated if either 24.576 MHz or 22.5792 MHz input is available. If a MCLK output is not necessary in slave mode, the pins can be configured as GPIOs.

In the Slave mode, the incoming/outgoing data can be 16, 20, 24 or 32 bits per channel. And the incoming number of BCLK cycles per channel per sampling period can be 16, 20, 24 or at most 32.

Copyright © Bridgetek Pte Ltd

The data length can be X-bit per channel whereas the number of BCLK cycles per channel per sampling period can be any setting mentioned that is larger or equal to X-bit. For example, 16-bit data length per channel can be supported with any of 16, 20, 24 or 32 BCLK cycles per channel per sampling period setting.

Important: To support 384 kHz sampling rate in slave mode, the BCLK divider has to be set to 1 (i.e., no division) in the Configuration Register 2 even if BCLK is provided by the I2S master.

14.1 Register Summary

Listed below are the registers with their offset from the base address (0x10350). All registers can only be accessed via Word (16-bit) mode.

Offset	Register	Default value	References
0x00	I2SCR – Configuration Register 1	0x00	Section 14.2.1
0x02	I2SCR2 – Configuration Register 2	0x00	Section 14.2.2
0x04	I2SIRQEN – Interrupt Enable Register	0x00	Section 14.2.3
0x06	I2SIRQPEND – Interrupt Pending Register	0x00	Section 14.2.4
0x08	I2SRWDATA – Transmit / Receive Data Register	0x00	Section 14.2.5
0x0C	I2SRXCOUNT – RX Count Register	0x00	Section 14.2.6
0x0E	I2STXCOUNT – TX Count Register	0x00	Section 14.2.7

Table 14.3 – Overview of I²S Registers

14.2 Register Details

14.2.1 I2SCR – Configuration Register 1 (address offset: 0x00)

Bit	Name	Type	Default Value	Description
15	Soft Reset	RW	1'b0	1: To turn on the soft reset. 0: To turn off the soft reset.
14:12	Padding	RW	3'h0	0: No padding required (i.e., bit length = number of BCLK cycles) 1: 4 extra 0 bits are added. 2: 8 extra 0 bits are added. 3: 12 extra 0 bits are added. 4: 16 extra 0 bits are added. Others: Reserved
11:10	Format	RW	2'h0	0: I²S Format. 1: Left-justified Format. 2: Right-justified Format. Others: Reserved
9:8	Bit Length	RW	2'h0	0: 16-bit Format. 1: 20-bit Format. 2: 24-bit Format. 3: 32-bit Format.
7	IsMaster64	RW	1'b0	This is valid for I²S Master mode only. 1: 32 BCLK cycles per channel. 0: 16 BCLK cycles per channel.

Copyright © Bridgetek Pte Ltd

6	MasterMode	RW	1'b0	1: Set I²S to the Master Mode.
5	IsMaster22	RW	1'b0	This is valid for I²S Master mode only. 1: Use 22.5792MHz input as the reference clock. 0: Use 24.576MHz input as the reference clock.
4	BCLK Polarity	RW	1'b0	1: Invert the polarity of BCLK.
3	LRCLK Out Polarity	RW	1'b0	1: Invert the polarity of LRCLK for reception.
2	LRCLK In Polarity	RW	1'b0	1: Inver the polarity of LRCLK for transmission.
1	RX Enable	RW	1'b0	1: Enable the receive channel.
0	TX Enable	RW	1'b0	1: Enable the transmit channel.

Table 14.4 – I2SCR – Configuration Register 1

14.2.2 I2SCR2 – Configuration Register 2 (address offset: 0x02)

Bit	Name	Type	Default Value	Description
15:12	Reserved	-	-	-
11:8	MCLK Divider	RW	4'h0	This is valid for I²S master mode only. 0: No division (MCLK = reference clock). 1: Divide by 2. 2: Divide by 3. 3: Divide by 4. 4: Divide by 6. 5: Divide by 8. 6: Divide by 12. 7: Divide by 16. 8: Divide by 24. 9: Divide by 32. A: Divide by 48. B: Divide by 64. Others: Reserved
7:4	Reserved	-	-	-
3:0	BCLK Divider	RW	4'h0	This is valid for I²S master mode only, *except when 384 kHz is desired in slave mode*. 0: No division (BCLK = reference clock). 1: Divide by 2. 2: Divide by 3. 3: Divide by 4. 4: Divide by 6. 5: Divide by 8. 6: Divide by 12. 7: Divide by 16. 8: Divide by 24. 9: Divide by 32. A: Divide by 48. B: Divide by 64. Others: Reserved

Table 14.5 – I2SCR2 – Configuration Register 2

Copyright © Bridgetek Pte Ltd

14.2.3 I2SIRQEN – Interrupt Enable Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
15:13	Reserved	-	-	-
12	RX FIFO Ov	RW	1'b0	1: Enable Receive FIFO Overflow Interrupt.
11	RX FIFO Full	RW	1'b0	1: Enable Receive FIFO Full Interrupt.
10	RX FIFO Half	RW	1'b0	1: Enable Receive FIFO Half Full Interrupt.
9	RX FIFO Empty	RW	1'b0	1: Enable Receive FIFO Empty Interrupt.
8	RX FIFO Under	RW	1'b0	1: Enable Receive FIFO Underflow Interrupt.
7:5	Reserved	-	-	-
4	TX FIFO Ov	RW	1'b0	1: Enable Transmit FIFO Overflow Interrupt.
3	TX FIFO Full	RW	1'b0	1: Enable Transmit FIFO Full Interrupt.
2	TX FIFO Half	RW	1'b0	1: Enable Transmit FIFO Half Full Interrupt.
1	TX FIFO Empty	RW	1'b0	1: Enable Transmit FIFO Empty Interrupt.
0	TX FIFO Under	RW	1'b0	1: Enable Transmit FIFO Underflow Interrupt.

Table 14.6 – I2SIRQEN – Interrupt Enable Register

14.2.4 I2SIRQPEND – Interrupt Pending Register (address offset: 0x06)

Bit	Name	Type	Default Value	Description
15:13	Reserved	-	-	-
12	RX FIFO Ov	RW1C	1'b0	1: Receive FIFO Overflow Interrupt Pending. Write 1 to clear.
11	RX FIFO Full	RW1C	1'b0	1: Receive FIFO Full Interrupt Pending. Write 1 to clear.
10	RX FIFO Half	RW1C	1'b0	1: Receive FIFO Half Full Interrupt Pending. Write 1 to clear.
9	RX FIFO Empty	RW1C	1'b0	1: Receive FIFO Empty Interrupt Pending. Write 1 to clear.
8	RX FIFO Under	RW1C	1'b0	1: Receive FIFO Underflow Interrupt Pending. Write 1 to clear.
7:5	Reserved	-	-	-
4	TX FIFO Ov	RW1C	1'b0	1: Transmit FIFO Overflow Interrupt Pending. Write 1 to clear.
3	TX FIFO Full	RW1C	1'b0	1: Transmit FIFO Full Interrupt Pending. Write 1 to clear.
2	TX FIFO Half	RW1C	1'b0	1: Transmit FIFO Half Full Interrupt Pending. Write 1 to clear.
1	TX FIFO Empty	RW1C	1'b0	1: Transmit FIFO Empty Interrupt Pending. Write 1 to clear.
0	TX FIFO Under	RW1C	1'b0	1: Transmit FIFO Underflow Interrupt Pending. Write 1 to clear.

Table 14.7 – I2SIRQPEND – Interrupt Pending Register

14.2.5 I2SRWDATA – Transmit / Receive Data Register (address offset: 0x08)

Bit	Name	Type	Default Value	Description
15:0	TX FIFO Data	WO	-	Data window for writing data to the transmit FIFO.
15:0	RX FIFO Data	RO	16'h0000	Data window for reading data from the receive FIFO.

Table 14.8 – I2SRWDATA – Transmit / Receive Data Register

Copyright © Bridgetek Pte Ltd

14.2.6 I2SRXCOUNT – RX Count Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
15:0	RX FIFO Data Count	RO	16'h0000	Indicates the number of 16-bit data entries in the receiver FIFO.

Table 14.9 – I2SRXCOUNT – RX Count Register

14.2.7 I2STXCOUNT – TX Count Register (address offset: 0x0E)

Bit	Name	Type	Default Value	Description
15:0	TX FIFO Data Count	RO	16'h0000	Indicates the number of 16-bit data entries in the transmitter FIFO.

Table 14.10 – I2STXCOUNT – TX Count Register

Copyright © Bridgetek Pte Ltd

15 SPI Master

There is a SPI Master module in the FT90x device.

Listed below are the key features of this SPI master:

- Full duplex synchronous serial data transfer
- Single, Dual and Quad SPI transfer
- Master operation
- Multi-master system supported
- Two modes of operations: SPI mode and FIFO mode
- FIFO size of 16, 64 or 256# bytes
- Support up to 8 SPI slaves
- System error detection
- Interrupt generation
- Clock rates supported are 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256, 1/511 of the system clock
- Four transfer formats supported

Note:
 # denotes in extended FIFO mode. See section 15.2.

15.1 Register Summary

Listed below are the registers with their offset from the base address (0x102A0). All registers can only be accessed via Double-Word (32-bit) mode. However, the least significant byte of the FIFO can be accessed via Byte (8-bit) mode. This facilitates fast byte-oriented operations. For more information about the FIFO access modes, refer to section 15.2.

Address Offset	Register	Default Value	References
0x00	SPIM_CNTL – Control Register	0x04	Section 15.5.1
0x04	SPIM_STATUS – Status Register	0x00	Section 15.5.2
0x08	SPIM_DATA – Receiver and Transmitter Data Registers	0x20	Section 15.5.3
0x0C	SPIM_SLV_SEL_CNTL – Slave Select Control Register	0xFF	Section 15.5.4
0x10	SPIM_FIFO_CNTL – FIFO Control Register	0x00	Section 15.5.5
0x14	SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register	0x00	Section 15.5.6
0x18	SPIM_ALT_DATA – Alternative SPI Master Data Register	0x00	Section 15.5.7
0x1C	SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register	0x00	Section 15.5.8

Table 15.1 – Overview of SPI Master Registers

Copyright © Bridgetek Pte Ltd

Listed below are the registers with their offset from the base address (0x10940). All registers can only be accessed via Double-Word (32-bit) mode.

Address Offset	Register	Default Value	References
0x00	SPIM_CNTL_2 – Control Register 2	0x01	Section 15.5.9
0x10	SPIM_STATUS_2 – Status Register 2	0x05	Section 15.5.10
0x14	SPIM_FIFO_CNTL_2 – FIFO Control Register 2	0x01	Section 15.5.11
0x18	SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register	0x00	Section 15.5.12
0x1C	SPIM_BAUD – SPI Master Baud Register	0x04	Section 15.5.13

Table 15.2 – Overview of SPI Master Extended Registers

15.2 Register/FIFO Access Modes and FIFO Size

Each SPI Master register resides in a 32-bit address location. The user has to access these locations in DW (32-bit) mode, even though the actual registers only take up the first 8 bits. The remaining bits are not accessible. The only exception to this rule is the FIFO in extended mode (enabled by setting SPIM_TNSFR_FRMT_CNTL[7]), for which the entire 32 bits are available. The following figures demonstrate the location of the registers, 8-bit FIFO and 32-bit FIFO in a 32-bit address location.

[31:24]	[23:16]	[15:8]	[7:0]
XX	XX	XX	Registers

Figure 15.1 – SPI Master Register Access

[31:24]	[23:16]	[15:8]	[7:0]
XX	XX	XX	FIFO in 8-bit mode

Figure 15.2 – SPI Master 8-bit FIFO Access

[31:24]	[23:16]	[15:8]	[7:0]
FIFO in 32-bit mode	FIFO in 32-bit mode	FIFO in 32-bit mode	FIFO in 32-bit mode

Figure 15.3 – SPI Master 32-bit FIFO Access

The FIFO size varies according the FIFO access mode. Specifically, it depends on:

- FIFO depth (16 or 64 levels), selected by SPIM_FIFO_CNTL[5].
- FIFO access mode (normal or extended mode), selected by SPIM_TNSFR_FRMT_CNTL[7].

The following table shows the maximum FIFO size, depending on the settings.

FIFO Depth	FIFO Access Mode	FIFO Size (bytes)
16	8-bit	16
16	32-bit	64
64	8-bit	64
64	32-bit	256

Table 15.3 – SPI Master FIFO Size

Copyright © Bridgetek Pte Ltd

Note: When the FIFO access mode is set to 32-bit, all accesses to the SPIM_DATA register are treated as 32-bit. If an 8-bit access is performed in this mode, undesirable result may occur. The software should switch the FIFO access mode to 8-bit if an 8-bit access is required.

15.3 Block Diagram

Below is the block diagram which demonstrates the overall operation of the SPI Master.

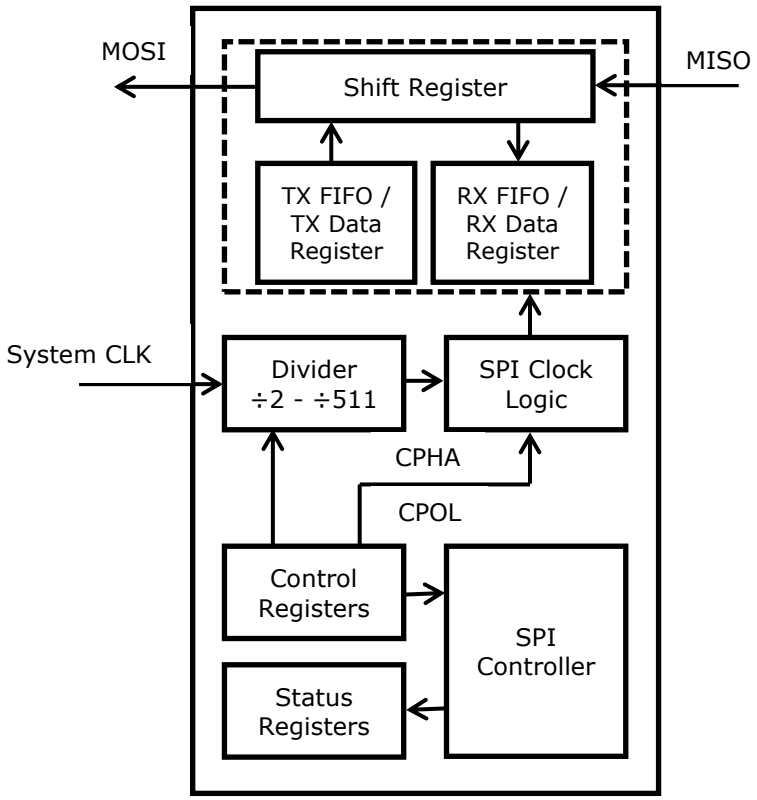


Figure 15.4 – SPI Master Block Diagram

Shift Register: shifts the data in and out from the SPI controller at the appropriate edge of the SPI clock. In single SPI transfer mode, an 8-bit character is shifted out on the MOSI pin while another 8-bit character is simultaneously shifted in on the MISO pin. Another way to view this transfer is that an 8-bit shift register in the SPI master and another 8-bit shift register in the SPI slave are connected as a circular 16-bit shift register. When a transfer occurs, this circular shift register is shifted eight-bit positions; thus, the characters in the master and slave are effectively exchanged. In dual or quad SPI transfer, the data is shifted in or out on 2 or 4 different SPI lines at once respectively.

Receiver (RX) FIFO: can be 16 or 64 levels deep. It receives data until the number of data entries in the FIFO equals the selected interrupt trigger level. At that time, if interrupt is enabled, the SPI controller will issue an interrupt to the CPU. The RX FIFO will continue to store data until it is full, and will not accept any more data afterwards. Any data entering the shift register when the RX FIFO is full will set the Overrun Error flag.

Transmitter (TX) FIFO: the data is transmitted on the MOSI pin as soon as the CPU loads an 8-bit character into the TX FIFO. The SPI controller prevents further loading to the TX FIFO if it is currently full. Loading to the TX FIFO is enabled again as soon as the next character is transferred to the shift register.

Divider: divides the system clock signal by 2 up to 511. The resulting signal is used to generate the SPI clock.

Copyright © Bridgetek Pte Ltd

SPI Clock Logic: controls the phase and polarity of the SPI clock. On the correct edge, a signal is generated to the shift register to shift the data in or out.

SPI Controller: controls the SPI operations, manages the interrupt requests, performs error detection etc.

15.4 SPI Master Interrupts

The SPI Master can generate an interrupt to the CPU when one of the following conditions occurs:

- The receiver FIFO trigger level is reached.[1 2]
- The transmitter FIFO trigger level is reached.[1 3]
- Receiver FIFO timeout.[1]
- Transfer error.[1]
- Transmitter FIFO empty.[4]

Notes:
[1] Masked by the SP_IE bit in the SPIM_CNTL register.
[2] Masked by the RX_TRIGGER_EN bit in the SPIM_FIFO_CNTL_2 register.
[3] Masked by the TX_TRIGGER_EN bit in the SPIM_FIFO_CNTL_2 register.
[4] Masked by the TX_IEN Bit in the SPIM_TNSFR_FRMT_CNTL register.

15.4.1 Receiver FIFO Trigger Level Interrupt

In the FIFO mode, an interrupt is generated whenever the Receiver FIFO level reaches the trigger level (RX_TRIGGER[1:0]) specified in the FIFO Control Register (SPIM_FIFO_CNTL).

The Receiver FIFO Trigger Level interrupt is cleared by reading the Receiver FIFO until the FIFO level drops below the trigger level.

15.4.2 Transmitter FIFO Trigger Level Interrupt

In the FIFO mode, an interrupt is generated whenever there are more vacant entries in the Transmitter FIFO than the trigger level (TX_TRIGGER[1:0]) specified in the FIFO Control Register 2 (SPIM_FIFO_CNTL_2). The user can check the TX_SPI_FLAG bit in the Status Register 2 to determine the interrupt source.

This interrupt is cleared by filling data into the Transmitter FIFO until the FIFO level is more than the trigger level.

15.4.3 Receiver FIFO Timeout Interrupt

A receiver FIFO timeout interrupt will occur if all of the following conditions exist:

- There is at least one character in the FIFO.
- The most recent character was received more than 3 continuous character times ago. A character time is defined as the time it takes to send/receive one 8-bit character. For example, when the SPI clock is 25 MHz, a character time is 8 x 0.04 us = 0.32 us.
- The most recent CPU read of the FIFO was longer than 3 continuous character times ago.

When a timeout interrupt has occurred, it is cleared and the timeout timer is reset when the CPU reads at least one character from the Receiver FIFO.

When a timeout interrupt has not occurred, the timeout timer is reset after a new character is received or after the CPU reads the Receiver FIFO.

15.4.4 Transfer Error Interrupt

The transfer error (Mode-Fault Error) interrupt occurs when the SPI Master is selected by another SPI master device as if it were a slave. When that happens:

- The MSTR bit is forced to zero to disable the SPI Master function.
- The SP_IE bit is forced to zero to disable the SPI engine.

Copyright © Bridgetek Pte Ltd

- The MOD_FAULT status flag is set and an interrupt request is generated to the CPU.

The interrupt is cleared by reading the SPIM_STATUS register (when MOD_FAULT is set) followed by writing to the SPIM_CNTL register after the error condition is removed.

15.4.5 Transmitter FIFO Empty Interrupt

This interrupt occurs when the SPI Master is idle and the Transmitter FIFO is empty. It can be used in FIFO Mode only.

15.4.6 FIFO Polling Mode

When the FIFO mode is enabled (FIFO_EN = 1), clearing SP_IE puts the SPI Master in the FIFO polling mode. In this mode, the user may check the RX and TX FIFO status via the SPIM_STATUS and SPIM_STATUS_2 registers.

There is no trigger level reached or timeout condition indicated in this mode. However, both TX and RX FIFOs are still fully functional.

15.5 Register Details

15.5.1 SPIM_CNTL – Control Register (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7	SP_IE	RW	1'b0	1: Enables interrupts for: • Reaching RX FIFO trigger level. • Reaching TX FIFO trigger level. • Receiver FIFO timeout. • Transfer error. 0: Disables SPI Master interrupts, polling mode is used.
6	SP_E	RW	1'b0	1: Turns on the SPI engine. Both this bit and the MSTR bit should be set to enable the SPI Master. 0: Turns off the SPI engine.
5	SPR2	RW	1'b0	See the table at SPR[1:0] (bits [1:0] of this register).
4	MSTR	RW	1'b0	1: Enables the SPI Master function.
3	CPOL	RW	1'b0	Clock polarity 0: High level; SCK idles Low 1: Low level; SCK idles High
2	CPHA	RW	1'b1	Clock phase 0: Shift Data Out on Falling edge; capture Data In on Rising edge 1: Shift Data Out on Rising edge; capture Data In on Falling edge

Copyright © Bridgetek Pte Ltd

<table>
<tr><td>1:0</td><td>SPR[1:0]</td><td>RW</td><td>1’b0</td><td>Together with SPR2, FAST_SPI and CHG_SPR they define the SPI clock rate.
<table>
<tr><th>FAST_SPI</th><th>CHG_SPR</th><th>SPR2</th><th>SPR1</th><th>SPR0</th><th>System Clock (100 MHz) divided by</th></tr>
<tr><td>1</td><td>0</td><td>X</td><td>X</td><td>X</td><td>2</td></tr>
<tr><td>0</td><td>X</td><td>0</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td>0</td><td>X</td><td>0</td><td>0</td><td>1</td><td>8</td></tr>
<tr><td>0</td><td>X</td><td>0</td><td>1</td><td>0</td><td>16</td></tr>
<tr><td>0</td><td>X</td><td>0</td><td>1</td><td>1</td><td>32</td></tr>
<tr><td>0</td><td>X</td><td>1</td><td>0</td><td>0</td><td>64</td></tr>
<tr><td>0</td><td>X</td><td>1</td><td>0</td><td>1</td><td>128</td></tr>
<tr><td>0</td><td>X</td><td>1</td><td>1</td><td>0</td><td>256</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>511</td></tr>
<tr><td>X</td><td>1</td><td>1</td><td>1</td><td>1</td><td>2</td></tr>
</table></td></tr>
</table>

Table 15.4 – SPIM_CNTL – Control Register

15.5.2 SPIM_STATUS – Status Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7	SPI_FLAG	RO	1’b0	Receiver data available. The flag is automatically set at the end of an SPI transfer, when a character is transferred from the shift register to the Receiver Data Register/Receiver FIFO. In SPI mode, it is cleared by reading the SPIM_STATUS register followed by accessing the SPIM_DATA register. In FIFO mode, it is cleared when the Receiver FIFO is empty.
6	WR_COL	RO	1’b0	Write collision error status flag. In SPI mode, the flag is automatically set if the SPIM_DATA register is written when the previous data has not been sent out. In FIFO Mode, the mode is set if the SPIM_DATA register is written when the Transmitter FIFO is full. It is cleared when there is a read or write to the SPIM_DATA register again, regardless of the status of the previous data/transmitter FIFO. The software should make sure that the collision has been resolved before clearing this bit.
5	SPI_BIS	RO	1’b0	Indicates the end of transmission from the SPIM_ALT_DATA register. This flag can generate an interrupt if SPIM_TNSFR_FRMT_CNTL[6] is set. It is cleared when there is a read or write to the SPIM_ALT_DATA register.
4	MOD_FAULT	RO	1’b0	SPI mode-fault error status flag. This flag is set if a slave select pin is set to low by another SPI master device. It is cleared by reading the SPIM_STATUS register (when this bit is set) followed by writing to the SPIM_CNTL register after the error condition is removed.
3	THRE	RO	1’b0	Transmitter Holding Register Empty 1: The SPI Master is in the idle state with the Transmitter FIFO empty (FIFO mode) or when a character is transferred from the Transmitter Data Register to the shift register (SPI mode).

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
				0: Transmission is in progress. There is data in the Transmitter Data Register (SPI mode) or at least one character by in the Transmitter FIFO (FIFO mode).
2:1	Reserved	-	-	-
0	SSC_EN	RW	1'b0	Slave Select (SS) Control Enable 1: auto SS assertion enabled. During the transmission, the SS pins are automatically loaded with the content of the SPIM_SLV_SEL_CNTL register before each byte transfer, and de-asserted when the transfer is completed. 0: auto SS assertion disabled. The SS pins always reflects the content of the SPIM_SLV_SEL_CNTL register.

Table 15.5 – SPIM_STATUS – Status Register

15.5.3 SPIM_DATA – Receiver and Transmitter Data Registers (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:0	SPDR_RX	RO	8'h00	Receiver data register. Contains data from the last Receive operation.
7:0	SPDR_TX	WO	8'h00	Transmitter data register. Contains data for the next Transmit operation

Table 15.6 – SPIM_DATA – Receiver and Transmitter Data Registers

15.5.4 SPIM_SLV_SEL_CNTL – Slave Select Control Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	SS_CR	RW	8'hFF	SS7-SS0 pin control. Each bit controls one SS pin. 0: asserts the corresponding SS pin 1: de-asserts the corresponding SS pin (the pin is forced to logic 1)

Table 15.7 – SPIM_SLV_SEL_CNTL – Slave Select Control Register

15.5.5 SPIM_FIFO_CNTL – FIFO Control Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7:6	RX_TRIGGER[1:0]	RW	2'b00	Receiver FIFO Trigger Level, depending on the FIFO depth specified by bit 5 of this register.

RX_TRIGGER[1:0]	Trigger level in 16-level FIFO Mode	Trigger level in 64-level FIFO mode
00	1 level	1 level
01	4 levels	16 levels
10	8 levels	32 levels
11	14 levels	56 levels

5	FIFO_DEPTH	RW	1'b0	FIFO depth selection. 1: FIFO size is 64 bytes 0: FIFO size is 16 bytes

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
4	TIMEOUT	RW	1'b0	Receiver FIFO timeout enable 1: Enables timeout interrupt 0: Disables timeout interrupt
3	Reserved	-	-	-
2	TX_RST	RWAC	1'b0	Transmitter FIFO reset. Writing a 1 to this bit clears all bytes in the Transmitter FIFO and resets its counter to 0. The shift register is not affected. This bit will clear itself
1	RX_RST	RWAC	1'b0	Receiver FIFO reset. Writing a 1 to this bit clears all bytes in the Receiver FIFO and resets its counter to 0. The shift register is not affected. This bit will clear itself
0	FIFO_EN	RW	1'b0	Receiver and Transmitter FIFO enable. 0: Disables FIFO 1: Enables FIFO

Table 15.8 – SPIM_FIFO_CNTL – FIFO Control Register

15.5.6 SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7	FIFO_EXT	RW	1'b0	FIFO extension 1: Allows 32-bit data transfer to/from FIFO. The FIFO size is also extended to 256 bytes.
6	BISINT_EN	RW	1'b0	1: Enables interrupt generation after the transfer is complete from the SPIM_ALT_DATA register.
5	MULTI_REC	RW	1'b0	1: Allows continuous reception of data without the necessity of loading the TX FIFO
4	FAST_SPI	RW	1'b0	1: Enables the FAST mode, in which the SPI clock speed is equal to ½ of the system clock. This bit is ignored when CHG_SPR is set (see the table at SPR[1:0]).
3	TX_IEN	RW	1'b0	1: Enables Transmitter FIFO empty interrupt
2	DIR	RW	1'b0	Controls the data transfer direction (read/write) in dual and quad SPI modes. 1: Performs multi-channel SPI Read 0: Performs multi-channel SPI Write **Note:** Requires bit 1 (QUAD_SPI) or 0 (DUAL_SPI) of this register to be set.
1	QUAD_SPI	RW	1'b0	1: Enables quad SPI transfer
0	DUAL_SPI	RW	1'b0	1: Enables dual SPI transfer

Table 15.9 – SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register

15.5.7 SPIM_ALT_DATA – Alternative SPI Master Data Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Alternative SPI Mode Data register. Data transmitted through this register are done as a single channel SPI only, regardless of QUAD/DUAL mode setting.

Table 15.10 – SPIM_ALT_DATA – Alternative SPI Master Data Register

Copyright © Bridgetek Pte Ltd

15.5.8 SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register (address offset: 0x1C)

Bit	Name	Type	Default Value	Description
7	Reserved	RO	1'b0	-
6:0	RX_FIFO_COUNT	RO	7'h00	The number of data entries available in the RX FIFO. The data can be 8-bit or 32-bit, depending on the FIFO extension setting. Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and SPI_FLAG bit is 1, it means there are 64 data entries in the FIFO.

Table 15.11 – SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register

Note: The following registers are located at a different base address (0x10940).

15.5.9 SPIM_CNTL_2 – Control Register 2 (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7	BAUD_REG _EN	RW	1'b0	1: The SPIM_BAUD register is used to generate the SPI clock. 0: The SPIM_BAUD register is disabled. The SPI clock rate is controlled by SPR[2:0], FAST_SPI and CHG_SPR.
6:4	Reserved	-	-	-
3	ALT_DATA_ LSBF	RW	1'b0	1 : The data in the SPIM_ALT_DATA register (normally used for SPI commands) is sent with the least-significant bit first. This bit affects only the TX operation. 0: The data in the SPIM_ALT_DATA register is sent on the bus with the most-significant bit first.
2	LSBF	RW	1'b0	1: The data in SPIM_DATA is sent/received with the least-significant bit first. This bit affects both TX and RX operations. 0: The data in SPIM_DATA is sent/received with the most-significant bit first.
1	CHG_SPR	RW	1'b0	Controls the SPI clock rate together with SPR[2:0] and FAST_SPI. Refer to the SPIM_CNTL register.
0	RX_EMPTY	RW	1'b1	Receiver data empty 1: the receiver data register (SPI mode) or the receiver FIFO (FIFO mode) is empty.

Table 15.12 – SPIM_CNTL_2 – Control Register 2

15.5.10 SPIM_STATUS_2 – Status Register 2 (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7	Reserved	RO	1'b0	-
6	TEMT	RO	1'b0	1: In SPI mode, it indicates that the transmitter data register and the shift register are both empty. In FIFO mode, it indicates that the transmitter FIFO and the shift register are both empty. 0: indicates that either the transmitter data register/ transmitter FIFO or the shift register contains a character data.
5	TX_SPI_FLAG	RO	1'b0	1 : indicates that the vacant entries in the TX FIFO are more than the TX FIFO trigger level.

Copyright © Bridgetek Pte Ltd

4	RX_FULL_OVERRIDDEN	RO	1'b0	1: RX FIFO overflows and data has been overridden. This bit is cleared by reading the SPIM_DATA register.
3	RX_FIFO_FULL	RO	1'b0	1: RX FIFO is full. 0: There is at least an empty entry in the RX FIFO.
2	RX_EMPTY	RO	1'b0	1: the receiver data register (SPI mode) or the receiver FIFO (FIFO mode) is empty.
1	TX_FIFO_FULL	RO	1'b0	1: TX FIFO is full. 0: There is at least an empty entry in the TX FIFO.
0	TX_EMPTY	RO	1'b0	1: the transmitter data register (SPI mode) or the transmitter FIFO (FIFO mode) is empty and the bus transaction is completed.

Table 15.13 – SPIM_STATUS_2 – Status Register 2

15.5.11 SPIM_FIFO_CNTL_2 – FIFO Control Register 2 (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7:4	Reserved	RO	4'h0	-
3:2	TX_TRIGGER [1:0]	RW	2'b00	Transmitter FIFO Trigger Level, depending on the FIFO depth specified in the SPIM_FIFO_CNTL register. **TX_TRIGGER [1:0]** / **Trigger level in 16-level FIFO Mode** / **Trigger level in 64-level FIFO mode** 00 / 1 level / 1 level 01 / 4 levels / 16 levels 10 / 8 levels / 32 levels 11 / 14 levels / 56 levels
1	TX_TRIGGER_EN	RW	1'b0	1 : enables interrupt when the TX FIFO level is greater than the TX trigger level.
0	RX_TRIGGER_EN	RW	1'b1	1 : enables interrupt when the RX FIFO level is greater than the RX trigger level.

Table 15.14 – SPIM_FIFO_CNTL_2 – FIFO Control Register 2

15.5.12 SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
7	Reserved	RO	1'b0	-
6:0	TX_FIFO_COUNT	RO	7'h00	The number of data entries available in the TX FIFO. The data can be 8-bit or 32-bit, depending on the FIFO extension setting. Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and SPI_FLAG bit is 1, it means there are 64 data entries in the FIFO.

Table 15.15 – SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register

15.5.13 SPIM_BAUD – SPI Master Baud Register (address offset: 0x1C)

When the BAUD_REG_EN bit (SPIM_CNTL_2[7]) is set, the Baud register allows the user to specify an accurate SPI clock rate which ranges from 1/2 to 1/256 of the system clock rate with a resolution of 1 system clock cycle. The SPI clock period is calculated as the system clock period

Copyright © Bridgetek Pte Ltd

multiplied by the value stored in the SPIM_BAUD register. When the SPIM_BAUD register holds an even number, the SPI clock low time is equal to the SPI clock high time. When the SPIM_BAUD register holds an odd number, the default state of the SPI clock specified by SPIM_CNTL[3] (CPOL) is 1 system clock period longer than the other state.

Bit	Name	Type	Default Value	Description
7:0	BAUD_FACTOR	RW	8'h04	The number of data entries available in the TX FIFO. The data can be 8-bit or 32-bit, depending on the FIFO extension setting. Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and SPI_FLAG bit is 1, it means there are 64 data entries in the FIFO.

Table 15.16 – SPIM_BAUD – SPI Master Baud Register

Copyright © Bridgetek Pte Ltd

16 SPI Slaves

There are two independent SPI slaves in the FT90X device.

Listed below are the key features of the SPI slaves:

- Full duplex synchronous serial data transfer
- Two modes of operations: SPI mode and FIFO mode
- FIFO size of 16, 64 or 256# bytes
- System error detection
- Interrupt generation
- Four transfer formats supported

Note:
 # denotes in extended FIFO mode. See section 16.2.

16.1 Register Summary

Listed below are the registers with their offset from the base addresses (0x102C0 and 0x102E0 respectively). All registers can only be accessed via Double-Word (32-bit) mode. However, the least significant byte of the FIFO can be accessed via Byte (8-bit) mode. This facilitates fast byte-oriented operations.

Address Offset	Register	Default value	References
0x00	SPIS_CNTL – Control Register	0x04	Section 16.5.1
0x04	SPIS_STATUS – Status Register	0x00	Section 16.5.2
0x08	SPIS_DATA – Receiver and Transmitter Data Registers	0x00	Section 16.5.3
0x0C	SPIS_RESERVED – Reserved Register	0x00	Section 16.5.4
0x10	SPIS_FIFO_CNTL – FIFO Control Register	0x00	Section 16.5.5
0x14	SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register	0x00	Section 16.5.6
0x18	SPIS_ALT_DATA – Alternative SPI Slave Data Register	0x00	Section 16.5.7
0x1C	SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register	0x00	Section 16.5.8

Table 16.1 – Overview of SPI Slave Registers

Copyright © Bridgetek Pte Ltd

Listed below are the registers with their offset from the base address (0x10960 for SPI Slave 0 and 0x10980 for SPI Slave 1). All registers can only be accessed via Double-Word (32-bit) mode.

Address Offset	Register	Default Value	References
0x00	SPIS_CNTL_2 – Control Register 2	0x01	Section 16.5.9
0x10	SPIS_STATUS_2 – Status Register 2	0x05	Section 16.5.10
0x14	SPIS_FIFO_CNTL_2 – FIFO Control Register 2	0x01	Section 16.5.11
0x18	SPIS_TX_FIFO_COUNT – SPI Master TX FIFO Count Register	0x00	Section 16.5.12

Table 16.2 – Overview of SPI Slave Extended Registers

16.2 Register/FIFO Access Modes and FIFO Size

Each SPI Slave register resides in a 32-bit address location. The user has to access these locations in DW (32-bit) mode, even though the actual registers only take up the first 8 bits. The remaining bits are not accessible. The only exception to this rule is the FIFO in extended mode (enabled by setting SPIS_TNSFR_FRMT_CNTL[7]), for which the entire 32 bits are available. The following figures demonstrate the location of the registers, 8-bit FIFO and 32-bit FIFO in a 32-bit address location.

[31:24]	[23:16]	[15:8]	[7:0]
XX	XX	XX	Registers

Figure 16.1 – SPI Slave Register Access

[31:24]	[23:16]	[15:8]	[7:0]
XX	XX	XX	FIFO in 8-bit mode

Figure 16.2 – SPI Slave 8-bit FIFO Access

[31:24]	[23:16]	[15:8]	[7:0]
FIFO in 32-bit mode	FIFO in 32-bit mode	FIFO in 32-bit mode	FIFO in 32-bit mode

Figure 16.3 – SPI Slave 32-bit FIFO Access

The FIFO size varies according the FIFO access mode. Specifically, it depends on:

- FIFO depth (16 or 64 levels), selected by SPIS_FIFO_CNTL[5].
- FIFO access mode (normal or extended mode), selected by SPIS_TNSFR_FRMT_CNTL[7].

The following table shows the maximum FIFO size, depending on the settings.

FIFO Depth	FIFO Access Mode	FIFO Size (bytes)
16	8-bit	16
16	32-bit	64
64	8-bit	64
64	32-bit	256

Table 16.3 – SPI Slave FIFO Size

Copyright © Bridgetek Pte Ltd

Note: When the FIFO access mode is set to 32-bit, all accesses to the SPIS_DATA register are treated as 32-bit. If an 8-bit access is performed in this mode, undesirable result may occur. The software should switch the FIFO access mode to 8-bit if an 8-bit access is required.

16.3 Block Diagram

Below is the block diagram which demonstrates the overall operation of the SPI Slave.

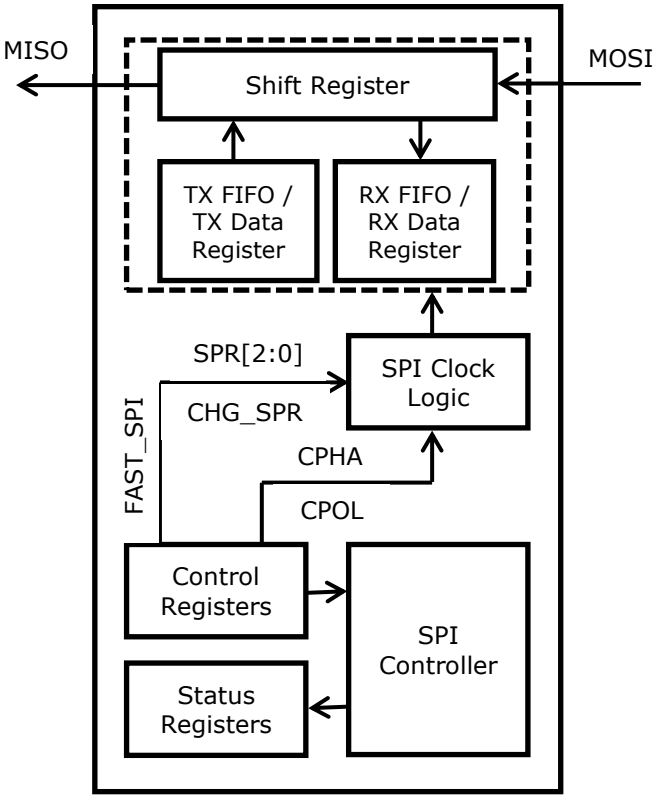


Figure 16.4 – SPI Slave Block Diagram

Shift Register: shifts the data in and out from the SPI controller at the appropriate edge of the SPI clock. In single SPI transfer mode, an 8-bit character is shifted out on the MISO pin while a another 8-bit character is simultaneously shifted in on the MOSI pin. Another way to view this transfer is that an 8-bit shift register in the SPI master and another 8-bit shift register in the SPI slave are connected as a circular 16-bit shift register. When a transfer occurs, this circular shift register is shifted eight-bit positions; thus, the characters in the master and slave are effectively exchanged.

Receiver (RX) FIFO: can be 16 or 64 levels deep. It receives data until the number of data entries in the FIFO equals the selected interrupt trigger level. At that time, if interrupt is enabled, the SPI controller will issue an interrupt to the CPU. The RX FIFO will continue to store data until it is full, and will not accept any more data afterwards. Any data entering the shift register when the RX FIFO is full will set the Overrun Error flag.

Transmitter (TX) FIFO: the data in the TX FIFO is transmitted on the MISO pin when the SPI Slave receives the SPI clock from a SPI master. The SPI controller prevents further loading to the TX FIFO if it is currently full. Loading to the TX FIFO is enabled again as soon as the next character is transferred to the shift register.

SPI Clock Logic: controls the phase and polarity of the SPI clock. On the correct edge, a signal is generated to the shift register to shift the data in or out.

Copyright © Bridgetek Pte Ltd

SPI Controller: controls the SPI operations, manages the interrupt requests, performs error detection etc.

16.4 SPI Slave Interrupts

The SPI Slave can generate an interrupt to the CPU when one of the following conditions occurs:

- The receiver FIFO trigger level is reached.[1] [2]
- The transmitter FIFO trigger level is reached.[1] [3]
- Receiver FIFO timeout.[1]
- Transmitter FIFO empty.[4]

Notes:
[1] Masked by the SP_IE bit in the SPIS_CNTL register.
[2] Masked by the RX_TRIGGER_EN bit in the SPIS_FIFO_CNTL_2 register.
[3] Masked by the TX_TRIGGER_EN bit in the SPIS_FIFO_CNTL_2 register.
[4] Masked by the TX_IEN Bit in the SPIS_TNSFR_FRMT_CNTL register.

16.4.1 Receiver FIFO Trigger Level Interrupt

In the FIFO mode, an interrupt is generated whenever the Receiver FIFO level reaches the trigger level (RX_TRIGGER[1:0]) specified in the FIFO Control Register (SPIS_FIFO_CNTL).

The Receiver FIFO Trigger Level interrupt is cleared by reading the Receiver FIFO until the FIFO level drops below the trigger level.

16.4.2 Transmitter FIFO Trigger Level Interrupt

In the FIFO mode, an interrupt is generated whenever there are more vacant entries in the Transmitter FIFO than the trigger level (TX_TRIGGER[1:0]) specified in the FIFO Control Register 2 (SPIS_FIFO_CNTL_2). The user can check the TX_SPI_FLAG bit the Status Register 2 to determine the interrupt source.

This interrupt is cleared by filling data into the Transmitter FIFO until the FIFO level is more than the trigger level.

16.4.3 Receiver FIFO Timeout Interrupt

A receiver FIFO timeout interrupt will occur if all of the following conditions exist:

- There is at least one character in the FIFO.
- The most recent character was received more than 32,767 system clock cycles ago.[#]
- The most recent CPU read of the FIFO was longer than 32,767 system clock cycles ago.[#]

Note:
[#]The system clock period of the FT90x series is 10ns, so the timeout period is 327.67 us.

When a timeout interrupt has occurred, it is cleared and the timeout timer is reset when the CPU reads at least one character from the Receiver FIFO.

When a timeout interrupt has not occurred, the timeout timer is reset after a new character is received or after the CPU reads the Receiver FIFO.

16.4.4 Transmitter FIFO Empty Interrupt

This interrupt occurs when the SPI Master is idle and the Transmitter FIFO is empty. It can be used in FIFO Mode only.

Copyright © Bridgetek Pte Ltd

16.4.5 FIFO Polling Mode

When the FIFO mode is enabled (FIFO_EN = 1), clearing SP_IE puts the SPI Slave in the FIFO polling mode. In this mode, the user may check the RX and TX FIFO status via the SPIS_STATUS and SPIS_STATUS_2 registers.

There is no trigger level reached or timeout condition indicated in this mode. However, both TX and RX FIFOs are still fully functional.

16.5 Register Details

16.5.1 SPIS_CNTL – Control Register (address offset: 0x00)

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Default Value</th><th>Description</th></tr>
<tr><td>7</td><td>SP_IE</td><td>RW</td><td>1’b0</td><td>1: To enable SPI Slave interrupt</td></tr>
<tr><td>6</td><td>SP_E</td><td>RW</td><td>1’b0</td><td>SPI System enable; 1 to enable.</td></tr>
<tr><td>5</td><td>SPR2</td><td>RW</td><td>1’b0</td><td>See the table at SPR[1:0] (bits [1:0] of this register).</td></tr>
<tr><td>4</td><td>Reserved</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>3</td><td>CPOL</td><td>RW</td><td>1’b0</td><td>Clock polarity select
0: High level; SCK idles Low
1: Low level; SCK idles high</td></tr>
<tr><td>2</td><td>CPHA</td><td>RW</td><td>1’b1</td><td>Clock phase
0: Shift Data Out on Falling edge; capture Data In on Rising edge
1: Shift Data Out on Rising edge; capture Data In on Falling edge</td></tr>
<tr><td>1:0</td><td>SPR[1:0]</td><td>RW</td><td>1’b0</td><td>Together with SPR2, CHG_SPR and FAST_SPI, these bits determine the maximum SPI clock speed that the SPI Slave can work with.
<table>
<tr><th>FAST_SPI</th><th>CHG_SPR</th><th>SPR2</th><th>SPR1</th><th>SPR0</th><th>System Clock (100 MHz) divided by</th></tr>
<tr><td>1</td><td>0</td><td>X</td><td>X</td><td>X</td><td>4</td></tr>
<tr><td>X</td><td>1</td><td>1</td><td>1</td><td>1</td><td>4</td></tr>
<tr><td colspan="5">Others</td><td>8</td></tr>
</table></td></tr>
</table>

Table 16.4 – SPIS_CNTL – Control Register

16.5.2 SPIS_STATUS – Status Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7	SPI_FLAG	RO	1’b0	Receiver data available. The flag is automatically set at the end of an SPI transfer, when a character is transferred from the shift register to the Receiver Data Register/Receiver FIFO. In SPI mode, it is cleared by reading the SPIM_STATUS register followed by accessing the SPIM_DATA register. In FIFO mode, it is cleared when the Receiver FIFO is empty.
6	WR_COL	RO	1’b0	Write collision error status flag. In SPI mode, the flag is automatically set if the SPIS_DATA register is written when the previous data has not been sent out. In FIFO Mode, the mode is set if the SPIS_DATA register is written when the Transmitter FIFO is full.

Copyright © Bridgetek Pte Ltd

				It is cleared when there is a read or write to the SPIS_DATA register again, regardless of the status of the previous data/transmitter FIFO. The software should make sure that the collision has been resolved before clearing this bit.
5	SPI_BIS	RO	1'b0	Indicates the end of transmission from the SPIS_ALT_DATA register. This flag can generate an interrupt if SPIS_TNSFR_FRMT_CNTL[6] is set. It is cleared when there is a read or write to the SPIS_ALT_DATA register.
4	Reserved	-	-	-
3	THRE	RO	1'b0	Transmitter Holding Register Empty 1: The SPI Slave is in the idle state with the Transmitter FIFO empty (FIFO mode) or when a character is transferred from the Transmitter Data Register to the shift register (SPI mode). 0: Transmission is in progress. There is data in the Transmitter Data Register (SPI mode) or at least one character by in the Transmitter FIFO (FIFO mode).
2:0	Reserved	-	-	-

Table 16.5 – SPIS_STATUS – Status Register

16.5.3 SPIS_DATA – Receiver and Transmitter Data Registers (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:0	SPDR_RX	RO	8'h00	Receiver data register. Contains data from the last Receive operation.
7:0	SPDR_TX	WO	8'h00	Transmitter data register. Contains data for the next Transmit operation

Table 16.6 – SPIS_DATA – Receiver and Transmitter Data Registers

16.5.4 SPIS_RESERVED – Reserved Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	Reserved	-	-	-

Table 16.7 – SPIS_RESERVED – Reserved Register

16.5.5 SPIS_FIFO_CNTL – FIFO Control Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7:6	RX_TRIGGER[	RW	2'b00	Receiver FIFO Trigger Level, depending on the FIFO depth specified by bit 5 of this register.

RX_TRIGGER [1:0]	Trigger level in 16-level FIFO Mode	Trigger level in 64-level FIFO mode
00	1 level	1 level
01	4 levels	16 level
10	8 levels	32 levels
11	14 levels	56 levels

Copyright © Bridgetek Pte Ltd

	1:0]			
5	FIFO_DEPTH	RW	1'b0	FIFO depth selection. 1: FIFO size is 64 bytes 0: FIFO size is 16 bytes
4	TIMEOUT	RW	1'b0	Receiver FIFO timeout enable 1: Enables timeout interrupt 0: Disables timeout interrupt
3	Reserved	-	-	-
2	TX_RST	RWAC	1'b0	Transmitter FIFO reset. Writing a 1 to this bit clears all bytes in the Transmitter FIFO and resets its counter to 0. The shift register is not affected. This bit will clear itself.
1	RX_RST	RWAC	1'b0	Receiver FIFO reset. Writing a 1 to this bit clears all bytes in the Receiver FIFO and resets its counter to 0. The shift register is not affected. This bit will clear itself.
0	FIFO_EN	RW	1'b0	Receiver and Transmitter FIFO enable. 0: Disables FIFO 1: Enables FIFO

Table 16.8 – SPIS_FIFO_CNTL – FIFO Control Register

Copyright © Bridgetek Pte Ltd

16.5.6 SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7	FIFO_EXT	RW	1'b0	FIFO extension 1: Allows 32-bit data transfer to/from FIFO. The FIFO size is also extended to 256 bytes.
6	BISINT_EN	RW	1'b0	1: Enables interrupt generation after the transfer is complete from the SPIS_ALT_DATA register.
5	MULTI_REC	RW	1'b0	1: Allows continuous reception of data without the necessity of loading the TX FIFO.
4	FAST_SPI	RW	1'b0	1: Enables the FAST mode, in which the maximum SPI clock speed supported is equal to 1/4 of the system clock. This bit is ignored when CHG_SPR is set (see the table at SPR[1:0]).
3	TX_IEN	RW	1'b0	1: Enables Transmitter FIFO empty interrupt
2:0	Reserved	-	-	-

Table 16.9 – SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register

16.5.7 SPIS_ALT_DATA – Alternative SPI Slave Data Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	Alternative SPI Mode Data register. Data transmitted through this register are done as a single channel SPI.

Table 16.10 – SPIS_ALT_DATA – Alternative SPI Slave Data Register

Copyright © Bridgetek Pte Ltd

16.5.8 SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register (address offset: 0x1C)

Bit	Name	Type	Default Value	Description
7:0	RX_FIFO_COUNT	RO	7'h00	The number of data entries available in the RX FIFO. The data can be 8-bit or 32-bit, depending on the FIFO extension setting. Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and SPI_FLAG bit is 1, it means there are 64 data entries in the FIFO.

Table 16.11 – SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register

Note: The following registers are located at a different base address (0x10960 for SPI Slave 0 and 0x10980 for SPI Slave 1).

16.5.9 SPIS_CNTL_2 – Control Register 2 (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7	Reserved	RW	1'b0	This bit should always be '0'.
6:4	Reserved	-	-	-
3	ALT_DATA_LSBF	RW	1'b0	1 : The data in the SPIS_ALT_DATA register (normally used for SPI commands) is sent with the least-significant bit first. This bit affects only the TX operation. 0: The data in the SPIS_ALT_DATA register is sent on the bus with the most-significant bit first.
2	LSBF	RW	1'b0	1: The data in SPIS_DATA is sent/received with the least-significant bit first. This bit affects both TX and RX operations. 0: The data in SPIS_DATA is sent/received with the most-significant bit first.
1	CHG_SPR	RW	1'b0	Determines the maximum SPI clock rate together with SPR[2:0] and FAST_SPI. Refer to the SPIS_CNTL register.
0	RX_EMPTY	RW	1'b1	Receiver data empty 1: The receiver data register (SPI mode) or the receiver FIFO (FIFO mode) is empty.

Table 16.12 – SPIS_CNTL_2 – Control Register 2

16.5.10 SPIS_STATUS_2 – Status Register 2 (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7	Reserved	RO	1'b0	-
6	TE	RO	1'b0	1: In SPI mode, it indicates that the transmitter data register and the shift register are both

Copyright © Bridgetek Pte Ltd

	MT			empty. In FIFO mode, it indicates that the transmitter FIFO and the shift register are both empty. 0: indicates that the transmitter data register/ transmitter FIFO or the shift register contains a character data.
5	TX_SPI_FLAG	RO	1'b0	1 : indicates that the vacant entries in the TX FIFO are more than the TX FIFO trigger level.
4	RX_FULL_OVERRIDDEN	RO	1'b0	1: RX FIFO overflows and data has been overridden. This bit is cleared by reading the SPIS_DATA register.
3	RX_FIFO_FULL	RO	1'b0	1: RX FIFO is full. 0: There is at least an empty entry in the RX FIFO.
2	RX_EMPTY	RO	1'b0	1: the receiver data register (SPI mode) or the receiver FIFO (FIFO mode) is empty.
1	TX_FI	RO	1'b0	1: TX FIFO is full. 0: There is at least an empty entry in the TX FIFO.

Copyright © Bridgetek Pte Ltd

	FO_FULL			
0	TX_EMPTY	RO	1'b0	1: the transmitter data register (SPI mode) or the transmitter FIFO (FIFO mode) is empty and the bus transaction is completed.

Table 16.13 – SPIS_STATUS_2 – Status Register 2

16.5.11 SPIS_FIFO_CNTL_2 – FIFO Control Register 2 (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7:4	Reserved	RO	4'h0	-
3:2	TX_TRIGGER [1:0]	RW	2'b00	Transmitter FIFO Trigger Level, depending on the FIFO depth specified in the SPIM_FIFO_CNTL register. **TX_TRIGGER [1:0]** / **Trigger level in 16-level FIFO Mode** / **Trigger level in 64-level FIFO mode** 00 / 1 level / 1 level 01 / 4 levels / 16 level 10 / 8 levels / 32 levels 11 / 14 levels / 56 levels
1	TX_TRIGGER _EN	RW	1'b0	1 : enables interrupt when the TX FIFO level is greater than the TX trigger level.
0	RX_TRIGGER _EN	RW	1'b1	1 : enables interrupt when the RX FIFO level is greater than the RX trigger level.

Table 16.14 – SPIS_FIFO_CNTL_2 – FIFO Control Register 2

16.5.12 SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
7	Reserved	RO	1'b0	-
6:0	TX_FIFO_COUNT	RO	7'h00	The number of data entries available in the TX FIFO. The data can be 8-bit or 32-bit, depending on the FIFO extension setting. Note that the counter can only count up to 63, then rolls back to 0. If it is 0 and SPI_FLAG bit is 1, it means there are 64 data entries in the FIFO.

Table 16.15 – SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register

Copyright © Bridgetek Pte Ltd

17 I²C Master

The I²C Master conforms to v2.1 and v3.0 of the I²C specification.

Listed below are the key features supported by the I²C master:

- Supports Standard Speed Mode (up to 100 kb/s)
- Supports Fast mode (up to 400 kb/s)
- Supports Fast-plus mode (up to 1 Mb/s)
- Supports High-speed mode (up to 3.4 Mb/s)
- Performs arbitration and clock synchronisation
- Supports multi-master systems
- Supports both 7-bit and 10-bit address modes
- Supports interrupt generation
- Supports FIFO mode

17.1 Register Summary

Listed below are the registers with their offset from the base address (0x10300). All registers can only be accessed via Byte (8-bit) mode.

Address Offset	Register	Default value	References
0x00	I2CM_SLV_ADDR – Slave Address Register	0x00	Section 17.3.1
0x01	I2CM_CNTL – Control Register	0x00	Section 17.3.2
0x01	I2CM_STATUS – Status Register	0x20	Section 17.3.3
0x02	I2CM_DATA – Receive / Transmit Data Register	0x00	Section 17.3.4
0x03	I2CM_TIMER_PERIOD – Timer Period Register	0x01	Section 17.3.5
0x03	I2CM_HS_TIME_PERIOD – High-speed Timer Period Register	0x01	Section 17.3.6
0x04	I2CM_FIFO_LEN – FIFO Mode Byte Length	0x00	Section 17.3.7
0x05	I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable	0x00	Section 17.3.8
0x06	I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending	0x00	Section 17.3.9
0x07	I2CM_FIFO_DATA – FIFO Data Register	0x00	Section 17.3.10
0x08	I2CM_TRIG – FIFO Operation Trigger Register	0x00	Section 17.3.11

Table 17.1 – Overview of I²C Master Registers

Copyright © Bridgetek Pte Ltd

17.2 I²C Master FIFO Mode

This I²C Master provides a simple mechanism to send/receive multiple bytes using a 16-byte deep FIFO.

The number of bytes to be received or transmitted can be set at **FIFO_BL** in **I2CM_FIFO_LEN – FIFO Mode Byte Length Register**. For transmission, the data should be filled to the FIFO first by writing to the data port **I2CM_FIFO_DATA**. This is also the port to read when receiving data.

The FIFO operation is triggered with a write to the **I2CM_TRIG** register. If bit 7 **(RX_OP)** of this register is set, receive operation will be performed. If that bit is cleared, transmit operation will be performed. Once triggered, **I2CM_TRIG** cannot be written until the FIFO operation completes.

After the FIFO operation starts, **I2CM_SLV_ADDR – Slave Address Register**, **I2CM_STATUS – Status Register, I2CM_CNTL – Control Register, I2CM_DATA – Receive / Transmit Data Register, I2CM_TIMER_PERIOD – Timer Period Register** and **I2CM_HS_TIMER_PERIOD – High-speed Timer Period Register** will not be accessible. Writing to them will have no effect, and reading from them returns undefined data. Hence **I2CM_SLAVE_ADDR** must be set correctly prior to triggering the FIFO operation, and bit 0 of **I2CM_SLAVE_ADDR** (R_S) should match **RX_OP**.

Various interrupts related to the FIFO operation can be triggered if enabled, namely:

- RX FIFO FULL/HALF FULL/EMPTY
- TX FIFO FULL/HALF FULL/EMPTY
- FIFO DONE.

The interrupts are enabled/disabled in **I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable Register**. The interrupt pending bits are in **I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending Register**. Writing a 1 to these pending bits will clear the bits; writing a 0 will have no effect.

The FIFO operation will terminate if one of these conditions occurs:

- The TX FIFO is empty during transmission
- The RX FIFO is full during reception
- An error has occurred such as arbitration loss or non-acknowledgement of data from the slave.

The FIFO operation will terminate properly when the number of bytes (**FIFO_BL**) to be received or transmitted has been successfully transferred. When the process is completed, the **DONE** interrupt will also be triggered. Consequently, there may be more than 1 interrupt if the FIFO is also FULL/HALF FULL/EMPTY.

It is recommended that the HALF FULL interrupt is employed in a manner that emulates a double buffer mechanism. For transmission, it's also recommended that the FIFO is fully filled and any pending interrupt is cleared prior to enabling the interrupt and triggering the FIFO operation.

17.3 Register Details

17.3.1 I2CM_SLV_ADDR – Slave Address Register (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7:1	SLV_ADDR	RW	7'h00	In 7-bit address mode, these bits store the 7-bit address. In 10-bit address mode, bits [7:3] store the 5 bits that signal 10-bit addressing (1111 0). Bits [2:1] store the two most-significant bits of the 10-bit address.
0	R_S	RW	1'b0	0: next operation is a transmission 1: next operation is a reception

Table 17.2 – I2CM_SLV_ADDR – Slave Address Register

Copyright © Bridgetek Pte Ltd

17.3.2 I2CM_CNTL – Control Register (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7	RST	WO	1'b0	Resets the I²C Master controller. The software may use this bit to re-initialize the I²C Master controller when a problem is encountered on the I2S bus.
6	SRST	WO	1'b0	If an I²C Slave device blocks the I²C bus, this bit should be set together with the RUN bit. The master will generate 9 SCL clocks without generating the START condition to recover the blocking Slave device to a known state. A STOP condition will be generated. This bit will be automatically cleared. The BUSY bit in the status register should be checked to know when this transmission is completed.
5	ADDR	WO	1'b0	Setting this together with the RUN bit will cause the generation of a START condition and transmission of a Slave address.
4	HS	WO	1'b0	Setting this together with the RUN bit switches the Bus controller into high-speed mode.
3	ACK	WO	1'b0	This bit should normally be set when the Master is in the receiver mode to generate ACK. It must be cleared when the master requires no further data from the Slave transmitter.
2	STOP	WO	1'b0	Setting this will cause the STOP condition to be generated.
1	START	WO	1'b0	Setting this will cause the START or Repeated START condition to be generated.
0	RUN	WO	1'b0	Setting this will cause the Bus controller to be active.

Table 17.3 – I2CM_CNTL – Control Register

Only certain combinations of the I2CM_CNTL bits and R_S (I2CM_SLV_ADDR[0]) are allowed. Illegal combinations are ignored. The supported combinations are listed below. 'X' denotes 'don't care'.

RST	SRST	ADDR	HS	R_S	ACK	STOP	START	RUN	Operation
0	0	0	0	0	X	0	1	1	START condition followed by SEND *(Master remains in Transmitter mode)*
0	0	0	0	0	X	1	1	1	START condition followed by SEND and STOP condition
0	0	0	0	1	0	0	1	1	START condition followed by RECEIVE operation with **negative Acknowledge** *(Master remains in Receiver mode)*
0	0	0	0	1	0	1	1	1	START condition followed by RECEIVE and STOP condition
0	0	0	0	1	1	0	1	1	START condition followed by RECEIVE *(Master remains in Receiver mode)*
0	0	0	0	1	1	1	1	1	Forbidden sequence
0	0	0	1	0	0	0	0	1	Master Code sending and switching to High-speed mode
1	0	0	X	X	X	X	X	X	I²C Master module software reset
0	1	0	0	0	0	0	0	1	Reset slaves connected to I²C bus by generating 9 SCK clocks followed by STOP
0	0	1	0	0	0	0	0	1	START condition followed by Slave Address

Table 17.4 – Control bit combinations permitted in idle state

Copyright © Bridgetek Pte Ltd

RST	SRST	ADDR	HS	R_S	ACK	STOP	START	RUN	Operation
0	0	0	0	X	X	0	0	1	SEND operation *(Master remains in Transmitter mode)*
0	0	0	0	X	X	1	0	0	STOP condition
0	0	0	0	X	X	1	0	1	SEND followed by STOP condition
0	0	0	0	0	X	0	1	1	Repeated START condition followed by SEND *(Master remains in Transmitter mode)*
0	0	0	0	0	X	1	1	1	Repeated START condition followed by SEND and STOP condition
0	0	0	0	1	0	0	1	1	Repeated START condition followed by RECEIVE operation with **negative Acknowledge** (*Master remains in Receiver mode)*
0	0	0	0	1	0	1	1	1	Repeated START condition followed by SEND and STOP condition
0	0	0	0	1	1	0	1	1	Repeated START condition followed by RECEIVE *(Master remains in Receiver mode)*
0	0	0	0	1	1	1	1	1	Forbidden sequence
1	0	0	X	X	X	X	X	X	I^2C Master module software reset
0	0	1	0	0	X	0	1	1	Repeated START condition followed by Slave Address

Table 17.5 – Control bit combinations permitted in Master Transmitter mode

RST	SRST	ADDR	HS	R_S	ACK	STOP	START	RUN	Operation
0	0	0	0	X	0	0	0	1	RECEIVE operation with negative Acknowledge *(Master remains in Receiver mode)*
0	0	0	0	X	X	1	0	0	STOP condition.*
0	0	0	0	X	0	1	0	1	RECEIVE followed by STOP condition
0	0	0	0	X	1	0	0	1	RECEIVE operation *(Master remains in Receiver mode)*
0	0	0	0	X	1	1	0	1	Forbidden sequence
0	0	0	0	1	0	0	1	1	Repeated START condition followed by RECEIVE operation with **negative Acknowledge** *(Master remains in Receiver mode)*
0	0	0	0	1	0	1	1	1	Repeated START condition followed by RECEIVE and STOP condition
0	0	0	0	1	1	0	1	1	Repeated START condition followed by RECEIVE *(Master remains in Receiver mode)*
0	0	0	0	0	X	0	1	1	Repeated START condition followed by SEND *(Master remains in Transmitter mode)*
0	0	0	0	0	X	1	1	1	Repeated START condition followed by SEND and STOP condition
1	0	0	X	X	X	X	X	X	I^2C Master module software reset

Table 17.6 – Control bit combinations permitted in Master Receiver mode

Note:
** In Master Receiver mode, STOP condition should be generated only after Data negative Acknowledge (executed by Master) or Address negative Acknowledge (executed by Slave).*

Copyright © Bridgetek Pte Ltd

17.3.3 I2CM_STATUS – Status Register (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6	BUS_BUSY	RO	1'b0	1: indicates the Bus is Busy, and access is not possible. It's set/cleared by START/STOP conditions.
5	IDLE	RO	1'b1	1: indicates the I^2C Master is in the idle state.
4	ARB_LOST	RO	1'b0	1: indicates that during the last operation, the I^2C Master lost the arbitration
3	DATA_ACK	RO	1'b0	1: indicates that during the last transmit operation the data wasn't acknowledged.
2	ADDR_ACK	RO	1'b0	1: indicates that during the last operation the slave address wasn't acknowledged.
1	ERR	RO	1'b0	1: indicates an error occurred during the last operation. The error can be either ARB_LOST, DATA_ACK or ADDR_ACK. If interrupt is enabled by I2C_INT, an interrupt is also triggered.
0	BUSY	RO	1'b0	1: indicates that the Bus controller is receiving / transmitting data on the bus, and other status bits of the Status Register are not valid.

Table 17.7 – I2CM_STATUS – Status Register

17.3.4 I2CM_DATA – Receive / Transmit Data Register (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	When read, this is the data received in the last transaction. When written, this is the data to be transmitted in the next transaction. In 10-bit address mode, this register holds the 8 least-significant bits of the 10-bit address. It should be written to after the 2 most-significant bits of the address are written to the Slave Address Register.

Table 17.8 – I2CM_DATA – Receive / Transmit Data Register

To generate a wide range of SCL frequencies, the I2C Master have a built-in 8-bit timer. Physically, there are two registers located at address 0x03 to setup the timer. The first **(I2CM_TIMER_PERIOD)** is used for Standard, Fast and Fast-plus speed modes. The second **(I2CM_HS_TIMER_PERIOD)** is used for High-speed mode (with an exception of bit 6 – FAST, refer to the register details below for more information). To access I2CM_TIMER_PERIOD, the most-significant bit of the input data should be '0'. If the bit is '1', I2CM_HS_TIMER_PERIOD is used. The registers are switched automatically. There is no need to reprogram their values while changing the HS bit of the Control Register. Both registers have a value of 0x01 by default.

17.3.5 I2CM_TIMER_PERIOD – Timer Period Register (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7	TIMER_SEL	RW	1'b0	This bit selects the Standard/Fast/Fast-plus or the High-speed Timer Period register. Write 0 to select Standard/Fast/Fast-plus Timer Period register (see also I2CM_HS_TIMER_PERIOD register).
6:0	TIMER_PRD	RW	7'h01	Frequency scaler used in Standard/Fast/Fast-plus modes to calculate the I2C clock period

Copyright © Bridgetek Pte Ltd

				(SCL_PRD). For Fast/Fast-plus modes, bit 6 (FAST) of I2CM_HS_TIMER_PERIOD also needs to be set. For Standard mode, it should be cleared. See the bit detail in I2CM_HS_TIMER_PERIOD. In Standard speed mode: SCL_PRD = (TIMER_PRD * 2 + 1) * 4 * CLK_PRD In Fast/Fast-plus speed mode: SCL_PRD = (TIMER_PRD * 2 + 1) * 3 * CLK_PRD where CLK_PRD is 10 ns, based on the 100-MHz system clock.

Table 17.9 – I2CM_TIMER_PERIOD – Timer Period Register

17.3.6 I2CM_HS_TIMER_PERIOD – High-speed Timer Period Register (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7	TIMER_SEL	WO	1'b0	This bit selects the Standard/Fast/Fast-plus or the High-speed Timer Period register. Write 1 to select High-speed Timer Period register (see also I2CM_TIMER_PERIOD register).
6	FAST	WO	1'b0	This bit informs the I^2C Master Controller to use the appropriate (internal) timing parameters for Standard, Fast and Fast-plus modes. In High-speed mode, it is ignored. 1: selects timing parameters for Fast and Fast-plus modes. 0: selects timing parameters for Standard mode.
5	Reserved	-	-	-
4:0	TIMER_PRD	WO	1'b1	Frequency scaler used in High-speed mode to calculate the I2C clock period (SCL_PRD). SCL_PRD = (TIMER_PRD * 2 + 1) * 3 * CLK_PERIOD where CLK_PRD is 10 ns, based on the 100-MHz system clock.

Table 17.10 – I2CM_HS_TIMER_PERIOD – High-speed Timer Period Register

17.3.7 I2CM_FIFO_LEN – FIFO Mode Byte Length (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7:0	FIFO_BL	RW	8'h00	Number of bytes (FIFO_BL + 1) to transmit/receive when FIFO mode is enabled.

Table 17.11 – I2CM_FIFO_LEN – FIFO Mode Byte Length

17.3.8 I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7	DONE	RW	1'b0	1: enables FIFO Operation Completed interrupt
6	I2C_INT	RW	1'b0	1: enables Non-FIFO Transfer Completed interrupt and Transfer Error interrupt.

Copyright © Bridgetek Pte Ltd

				Non-FIFO Transfer Completed interrupt is triggered when a transfer (transmission/reception) of a byte is completed. Transfer Error interrupt is triggered when an error occurs during the data transfer. See I2CM_STATUS[1].
5	RX_FULL	RW	1'b0	1: enables RX FIFO Full interrupt
4	RX_HALF	RW	1'b0	1: enables RX FIFO Half Full interrupt
3	RX_EMPTY	RW	1'b0	1: enables RX FIFO Empty interrupt
2	TX_FULL	RW	1'b0	1: enables TX FIFO Full interrupt
1	TX_HALF	RW	1'b0	1: enables TX FIFO Half Full interrupt
0	TX_EMPTY	RW	1'b0	1: enables TX FIFO Empty interrupt

Table 17.12 – I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable

17.3.9 I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06)

Bit	Name	Type	Default Value	Description
7	DONE	RW1C	1'b0	1: FIFO Operation Completed interrupt pending
6	I2C_INT	RW1C	1'b0	1: Non-FIFO Transfer Completed interrupt or Transfer Error interrupt pending
5	RX_FULL	RW1C	1'b0	1: RX FIFO Full interrupt pending
4	RX_HALF	RW1C	1'b0	1: RX FIFO Half Full interrupt pending
3	RX_EMPTY	RW1C	1'b0	1: RX FIFO Empty interrupt pending
2	TX_FULL	RW1C	1'b0	1: TX FIFO Full interrupt pending
1	TX_HALF	RW1C	1'b0	1: TX FIFO Half Full interrupt pending
0	TX_EMPTY	RW1C	1'b0	1: TX FIFO Empty interrupt pending

Table 17.13 – I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending

17.3.10 I2CM_FIFO_DATA – FIFO Data Register (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7:0	FIFO_READ_DATA	RO	8'h00	Data port for reading data during FIFO operations.
7:0	FIFO_WRITE_DATA	WO	8'h00	Data port for writing data during FIFO operations.

Table 17.14 – I2CM_FIFO_DATA – FIFO Data Register

17.3.11 I2CM_TRIG – FIFO Operation Trigger Register (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7	RX_OP	RW	1'b0	A write to this register triggers the FIFO mode operation. Set this bit to 1 for RX, and 0 for TX FIFO operations. The operation will end when FIFO_BL expires.
6:0	Reserved	-	-	-

Table 17.15 – I2CM_TRIG – FIFO Operation Trigger Register

Copyright © Bridgetek Pte Ltd

18 I²C Slave

The I²C Slave conforms to v2.1 and v3.0 of the I²C specification.

Listed below are the key features supported by the I²C slave:

- Supports Standard Speed Mode (up to 100kb/s)
- Supports Fast mode (up to 400kb/s)
- Supports Fast-plus mode (up to 1Mb/s)
- Supports High-speed mode (up to 3.4Mb/s)
- Performs arbitration and clock synchronisation
- Supports interrupt generation
- Supports FIFO mode
- Supports clock stretching

18.1 Register Summary

Listed below are the registers with their offset from the base address (0x10310). All registers can only be accessed via Byte (8-bit) mode.

Address Offset	Register	Default value	References
0x00	I2CS_OWN_ADDR – Own Address Register	0x00	Section 18.3.1
0x01	I2CS_CNTL – Control Register	0x00	Section 18.3.2
0x01	I2CS_STATUS – Status Register	0x00	Section 18.3.3
0x02	I2CS_DATA – Receive / Transmit Data Register	0x00	Section 18.3.4
0x04	I2CS_FIFO_STATUS – FIFO Mode Status Register	0x00	Section 18.3.5
0x05	I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable	0x00	Section 18.3.6
0x06	I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending	0x00	Section 18.3.7
0x07	I2CS_FIFO_DATA – FIFO Data Register	0x00	Section 18.3.8

Table 18.1 – Overview of I2C Master Registers

18.2 I²C Slave FIFO Mode

The I²C Slave provides a simple mechanism to send/receive multiple bytes using a 16-byte deep FIFO.

For transmission, the data should be filled first by writing to the data port **I2CS_FIFO_DATA**. This is also the port to read when receiving data.

The FIFO operation is armed when **DO_OP** in **I2CS_FIFO_STATUS – FIFO Mode Status Register** is set. The process starts when there are activities from the master. Once the FIFO mode

Copyright © Bridgetek Pte Ltd

operation starts, **I2CS_OWN_ADDR – Own Address Register**, **I2CS_STATUS – Status Register, I2CS_CNTL – Control Register** will not be accessible. Writing to them will have no effect, and reading from them returns undefined data. Hence **I2CS_OWN_ADDR** must be set correctly prior to triggering the FIFO mode operation.

Various interrupts related to the FIFO operations can be triggered if enabled, namely:

- RX FIFO FULL/HALF FULL/EMPTY
- TX FIFO FULL/HALF FULL/EMPTY
- FIFO DONE

The interrupts are enabled/disabled in **I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable Register**. The interrupt pending bits are in **I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending Register**. Writing a 1 to these pending bits will clear the bits; writing a 0 will have no effect.

The FIFO mode operation will terminate if the FIFO is empty during transmission or if the FIFO is full during reception. An error will also be generated. **RX_ERROR** in **I2CS_FIFO_STATUS** denotes a reception error while **TX_ERROR** denotes a transmission error. These bits can only be cleared by writing a 1 to them. When there is an error, **DO_OP** should be cleared, and the I2C Slave should be reset with **I2C_RST** in **I2CS_CNTL – Control Register.**

When the process is completed whether normally or due to an error, the FIFO **DONE** interrupt will also be triggered. Consequently, there may be more than 1 interrupt if the FIFO is also FULL/HALF FULL/EMPTY.

18.3 Register Details

18.3.1 I2CS_OWN_ADDR – Own Address Register (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6:0	OWN_ADDR	RW	7'h00	This is the seven address bits of the I^2C Slave controller.

Table 18.2 – I2CS_OWN_ADDR – Own Address Register

18.3.2 I2CS_CNTL – Control Register (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7	I2C_RST	WO	1'b0	Setting this bit will reset the whole Slave controller.
6	DEV_ACTV	WO	1'b0	Device Active 1: enables the I^2C Slave Controller operations 0: disables the I^2C Slave Controller operations Writing 1 sets DEV_ACTV to 1 immediately while writing 0 will not be effective immediately if there is any on-going transmission. It's suggested that the DEV_ACTV bit the Status Register is polled if a 0 is written to this bit.
5	R	-	-	-

Copyright © Bridgetek Pte Ltd

:4	eserved			
3	REC_FIN_CLR	WO	1'b0	Writing 1 to this bit clears the REC_FIN bit from the Status Register.
2	SEND_FIN_CLR	WO	1'b0	Writing 1 to this bit clears the SEND_FIN bit from the Status Register.
1:0	Reserved	-	-	-

Table 18.3 – I2CS_CNTL – Control Register

18.3.3 I2CS_STATUS – Status Register (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6	DEV_ACTV	RO	1'b0	Device Active 1: the I2C Slave Controller operations are enabled 0: the I2C Slave Controller operations are disabled This bit should be polled when a '0' is written to the DEV_ACTV bit of the Control Register.
5	Reserved	-	-	-
4	BUS_ACTV	RO	1'b0	1: indicates that there is a transmission (sending, receiving or own address detection) in progress

Copyright © Bridgetek Pte Ltd

Bit	Name	Type	Default Value	Description
3	REC_FIN	RO	1'b0	1: indicates that the Master has ended the transmit operation. It means no more RX_REQ will be set during this single or bursts receive operation. It is cleared by writing 1 to REC_FINCLR bit in the Control register.
2	SEND_FIN	RO	1'b0	1: indicates that the Master has ended the receive operation. It means no more TX_REQ will be set during this single or burst send operation. It is cleared by writing 1 to SEND_FINCLR bit in the Control register.
1	TX_REQ	RO	1'b0	1: indicates the Slave controller is addressed as transmitter and requires data from the host device.
0	RX_REQ	RO	1'b0	1: indicates the Slave controller has received data from the Master. It is automatically cleared by reading of I2CS_DATA.

Table 18.4 – I2CS_STATUS – Status Register

18.3.4 I2CS_DATA – Receive / Transmit Data Register (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7:0	DATA	RW	8'h00	When read, this is the data received in the last transaction. When written, this is the data to be transmitted in the next transaction.

Table 18.5 – I2CS_DATA – Receive / Transmit Data Register

18.3.5 I2CS_FIFO_STATUS – FIFO Mode Status Register (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7	DO_OP	RW	1'b0	Write 1 to start FIFO operation
6:2	Reserved	-	-	-
1	RX_ERROR	RW1C	1'b0	RX FIFO error, see section 18.2
0	TX_ERROR	RW1C	1'b0	TX FIFO error, see section 18.2

Table 18.6 – I2CS_FIFO_STATUS – FIFO Mode Status Register

18.3.6 I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7	DONE	RW	1'b0	1: enables FIFO Operation Completed Interrupt
6	I2C_INT	RW	1'b0	1: enables Non-FIFO Transfer Completed interrupt Non-FIFO Transfer Completed interrupts is triggered when a transfer (transmission/reception) of a byte is completed.
5	RX_FULL	RW	1'b0	1: enables RX FIFO Full interrupt
4	RX_HALF	RW	1'b0	1: enables RX FIFO Half Full interrupt
3	RX_EMPTY	RW	1'b0	1: enables RX FIFO Empty interrupt
2	TX_FULL	RW	1'b0	1: enables TX FIFO Full interrupt
1	TX_HALF	RW	1'b0	1: enables TX FIFO Half Full interrupt
0	TX_EMPTY	RW	1'b0	1: enables TX FIFO Empty interrupt

Table 18.7 – I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable

18.3.7 I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending (address offset: 0x06)

Bit	Name	Type	Default	Description

Copyright © Bridgetek Pte Ltd

			Value	
7	DONE	RW1C	1'b0	1: FIFO Operation Completed interrupt pending
6	I2C_INT	RW1C	1'b0	1: Non-FIFO Transfer Completed interrupt pending
5	RX_FULL	RW1C	1'b0	1: RX FIFO Full interrupt pending
4	RX_HALF	RW1C	1'b0	1: RX FIFO Half Full interrupt pending
3	RX_EMPTY	RW1C	1'b0	1: RX FIFO Empty interrupt pending
2	TX_FULL	RW1C	1'b0	1: TX FIFO Full interrupt pending
1	TX_HALF	RW1C	1'b0	1: TX FIFO Half Full interrupt pending
0	TX_EMPTY	RW1C	1'b0	2: TX FIFO Empty interrupt pending

Table 18.8 – I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending

18.3.8 I2CS_FIFO_DATA – FIFO Data Register (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7:0	FIFO_READ_DATA	RO	8'h00	Data port for reading data during FIFO operations.
7:0	FIFO_WRITE_DATA	WO	8'h00	Data port for writing data during FIFO operations.

Table 18.9 – I2CS_FIFO_DATA – FIFO Data Register

Copyright © Bridgetek Pte Ltd

19 RTC

This is a Real Time Clock (RTC) running off a dedicated 32.768 kHz oscillator. It is powered by a 1.2V input (VBAT). If power is lost to VBAT, all RTC register values will reset to their default values.

19.1 Register Summary

Listed below are the registers with their offset from the base address (0x10900). All registers can be accessed via Double-Word (32-bit) mode.

Address Offset	Register	Default Value	References
0x00	RTC_TIME – Time Register	0x0000 0000	Section 19.3.1
0x04	RTC_DATE – Date Register	0x0001 0101	Section 19.3.2
0x08	RTC_AALARM1 – Auto Alarm 1 Register	0x0000 0000	Section 19.3.3
0x0C	RTC_AALARM2 – Auto Alarm 2 Register	0x0000 0000	Section 19.3.4
0x10	RTC_CTRL – Control Register	0x0000 0000	Section 19.3.5
0x14	RTC_STAT – Status Register	0x0000 0000	Section 19.3.6
0x18	RTC_TRIM – Trimming Register	0x03E8 7FFF	Section 19.3.7
0x1C	RTC_SLEEP_TIME – Sleep Time Register	0x0000 0000	Section 19.3.8
0x20	RTC_SLEEP_DATE – Sleep Date Register	0x0001 0101	Section 19.3.9
0x24	RTC_COMP_PER – Compare Period Register	0x0000 0000	Section 19.3.10
0x28	RTC_MISC – Miscellaneous Register	0x0000 0003	Section 19.3.11

Table 19.1 – Overview of RTC Registers

19.2 Functional Description

The RTC consists of two sub modules, RTC Alive and RTC Wrapper. The function of each module is described below.

19.2.1 RTC Alive Module

RTC Alive will remain functioning when the system enters the sleep mode. It only maintains the main registers: Counter, Alarm, Control, Sleep and Trimming. These registers will be sent to RTC Wrapper either:

- when the system wakes up,
- per second or,
- on-demand

RTC Alive divides the 32.768 kHz input clock to generate a 1 Hz clock to be used internally. It also adds and removes trimming cycles as well as counts for time and date.

Copyright © Bridgetek Pte Ltd

19.2.2 RTC Wrapper Module

RTC Wrapper controls the operation of the RTC when the system power is turned on. It updates the RTC counters and status from RTC Alive every second as well as asserts alarms and interrupts to the CPU.

19.2.3 RTC Operation

Upon reset, the RTC will be in standby mode with no counter nor timer in operation. The user must enable the RTC by writing '1' into bit 0 of the Control Register (RTCEN bit).

The user can read and write RTC registers at any time but two continuous write cycles should be longer than 4 RTC clock cycles#. This is the time needed to transfer data between Wrapper and Alive modules. Note that to read the Date Register, the Time Register must be read in advance.

Once the timers match the values of the Alarm Registers, the corresponding ALARM signal will be asserted. These alarms will cause an interrupt if it is enabled in the Control Register.

When the system power is turned off, the sleep time and date will be recorded in the Alive module. After system wake-up, all registers values will be updated from Alive to Wrapper and ready to read after 2 RTC clock cycles#. The user should not write/read RTC registers during this time. If bit 7 (ARFEN) of the Control Register is '1', Time and Date values will be updated from Alive to Wrapper per second. Otherwise, the user must do the on-demand update by setting bit 8 (MRFEN) of the Control Register and read the Time and Date Registers after 2 RTC clock cycles#. An on-demand update will update all registers.

The RTC can automatically detect if the oscillator is not running by comparing the old and new values of the Alive counter periodically. If these two values are the same, an interrupt will be asserted. The compare period is programmable in the Compare Period Register.

19.2.4 RTC Auto Calibration using a 32,768 Hz Reference Clock

The RTC has a feature to compensate for the frequency variations caused by crystal tolerance and temperature. On-chip calibration and trimming support is provided to adjust the RTC's frequency.

Two counters are used during the calibration process: one for counting the RTC reference clock (typically generated by a signal generator) and one for counting the RTC clock from the 32.768 kHz crystal. The difference between the two counters at the end of the trimming period is recorded as the trimming value. The trimming period is determined by bit 4 and 5 of the RTC_CTRL Register (see Table 19.2). The trimming value and trimming period are loaded to the RTC_TRIM Register automatically when the process is completed. The 1.2V input on VBAT has to be presented to maintain the value in the register.

During normal RTC operation, the trimming value is added (or subtracted) at the end of each trimming period to compensate for the difference between the 32.768 kHz RTC clock and the reference clock used during calibration.

To perform the calibration, the user has to provide an accurate 32,768 Hz reference clock to the RTC_REF pin and follow the procedure below. The RTC accuracy is the sum of the frequency accuracy in PPM of the reference clock and the accuracy per trimming code in PPM (see Table 19.3).

1. Ensure that the RTC is disabled by clearing the RTCEN bit in the RTC_CTRL Register.
2. Configure GPIO6 to be an input for the clock reference (RTC_REF).
3. Configure the desired calibration time by setting RTC_CTRL[5] and RTC_CTRL[4] bits. Then set AUCAL bit, INTEN bit, and WRST/BS bits in the RTC_CTRL register to start the calibration. The RTC_CTRL[5] and RTC_CTRL[4] bits are set according to the selected calibration time.

Copyright © Bridgetek Pte Ltd

RTC_CTRL[5]	RTC_CTRL[4]	Calibration Time (i.e., Trimming Period)
0	0	1 second
0	1	60 seconds
1	0	600 seconds
1	1	1000 seconds

Table 19.2 – Calibration Time

4. Wait until the calibration period has passed, either by using a timer or by polling the AUCAL_DONE bit in the Status Register. The trimming value and trimming period will be automatically recorded in the Trimming Register (RTC_TRIM) afterwards. After the calibration process is completed, it should be disabled by writing '1' to the AUCAL_DONE bit in the Status Register and writing 0x200 to the Control Register.
5. The user may now program a time and a date and then set the RTCEN bit and the WRST/BS bit in the Control Register. The RTC will start normal operation with the new trimmed time and trimmed value.

The accuracy of the trimming value from the calibration process depends upon the selected calibration time. Table 19.3 shows the accuracy per trimming code versus the calibration time. The frequency range to be calibrated is also referenced in the table. A trimming value from the calibration process can be applied to the different frequency ranges based on the calibration time.

Calibration Time (in seconds)	Accuracy per Trimming Code (PPM)	Minimum Frequency (Hz) to be calibrated	Maximum Frequency (Hz) to be calibrated
1	30.51757813	16385	49153
60	0.508626302	32494.95	33041.083
600	0.05086263	32740.695	32795.3083
1000	0.030517578	32751.617	32784.385

Table 19.3 – Calibration Time/Accuracy vs. Frequency Range

19.3 Register Details

19.3.1 RTC_TIME – Time Register (address offset: 0x00)

- Type : RW
- Reset Value : 0x0000 0000
- Description : Contains the Second, Minute, Hour counters.

Bit / Byte	7	6	5	4	3	2	1	0	Function	Range
0	0	10 Seconds			Seconds				Seconds	00 – 59
1	0	10 Minutes			Minutes				Minutes	00 – 59
2	0	1:12 0:24	0: AM 1: PM 20 Hours	10 Hours	Hours				Hours	01 – 12 +AM/PM 00 – 23
3	0	0	0	0	0	0	0	0	Unused	

Table 19.4 – RTC_TIME – Time Register

Copyright © Bridgetek Pte Ltd

19.3.2 RTC_DATE – Date Register (address offset: 0x04)

- Type : RW
- Reset Value : 0x0001 0101
- Description : Contains the Day, Date, Month, Year counters.

Byte \ Bit	7	6	5	4	3	2	1	0	Function	Range
0	0	0	0	0	0	Day			Day	1 – 7
1	0	0	10 Date		Date				Date	01 – 31
2	Century	0	0	10 Month	Month				Month	01 – 12 + Century
3	10 Year				Year				Year	00 – 99

Table 19.5 – RTC_DATE – Date Register

19.3.3 RTC_AALARM1 – Auto Alarm 1 Register (address offset: 0x08)

- Type : RW
- Reset Value : 0x0000 0000
- Description : When counters match the configured values of this register, Alarm 1 will be asserted.

Byte \ Bit	7	6	5	4	3	2	1	0	Function	Range
0	A1M1	10 Seconds			Seconds				Seconds	00 – 59
1	A1M2	10 Minutes			Minutes				Minutes	00 – 59
2	A1M3	1:12 0:24	0:AM 1:PM 20 Hours	10 Hours	Hours				Hours	01 – 12 +AM.PM 00 – 23
3	A1M4	0:Date 1: Day	10 Date		Day/Date				Date Day	01-07 01-31

Table 19.6 – RTC_AALARM1 – Auto Alarm 1 Register

DAY/DATE (Byte 3 Bit 6)	AALARM1 Register Mask (bit 7)				Alarm Rate
	A1M4	A1M3	A1M2	A1M1	
X	1	1	1	1	Alarm once per second
X	1	1	1	0	Alarm when seconds match
X	1	1	0	0	Alarm when minutes and seconds match
X	1	0	0	0	Alarm when hours, minutes and seconds match
0	0	0	0	0	Alarm when date, hours, minutes and second match
1	0	0	0	0	Alarm when day, hours, minutes and second match

Table 19.7 – Alarm 1 Mask Bits

Note: *Configurations not listed in this table result in illogical operation.*

Copyright © Bridgetek Pte Ltd

19.3.4 RTC_AALARM2 – Auto Alarm 2 Register (address offset: 0x0C)

- Type : RW
- Reset Value : 0x0000 0000
- Description : When counters match the configured values of this register, Alarm 2 will be asserted.

Byte \ Bit	7	6	5	4	3	2	1	0	Function	Range
0	A2M1	10 Seconds			Seconds				Seconds	00 – 59
1	A2M2	10 Minutes			Minutes				Minutes	00 – 59
2	A2M3	1:12 0:24	0:AM 1:PM 20 Hours	10 Hours	Hours				Hours	01 – 12 +AM.PM 00 – 23
3	A2M4	0:Date 1: Day	10 Date		Day/Date				Date Day	01-07 01-31

Table 19.8 – RTC_AALARM2 – Auto Alarm 2 Register

DAY/DATE (Byte 3 Bit 6)	AALARM2 Register Mask (bit 7)				Alarm Rate
	A2M4	A2M3	A2M2	A2M1	
X	1	1	1	1	Alarm once per second
X	1	1	1	0	Alarm when seconds match
X	1	1	0	0	Alarm when minutes and seconds match
X	1	0	0	0	Alarm when hours, minutes and seconds match
0	0	0	0	0	Alarm when date, hours, minutes and second match
1	0	0	0	0	Alarm when day, hours, minutes and second match

Table 19.9 – Alarm 2 Mask Bits

Note: *Configurations not listed in this table result in illogical operation.*

19.3.5 RTC_CTRL – Control Register (address offset: 0x10)

Bit	Name	Type	Default Value	Description
31:10	Reserved	-	-	-
9	WRST/BS	RW	1'b0	Write '1' to this bit to start writing target registers. Read this bit to check the status. 1: Busy 0: Ready This bit is auto cleared.
8	MRFEN	RW	1'b0	Manual Refresh Enable Write 1 to this bit to update time and date counters from Alive to Wrapper once. This bit will be cleared automatically after updating. Poll this bit to get the status of the Manual Refresh function. 1: Updating 0: Done

Copyright © Bridgetek Pte Ltd

7	ARFEN	RW	1'b0	Auto Refresh Enable When this bit is turned on, time and date counters from Alive will be updated to the Wrapper registers every second.
6	AUCAL	RW	1'b0	Write '1' to this bit to enable auto calibration. The calibration time depends on bit 5 and 4 of this register. See Table 19.2 for the calibration time.
5	IOSIE/ CATIM[1]	RW	1'b0	Internal Oscillator Stop Interrupt Enable / Calibration Time bit 1 An Oscillator Stop event will cause an interrupt if this bit is enabled. For calibration, this is bit 1 of the calibration time.
4	CATIM[0]	RW	1'b0	Calibration Time bit 0
3	A2IE	RW	1'b0	Alarm 2 Interrupt Enable Alarm 2 will cause an interrupt if this bit is enabled.
2	A1IE	RW	1'b0	Alarm 1 Interrupt Enable Alarm 1 will cause an interrupt if this bit is enabled.
1	INTEN	RW	1'b0	Interrupt Enable bit. When this bit is disabled, no interrupt occurs. ***Note:*** *This bit must be '0' during auto calibration.*
0	RTCEN	RW	1'b0	RTC Enable 1: Enables the RTC. 0: Disables the RTC. Interrupt and alarms will also be disabled, and the RTC counters will be held.

Table 19.10 – RTC_CTRL – Control Register

19.3.6 RTC_STAT – Status Register (address offset: 0x14)

Bit	Name	Type	Default Value	Description
31:7	Reserved	-	-	-
6	AUCAL_DONE	RW1C	1'b0	1: Auto Calibration Completed. This bit should be cleared by writing '1' to it after the calibration process.
5	IOSINT	RW1C	1'b0	Internal Oscillator Stop Interrupt 1: Two continuous values are the same after a defined period of time, i.e., the oscillator has stopped. If RTC_CTRL[5] (IOSIE) is enabled, an interrupt will be asserted.
4	Reserved	-	-	-
3	A2INT	RW1C	1'b0	Alarm 2 Interrupt 1: the timers match the value of the Auto Alarm 2 Register; an interrupt will be asserted if RTC_CTRL[3] (A2IE) is set.
2	A1INT	RW1C	1'b0	Alarm 1 Interrupt 1: the timers match the value of the Auto Alarm 1 Register; an interrupt will be asserted if RTC_CTRL[2] (A1IE) is set.
1:0	Reserved	-	-	-

Table 19.11 – RTC_STAT – Status Register

Copyright © Bridgetek Pte Ltd

19.3.7 RTC_TRIM – Trimming Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
31:16	TRIM_TIME	RW	16'h03E8	The trimming period (in seconds). The default value is 0x03E8, or 1000 seconds. After auto calibration, the new value will be loaded automatically. The user may also write to this register manually, but the value should match the trimming period used for the calibration.
15:0	TRIM_VAL	RW	16'h7FFF	The trimming value. After auto calibration, the new value will be loaded automatically. The user may also write to this register manually.

Table 19.12 – RTC_TRIM – Trimming Register

19.3.8 RTC_SLEEP_TIME Sleep Time Register (address offset: 0x1C)

- Type : RW
- Reset Value : 0x00–00 0000
- Description : Contains the Second, Minute, Hour values when the system power is turned off.

Byte \ Bit	7	6	5	4	3	2	1	0	Function	Range
0	0	10 Seconds			Seconds				Seconds	00 – 59
1	0	10 Minutes			Minutes				Minutes	00 – 59
2	0	1:12 0:24	0: AM 1: PM 20 Hours	10 Ho urs	Hours				Hours	01 – 12 + AM.PM 00 – 23
3	0	0	0	0	0	0	0	0	Unused	

Table 19.13 – RTC_COMP_VERSION – Component Version Register

19.3.9 RTC_SLEEP_DATE – Sleep Date Register (address offset: 0x20)

- Type : RW
- Reset Value : 0x0001 0101
- Description : Contains Day, Date, Month, Year counters when the system power is turned off.

Byte \ Bit	7	6	5	4	3	2	1	0	Function	Range
0	0	0	0	0	0	Day			Day	1 – 7
1	0	0	10 Date		Date				Date	01 – 31
2	Century	0	0	10 Month	Month				Month	01 – 12 + Century
3	10 Year				Year				Year	00 – 99

Table 19.14 – RTC_SLEEP_DATE - Sleep Date Register

Copyright © Bridgetek Pte Ltd

19.3.10 RTC_COMP_PER – Compare Period Register (address offset: 0x20)

- Type : RW
- Reset Value : 0x0000 0000
- Description : Contains the Oscillator Stop compare period which is automatically detected. The user should set the compare period within the range of 100ms ~ 2s based on the system clock cycle, where the system clock is 100 MHz.

Byte \ Bit	7	6	5	4	3	2	1	0	Function
0	Unused					0	0	0	Auto detection is turned off
						0	0	1	Auto detect per 100,000 system clock cycles
						0	1	0	Auto detect per 1,000,000 system clock cycles
						0	1	1	Auto detect per 10,000,000 system clock cycles
						1	0	0	Auto detect per 100,000,000 system clock cycles
1, 2, 3	Reserved								

Table 19.15 – RTC_COMP_PER – Compare Period Register

19.3.11 RTC_MISC – Miscellaneous Register

Bit	Name	Type	Default Value	Description
31:17	Reserved	-	-	-
16	DUMMY	RW	1'b0	This is a Read/Write dummy bit. This bit can be used to indicate whether the RTC power has been interrupted after the system is initialized. For example, write "1" to this bit during system initialization. After that, if both Battery and system power are removed, this bit will be reset to "0".
15:9	Reserved	-	-	-
8	SEL_VBAT	RW1C	1'b0	Battery Low Detection Threshold Selection. The battery low interrupt will be triggered if the battery level falls below the threshold selected here. 0: 84% of VCCK 1: 80% of VCCK where VCCK is the core voltage (~1.2V).
7:0	Reserved	-	-	-

Table 19.16 – RTC_MISC – Miscellaneous Register

Copyright © Bridgetek Pte Ltd

20 PWM

The device supports 8 separate independent PWM channels, though only 7 channels (except channel 3) are bonded out to the package pins. All channels share an 8-bit pre-scaler to scale the system clock frequency to the desired channels.

Each channel has its own 16-bit comparator value. This is the value that would be matched to a preset 16-bit counter. When a channel's 16-bit comparator value matches that of the 16-bit counter, the corresponding PWM channel output will toggle. This 16-bit comparator value will continue to count until it reaches its preset value, and the counter will just roll over.

A special feature allows the 8 channels each to also toggle its own output based on the comparison results of other channels. Hence each channel potentially can have up to 8 toggle edges.

The PWM can be generated as a multi-shot or continuously. When defined as a multi-shot, an interrupt may be generated at the end of the PWM production.

Channels 0 and 1 can double as a stereo 11.025 kHz or 22.050 kHz PWM audio channel. Once it's set up, the 16-bit or 8-bit PWM audio data can be downloaded to the PWM's local FIFO which can hold up to 64 stereo or 128 mono audio samples. The data will be played back based on the pre-scaler and 16-bit counter and the data will be automatically scaled to fit the playback period if necessary.

The FIFO can generate several interrupts for FIFO management. They are FIFO full, empty, half-empty, overflow and underflow interrupts. These interrupts can be individually masked if required.

As there are 16-bit registers but the read/write operations are performed in 8-bit mode, the 16-bit registers must be read/write in a specific manner.

For write operations, the LSB must be written first. Only when the MSB is written will the full 16-bit data get transferred to the MSB and LSB registers together. If MSB is written without LSB being written first, LSB assumes the value 0.

For read operations, the LSB must be read first. The MSB will be held in a temporary register that can be read with a subsequent MSB read. It is not recommended to read MSB of a 16-bit register without first performing the corresponding LSB read. The MSB value should be treated as undefined in such cases.

20.1 Register Summary

Listed below are the registers with their offset from the base address (0x103C0). All registers can only be accessed via Byte (8-bit) mode but the FIFO can only be accessed via Word (16-bit) mode.

Address Offset	Register	Default value	References
0x00	PWM_CTRL0 – PCM Control Register	0x00	Section 20.3.1
0x01	PWM_CTRL1 – PWM Control Register	0x00	Section 20.3.2
0x02	PWM_PRESCALER – PWM Pre-scaler Register	0x00	Section 20.3.3
0x03	PWM_CNTL – PWM Counter Register (LSB)	0x00	Section 20.3.4
0x04	PWM_CNTH – PWM Counter Register (MSB)	0x00	Section 20.3.5
0x05	PWM_CMP0L – Comparator 0 Value Register (LSB)	0x00	Section 20.3.6
0x06	PWM_CMP0H – Comparator 0 Value Register (MSB)	0x00	Section 20.3.7
0x07	PWM_CMP1L – Comparator 1 Value Register (LSB)	0x00	Section 20.3.8
0x08	PWM_CMP1H – Comparator 1 Value Register (MSB)	0x00	Section 20.3.9
0x09	PWM_CMP2L – Comparator 2 Value Register (LSB)	0x00	Section 20.3.10

Copyright © Bridgetek Pte Ltd

0x0A	PWM_CMP2H – Comparator 2 Value Register (MSB)	0x00	Section 20.3.11
0x0B	PWM_CMP3L – Comparator 3 Value Register (LSB)	0x00	Section 20.3.12
0x0C	PWM_CMP3H – Comparator 3 Value Register (MSB)	0x00	Section 20.3.13
0x0D	PWM_CMP4L – Comparator 4 Value Register (LSB)	0x00	Section 20.3.14
0x0E	PWM_CMP4H – Comparator 4 Value Register (MSB)	0x00	Section 20.3.15
0x0F	PWM_CMP5L – Comparator 5 Value Register (LSB)	0x00	Section 20.3.16
0x10	PWM_CMP5H – Comparator 5 Value Register (MSB)	0x00	Section 20.3.17
0x11	PWM_CMP6L – Comparator 6 Value Register (LSB)	0x00	Section 20.3.18
0x12	PWM_CMP6H – Comparator 6 Value Register (MSB)	0x00	Section 20.3.19
0x13	PWM_CMP7L – Comparator 7 Value Register (LSB)	0x00	Section 20.3.20
0x14	PWM_CMP7H – Comparator 7 Value Register (MSB)	0x00	Section 20.3.21
0x15	PWM_TOGGLE0 – Channel 0 OUT Toggle Comparator Mask Register	0x00	Section 20.3.22
0x16	PWM_TOGGLE1 – Channel 1 OUT Toggle Comparator Mask Register	0x00	Section 20.3.23
0x17	PWM_TOGGLE2 – Channel 2 OUT Toggle Comparator Mask Register	0x00	Section 20.3.24
0x18	PWM_TOGGLE3 – Channel 3 OUT Toggle Comparator Mask Register	0x00	Section 20.3.25
0x19	PWM_TOGGLE4 – Channel 4 OUT Toggle Comparator Mask Register	0x00	Section 20.3.26
0x1A	PWM_TOGGLE5 – Channel 5 OUT Toggle Comparator Mask Register	0x00	Section 20.3.27
0x1B	PWM_TOGGLE6 – Channel 6 OUT Toggle Comparator Mask Register	0x00	Section 20.3.28
0x1C	PWM_TOGGLE7 – Channel 7 OUT Toggle Comparator Mask Register	0x00	Section 20.3.29
0x1D	PWM_OUT_CLR_EN – PWM OUT Clear Enable Register	0x00	Section 20.3.30
0x1E	PWM_CTRL_BL_CMP8 – Control Block CMP8 Value Register	0x00	Section 20.3.31
0x1F	PWM_INIT – PWM Initialization Register	0x00	Section 20.3.32
0x20	PWM_INTMASK – PWM Interrupt Mask Register	0x00	Section 20.3.33
0x21	PWM_INTSTATUS – PWM Interrupt Status Register	0x00	Section 20.3.34
0x22	PWM_SAMPLE_FREQ_H – PWM Data Sampling Frequency High Byte Register	0x56	Section 20.3.35
0x23	PWM_SAMPLE_FREQ_L – PWM Data Sampling Frequency Low Byte Register	0x22	Section 20.3.36
0x24	PCM_VOLUME – PCM Volume Register	0x00	Section 20.3.37
0x3C	PWM_BUFFER – PCM Buffer Register	0xXXXX	Section 20.3.38

Table 20.1 – Overview of PWM Registers

Copyright © Bridgetek Pte Ltd

20.2 Functional Description

20.2.1 Block Diagram

The FT90X series has 8 separate independent PWM channels. The following section describes the building blocks used to control these PWM channels.

Pre-scaler - This is a programmable counter that reduces the frequency of the system clock (100 MHz) to the desired frequency. The pre-scaler is shared by all 8 PWM channels.

16 Bit Counter Block - This is a programmable counter that determines the period of the PWM signal. The input clock is from the pre-scaler block. The 16-bit counter is shared by all 8 PWM channels.

16 Bit Comparator – Up to 8 comparators can be used per PWM channel. The number of comparators assigned to a PWM channel determines the toggle events (up to 8), which give up to 4 data pulses. Simple duty cycle-based pulse width modulation can be programmed by using only 2 comparators. There are a total of 8 comparators in the PWM module.

Control Block – This controls the number of times to repeat the PWM waveform. The control block is shared by all 8 PWM channels.

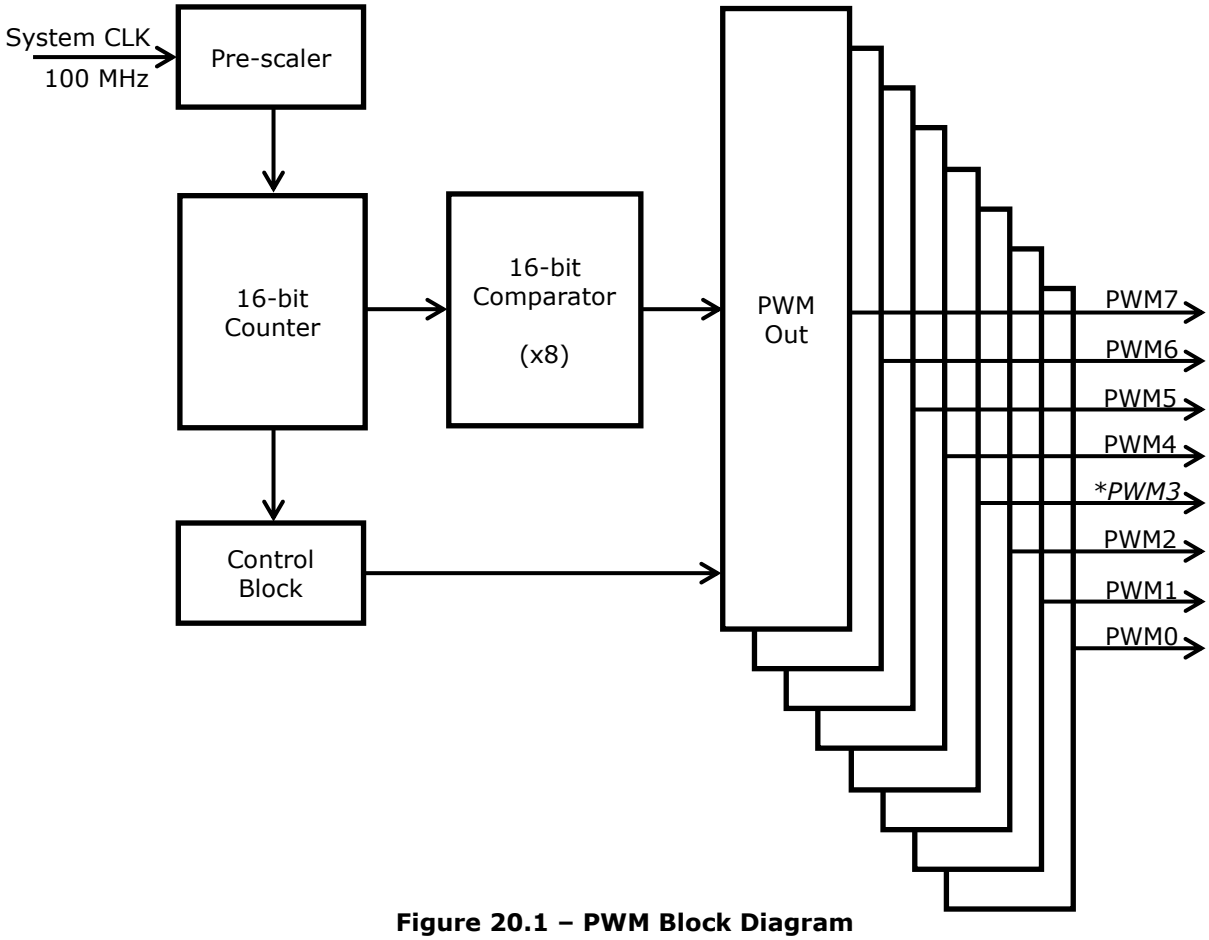


Figure 20.1 – PWM Block Diagram

Note: **PWM3 is not bonded out to a package pin.*

20.2.2 Duty Cycle Generation Example

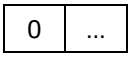
This section describes an example of how to generate a 4 pulse PWM waveform at the PWM0 output, which toggles at the following counter states:

Copyright © Bridgetek Pte Ltd

7, 8, 12, 14

The process is demonstrated in Figure 20.2 where:

represents the PWM clock



represents the PWM counter value at each clock cycle

represents the PWM0 output

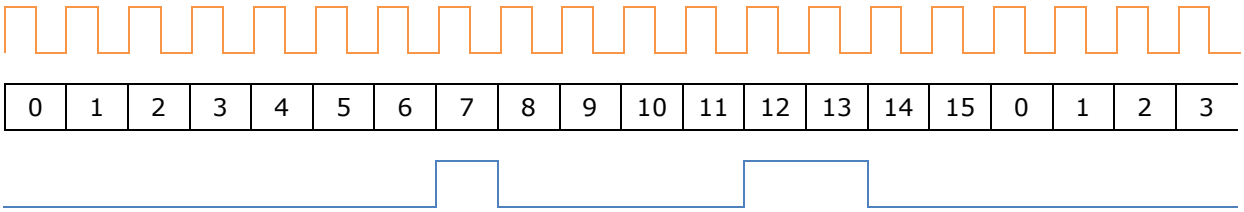


Figure 20.2 – PWM Waveform generated by 4 Comparators

In this example, 4 comparators are used to generate 4 toggle events. The 16-bit counter is programmed to count 16 states and then re-start, and 2 pulses are generated.

Comparator #	Comparator Value	Pulse Width (clock cycles)
0	7	
1	8	1
2	12	
3	14	2

Table 20.2 – Comparators values to generate the example waveform

For a simple duty cycle PWM where only a single pulse is required, only 2 comparators would be necessary and only a single pulse is generated. For example, to generate a 50% duty cycle waveform for the above clock/counter combination, comparators 0 & 1 could be programmed as follows:

Comparator #	Comparator Value	Pulse Width (clock cycles)
0	2	
1	10	8

Table 20.3 – Comparators values to generate 50% duty cycle

With this configuration, there are 16 clocks per cycle and the PWM output changes state (toggles) every 8 clocks (8/16). This produces a 50 % duty cycle. By programming different toggling values into the FT90x PWM comparators, a wide range of duty cycles can be generated.

Copyright © Bridgetek Pte Ltd

20.3 Register Details

20.3.1 PWM_CTRL0 – PCM Control Register (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7	PWM_CH01_AUTO	RW	1'b0	Set to 1 to use channels 0 & 1 for audio playback via the internal FIFO.
6	PWM_CH01_MONO	RW	1'b0	Set to 1 for mono audio. Both channels will playback the same data.
5	PWM_CH01_8BIT	RW	1'b0	Set to 1 for 8-bit audio data.
4	PWM_CH01_SCALE	RW	1'b0	Set to 1 if automatic scaling of data is required.
3	PWM_CH01_FILTER	RW	1'b0	Set to 1 if PCM filter is required. This is valid only with PWM_CH01_AUTO and PCM_EN set.
2	PCM_EN	RW	1'b0	Set to 1 if channel 0 and channel 1 are used for PCM playback.
1	PWM_DEV_EN	RW	1'b0	Set to 1 to enable PWM. This bit is not really used.
0	PWM_SOFT_RESET	RW	1'b0	Set to 1 to reset PWM.

Table 20.4 – PWM_CTRL0 – PCM Control Register

20.3.2 PWM_CTRL1 – PWM Control Register (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6	PCM_BIYTEREVERSE	RW	1'b0	0: Data is treated as in big-endian format For 8-bit, this has no effect. 1: Data is treated as in little-endian format For 8-bit, this has no effect. For 16-bit, the lower and upper bytes are swapped.
5	PWM_INT	RW1C	1'b0	PWM Interrupt Flag This bit is set if a GPIO is configured to trigger the PWM to start with PWM_TRIG_SEL (see MSC0CFG - Miscellaneous Configuration Register). If PWM_INT_MASK is set, an interrupt will be generated to the CPU.
4	PWM_INT_MASK	RW	1'b0	Interrupt mask bit for PWM_INT
2:1	PWM_TRIGGER_EN	RW	2'b00	PWM Trigger Enable for the GPIO selected with PWM_TRIG_SEL in MSC0CFG. 00 – Disabled 01 – Positive Edge 10 – Negative Edge 11 – Any Edge
0	PWM_EN	RW	1'b0	Set to 1 to enable PWM.

Table 20.5 – PWM_CTRL1 – PWM Control Register

Copyright © Bridgetek Pte Ltd

20.3.3 PWM_PRESCALER – PWM Pre-scaler Register (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7:0	PRESCALER	RW	8'h00	8-bit Pre-scaler value. The actual pre-scaler value is (PRESCALER + 1).

Table 20.6 – PWM_PRESCALER – PWM Pre-scaler Register

20.3.4 PWM_CNTL – PWM Counter Register (LSB) (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7:0	CNT16_LSB	RW	8'h00	16-bit counter LSB. The actual counter value is: (CNT16_MSB << 8) + CNT16_LSB + 1

Table 20.7 – PWM_CNTL – PWM Counter Register (LSB)

20.3.5 PWM_CNTH – PWM Counter Register (MSB) (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7:0	CNT16_MSB	RW	8'h00	16-bit counter MSB.

Table 20.8 – PWM_CNTH – PWM Counter Register (MSB)

20.3.6 PWM_CMP0L – Comparator 0 Value Register (LSB) (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7:0	CMP16_0_LSB	RW	8'h00	LSB of comparator 0 16-bit value.

Table 20.9 – PWM_CMP0L – Comparator 0 Value Register (LSB)

20.3.7 PWM_CMP0H – Comparator 0 Value Register (MSB) (address offset: 0x06)

Bit	Name	Type	Default Value	Description
7:0	CMP16_0_MSB	RW	8'h00	MSB of comparator 0 16-bit value.

Table 20.10 – PWM_CMP0H – Comparator 0 Value Register (MSB)

20.3.8 PWM_CMP1L – Comparator 1 Value Register (LSB) (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7:0	CMP16_1_LSB	RW	8'h00	LSB of comparator 1 16-bit value.

Table 20.11 – PWM_CMP1L – Comparator 1 Value Register (LSB)

20.3.9 PWM_CMP1H – Comparator 1 Value Register (MSB) (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:0	CMP16_1_MSB	RW	8'h00	MSB of comparator 1 16-bit value.

Table 20.12 – PWM_CMP1H – Comparator 1 Value Register (MSB)

Copyright © Bridgetek Pte Ltd

20.3.10 PWM_CMP2L – Comparator 2 Value Register (LSB) (address offset: 0x09)

Bit	Name	Type	Default Value	Description
7:0	CMP16_2_LSB	RW	8'h00	LSB of comparator 2 16-bit value.

Table 20.13 – PWM_CMP2L – Comparator 2 Value Register (LSB)

20.3.11 PWM_CMP2H – Comparator 2 Value Register (MSB) (address offset: 0x0A)

Bit	Name	Type	Default Value	Description
7:0	CMP16_2_MSB	RW	8'h00	MSB of comparator 2 16-bit value.

Table 20.14 – PWM_CMP2H – Comparator 2 Value Register (MSB)

20.3.12 PWM_CMP3L – Comparator 3 Value Register (LSB) (address offset: 0x0B)

Bit	Name	Type	Default Value	Description
7:0	CMP16_3_LSB	RW	8'h00	LSB of comparator 3 16-bit value.

Table 20.15 – PWM_CMP3L – Comparator 3 Value Register (LSB)

20.3.13 PWM_CMP3H – Comparator 3 Value Register (MSB) (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:0	CMP16_3_MSB	RW	8'h00	MSB of comparator 3 16-bit value.

Table 20.16 – PWM_CMP3H – Comparator 3 Value Register (MSB)

20.3.14 PWM_CMP4L – Comparator 4 Value Register (LSB) (address offset: 0x0D)

Bit	Name	Type	Default Value	Description
7:0	CMP16_4_LSB	RW	8'h00	LSB of comparator 4 16-bit value.

Table 20.17 – PWM_CMP4L – Comparator 4 Value Register (LSB)

20.3.15 PWM_CMP4H – Comparator 4 Value Register (MSB) (address offset: 0x0E)

Bit	Name	Type	Default Value	Description
7:0	CMP16_4_MSB	RW	8'h00	MSB of comparator 4 16-bit value.

Table 20.18 – PWM_CMP4H – Comparator 4 Value Register (MSB)

20.3.16 PWM_CMP5L – Comparator 5 Value Register (LSB) (address offset: 0x0F)

Bit	Name	Type	Default Value	Description
7:0	CMP16_5_LSB	RW	8'h00	LSB of comparator 5 16-bit value.

Table 20.19 – PWM_CMP5L – Comparator 5 Value Register (LSB)

Copyright © Bridgetek Pte Ltd

20.3.17 PWM_CMP5H – Comparator 5 Value Register (MSB) (address offset: 0x10)

Bit	Name	Type	Default Value	Description
7:0	CMP16_5_MSB	RW	8'h00	MSB of comparator 5 16-bit value.

Table 20.20 – PWM_CMP5H – Comparator 5 Value Register (MSB)

20.3.18 PWM_CMP6L – Comparator 6 Value Register (LSB) (address offset: 0x11)

Bit	Name	Type	Default Value	Description
7:0	CMP16_6_LSB	RW	8'h00	LSB of comparator 6 16-bit value.

Table 20.21 – PWM_CMP6L – Comparator 6 Value Register (LSB)

20.3.19 PWM_CMP6H – Comparator 6 Value Register (MSB) (address offset: 0x12)

Bit	Name	Type	Default Value	Description
7:0	CMP16_6_MSB	RW	8'h00	MSB of comparator 6 16-bit value.

Table 20.22 – PWM_CMP6H – Comparator 6 Value Register (MSB)

20.3.20 PWM_CMP7L – Comparator 7 Value Register (LSB) (address offset: 0x13)

Bit	Name	Type	Default Value	Description
7:0	CMP16_7_LSB	RW	8'h00	LSB of comparator 7 16-bit value.

Table 20.23 – PWM_CMP7L – Comparator 7 Value Register (LSB)

20.3.21 PWM_CMP7H – Comparator 7 Value Register (MSB) (address offset: 0x14)

Bit	Name	Type	Default Value	Description
7:0	CMP16_7_MSB	RW	8'h00	MSB of comparator 7 16-bit value.

Table 20.24 – PWM_CMP7H – Comparator 7 Value Register (MSB)

20.3.22 PWM_TOGGLE0 – Channel 0 OUT Toggle Comparator Mask Register (address offset: 0x15)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_0	RW	8'h00	Channel 0 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.25 – PWM_TOGGLE0 – Channel 0 OUT Toggle Comparator Mask

Copyright © Bridgetek Pte Ltd

20.3.23 PWM_TOGGLE1 – Channel 1 OUT Toggle Comparator Mask Register (address offset: 0x16)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_1	RW	8'h00	Channel 1 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.26 – PWM_TOGGLE1 – Channel 1 OUT Toggle Comparator Mask

20.3.24 PWM_TOGGLE2 – Channel 2 OUT Toggle Comparator Mask Register (address offset: 0x17)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_2	RW	8'h00	Channel 2 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.27 – PWM_TOGGLE2 – Channel 2 OUT Toggle Comparator Mask Register

20.3.25 PWM_TOGGLE3 – Channel 3 OUT Toggle Comparator Mask Register (address offset: 0x18)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_3	RW	8'h00	Channel 3 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.28 – PWM_TOGGLE3 – Channel 3 OUT Toggle Comparator Mask Register

20.3.26 PWM_TOGGLE4 – Channel 4 OUT Toggle Comparator Mask Register (address offset: 0x19)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_4	RW	8'h00	Channel 4 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.29 – PWM_TOGGLE4 – Channel 4 OUT Toggle Comparator Mask Register

20.3.27 PWM_TOGGLE5 – Channel 5 OUT Toggle Comparator Mask Register (address offset: 0x1A)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_5	RW	8'h00	Channel 5 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.30 – PWM_TOGGLE5 – Channel 5 OUT Toggle Comparator Mask Register

Copyright © Bridgetek Pte Ltd

20.3.28 PWM_TOGGLE6 – Channel 6 OUT Toggle Comparator Mask Register (address offset: 0x1B)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_6	RW	8'h00	Channel 6 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.31 – PWM_TOGGLE6 – Channel 6 OUT Toggle Comparator Mask Register

20.3.29 PWM_TOGGLE7 – Channel 7 OUT Toggle Comparator Mask Register (address offset: 0x1C)

Bit	Name	Type	Default Value	Description
7:0	TOGGLE_EN_7	RW	8'h00	Channel 7 PWM OUT toggle comparator mask (each bit corresponds to 1 comparator)

Table 20.32 – PWM_TOGGLE7 – Channel 7 OUT Toggle Comparator Mask Register

20.3.30 PWM_OUT_CLR_EN – PWM OUT Clear Enable Register (address offset: 0x1D)

Bit	Name	Type	Default Value	Description
7:0	CLR_EN	RW	8'h00	PWM Out clear enable

Table 20.33 – PWM_OUT_CLR_EN – PWM OUT Clear Enable Register

20.3.31 PWM_CTRL_BL_CMP8 – Control Block CMP8 Value Register (address offset: 0x1E)

Bit	Name	Type	Default Value	Description
7:0	CTRL_BL_CMP8	RW	8'h00	Control block CMP8 value. 0: continuous 1: one-shot, 2 – 255

Table 20.34 – PWM_CTRL_BL_CMP8 – Control Block CMP8 Value Register

20.3.32 PWM_INIT – PWM Initialization Register (address offset: 0x1F)

Bit	Name	Type	Default Value	Description
7:0	INIT	RW	8'h00	PWM Initialisation register

Table 20.35 – PWM_INIT – PWM Initialization Register

20.3.33 PWM_INTMASK – PWM Interrupt Mask Register (address offset: 0x20)

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5	FIFO_EMPTY_MASK	RW	1'b0	FIFO empty interrupt mask
4	FIFO_FULL_MASK	RW	1'b0	FIFO full interrupt mask
3	FIFO_HALF_MASK	RW	1'b0	FIFO half full interrupt mask
2	FIFO_OV_MASK	RW	1'b0	FIFO overflow interrupt mask
1	FIFO_UNDER_MASK	RW	1'b0	FIFO underflow interrupt mask
0	PWM_INT_MASK	RW	1'b0	PWM_INT interrupt mask (same as bit[4] of PWM_CTRL)

Table 20.36 – PWM_INTMASK – PWM Interrupt Mask Register

Copyright © Bridgetek Pte Ltd

20.3.34 PWM_INTSTATUS – PWM Interrupt Status Register (address offset: 0x21)

Bit	Name	Type	Default Value	Description
7:6	Reserved	-	-	-
5	FIFO_EMPTY_INT	RW1C	1'b0	FIFO empty interrupt; write 1 to clear.
4	FIFO_FULL_INT	RW1C	1'b0	FIFO full interrupts; write 1 to clear.
3	FIFO_HALF_INT	RW1C	1'b0	FIFO half full interrupts; write 1 to clear.
2	FIFO_OV_INT	RW1C	1'b0	FIFO overflow interrupt; write 1 to clear.
1	FIFO_UNDER_INT	RW1C	1'b0	FIFO underflow interrupts; write 1 to clear.
0	PWM_INT	RW1C	1'b0	PWM interrupt; write 1 to clear. (same as bit[5] of PWM_CTRL1)

Table 20.37 – PWM_INTSTATUS – PWM Interrupt Status Register

20.3.35 PWM_SAMPLE_FREQ_H – PWM Data Sampling Frequency High Byte Register (address offset: 0x22)

Bit	Name	Type	Default Value	Description
7:0	PWM_SAMPLE_FREQ_H	RW	8'h56	PWM Data Sampling Frequency High Byte

Table 20.38 – PWM_SAMPLE_FREQ_H – PWM Data Sampling Frequency High Byte Register

20.3.36 PWM_SAMPLE_FREQ_L – PWM Data Sampling Frequency Low Byte Register (address offset: 0x23)

Bit	Name	Type	Default Value	Description
7:0	PWM_SAMPLE_FREQ_L	RW	8'h22	PWM Data Sampling Frequency Low Byte

Table 20.39 – PWM_SAMPLE_FREQ_L – PWM Data Sampling Frequency Low Byte Register

Copyright © Bridgetek Pte Ltd

20.3.37 PCM_VOLUME – PCM Volume Register (address offset: 0x24)

Bit	Name	Type	Default Value	Description	
7:5	Reserved	-	-	-	
4:0	VOLUME	RW	5'h00	0x00	Mute
				0x01	~6.25%
				0x02	~12.5%
				0x03	~19%
				0x04	~25%
				0x05	~31%
				0x06	~37%
				0x07	~44%
				0x08	~50%
				0x09	~56%
				0x0A	~63%
				0x0B	~69%
				0x0C	~75%
				0x0D	~81%
				0x0E	~88%
				0x0F	~94%
				0x10	~100%
				0x11-0x1F	Illegal; forced to 0x10

Table 20.40 – PCM_VOLUME – PCM Volume Register

20.3.38 PWM_BUFFER – PCM Buffer Register (address offset: 0x3C)

Bit	Name	Type	Default Value	Description
15:0	DATA	WO	-	The entry point to the FIFO. It must be written in 16-bit. For 8-bit data, the upper 8-bit will be ignored.

Table 20.41 – PWM_BUFFER – PCM Buffer Register

Copyright © Bridgetek Pte Ltd

21 Data Capture Interface

There is an 8-bit parallel interface to collect byte streaming data from a sensor peripheral – e.g., a camera module – in a 2Kbyte internal FIFO. The interface will provide a clock to the peripheral at a speed of 25MHz (max).

21.1 Register Summary

Listed below are the registers with their offset from the base address (0x10360). All registers can only be accessed via Double-Word (32-bit) mode.

Address Offset	Register	Default Value	References
0x00	DCAP_REG1 – Data Capture Interface Register 1	0x00000FFC	Section 21.2.1
0x04	DCAP_REG2 – Data Capture Interface Register 2	0x00000000	Section 21.2.2
0x08	DCAP_REG3 – Data Capture Interface Register 3	0x00000000	Section 21.2.3
0x0C	DCAP_REG4 – Data Capture Interface Register 4	0x00000000	Section 21.2.4

Table 21.1 – Overview of Data Capture Interface Registers

21.2 Register Details

21.2.1 DCAP_REG1 – Data Capture Interface Register 1 (address offset: 0x00)

Bit	Name	Type	Default Value	Description
31:16	COUNT	RW	16'h0000	Number of bytes to capture after the last trigger event. This must be in multiple of 4 bytes
15:12	TRIG_PAT	RW	4'h0	This specifies the data capture trigger position. Data capture starts on the cycle when trigger transitions from 0 to 1. **En[1] / VD** / **En[0] / HD** / **Trigger** 0 / 0 / TRIG_PAT[0] 1 / 1 / TRIG_PAT[1] 1 / 0 / TRIG_PAT[2] 1 / 1 / TRIG_PAT[3]
11:0	THRESHOLD	RW	12'hFFC	Specifies the threshold value for the HAS_DATA signal. It must be a multiple of 4 bytes.

Table 21.2 – DCAP_REG1 – Data Capture Interface Register 1

21.2.2 DCAP_REG2 – Data Capture Interface Register 2 (address offset: 0x04)

Bit	Name	Type	Default Value	Description
31:12	Reserved	-	-	-
11:0	FULLNESS	RO	12''h000	Specifies the number of bytes that can be safely read from the FIFO. Its value is always a multiple of 4.

Table 21.3 – DCAP_REG2 – Data Capture Interface Register 2

Copyright © Bridgetek Pte Ltd

21.2.3 DCAP_REG3 – Data Capture Interface Register 3 (address offset: 0x08)

Bit	Name	Type	Default Value	Description
31:0	DATA	RO	-	Next four samples from the FIFO. Reading this will remove the sample from the FIFO. When FULLNESS is 0, the effect is non-deterministic

Table 21.4 – DCAP_REG3 – Data Capture Interface Register 3

21.2.4 DCAP_REG4 – Data Capture Interface Register 4 (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
31:3	Reserved	-	-	-
2	HAS_DATA	RO	1'b0	Set when FIFO contains at least THRESHOLD bytes.
1	CLK_SENSE	RW	1'b0	1: capture data on clock rising edge
0	INT_ENB	RW	1'b0	1: enable interrupt; an interrupt will be generated when HAS_DATA = 1

Table 21.5 – DCAP_REG4 – Data Capture Interface Register 4

Copyright © Bridgetek Pte Ltd

22 Flash Controller

There are two ways to access the memory control unit. One is via the CPU I/O interface and the other via the one-wire debugger interface. The CPU I/O interface is described here.

From the CPU I/O interface, memory transfer can occur between the Flash and the Program Memory, and between the Flash and the Data Memory. A number of the serial Flash commands are also supported, and the CPU may issue these commands to the Flash.

As this is a shared resource between the debugger interface and CPU I/O interface, the interface must acquire the resource first before performing any of the activities. This is done by reading a test-and-set semaphore. If the interface reads a 0 from the semaphore it can safely assume it has acquired the resource. Any further read to this semaphore will return a 1 until it is released by the interface that acquired the resource.

If both the debugger and CPU attempt to acquire the resource while it is free at exactly the same time, priority is given to the debugger and the CPU interface will read a 1 instead.

Until the semaphore is acquired, the semaphore is the only register any of the interfaces can read. Once the semaphore is acquired by the interface, the interface will have full access to all the registers in the control unit.

If an interface that does not have the resource writes to the control unit registers, they will be ignored while a read from any of the control unit registers will always return 0.

22.1 Register Summary

Listed below are the registers with their offset from the base address (0x10800). All registers can only be accessed via Byte (8-bit) mode.

Address Offset	Register	Default Value	References
0x00	RSADDR0 – Memory Start Address Register (LSB)	0x00	Section 22.2.1
0x01	RSADDR1 – Memory Start Address Register (Byte 1)	0x00	Section 22.2.2
0x02	RSADDR2 – Memory Start Address Register (MSB)	0x00	Section 22.2.3
0x03	FSADDR0 – Flash Start Address Register (LSB)	0x00	Section 22.2.4
0x04	FSADDR1 – Flash Start Address Register (Byte 1)	0x00	Section 22.2.5
0x05	FSADDR2 – Flash Start Address Register (MSB)	0x00	Section 22.2.6
0x06	BLENGTH0 – Data Byte Length Register (LSB)	0x00	Section 22.2.7
0x07	BLENGTH1 – Data Byte Length Register (Byte 1)	0x00	Section 22.2.8
0x08	BLENGTH2 – Data Byte Length Register (MSB)	0x00	Section 22.2.9
0x09	COMMAND – Command Register	0x00	Section 22.2.10
0x0A	Reserved	-	
0x0B	SEMAPHORE – Semaphore Register	0x00	Section 22.2.11
0x0C	CONFIG – Configuration Register	0x00	Section 22.2.12
0x0D	STATUS – Status Register	0x00	Section 22.2.13

Copyright © Bridgetek Pte Ltd

0x0E	CRCL – Flash Content CRC Register (LSB)	0xXX	Section 22.2.14
0x0F	CRCH – Flash Content CRC Register (MSB)	0xXX	Section 22.2.15
0x7C	CHIPID0 – Chip ID Register (LSB) (only via one-wire debugger)	0xXX	Section 22.2.16
0x7D	CHIPID1 – Chip ID Register (Byte 1) (only via one-wire debugger)	0xXX	Section 22.2.17
0x7E	CHIPID2 – Chip ID Register (Byte 2) (only via one-wire debugger)	0xXX	Section 22.2.18
0x7F	CHIPID3 – Chip ID Register (MSB) (only via one-wire debugger)	0xXX	Section 22.2.19
0x80	DRWDATA – Data Register	0x00	Section 22.2.20

Table 22.1 – Overview of Data Capture Interface Registers

22.2 Register Details

22.2.1 RSADDR0 – Memory Start Address Register (LSB) (address offset: 0x00)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	LSB of the start address of memory location to perform read/write for either program or data memory.

Table 22.2 – RSADDR1 – Memory Start Address Register (LSB)

22.2.2 RSADDR1 – Memory Start Address Register (Byte 1) (address offset: 0x01)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	Byte 1 of the start address of memory location to perform read/write for either program or data memory.

Table 22.3 – RSADDR1 – Memory Start Address Register (Byte 1)

22.2.3 RSADDR2 – Memory Start Address Register (MSB) (address offset: 0x02)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	MSB of the start address of memory location to perform read/write for either program or data memory

Table 22.4 – RSADDR2 – Memory Start Address Register (MSB)

Note: *The memory start address must always be aligned to a 4-byte boundary in both read/write cases. The address is treated as a double word address (e.g., 01 is byte address 4).*

Copyright © Bridgetek Pte Ltd

22.2.4 FSADDR0 – Flash Start Address Register (LSB) (address offset: 0x03)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	LSB of the start address of flash location to perform read/write

Table 22.5 – FSADDR0 – Flash Start Address Register (LSB)

22.2.5 FSADDR1 – Flash Start Address Register (Byte 1) (address offset: 0x04)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	Byte 1 of the start address of flash location to perform read/write

Table 22.6 – FSADDR1 – Flash Start Address Register (Byte 1)

22.2.6 FSADDR2 – Flash Start Address Register (MSB) (address offset: 0x05)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	MSB of the start address of flash location to perform read/write

Table 22.7 – FSADDR2 – Flash Start Address Register (MSB)

Note: *The flash start address must always be aligned to 256-byte boundary in the write case, and can be any value in the read case. The address is treated as a byte address (e.g., 01 is byte address 1).*

22.2.7 BLENGTH0 – Data Byte Length Register (LSB) (address offset: 0x06)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	LSB of the Byte Length to transfer

Table 22.8 – BLENGTH0 – Data Byte Length Register (LSB) (LSB)

22.2.8 BLENGTH1 – Data Byte Length Register (Byte 1) (address offset: 0x07)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	Byte 1 of the Byte Length to transfer

Table 22.9 – BLENGTH1 – Data Byte Length Register (Byte 1)

22.2.9 BLENGTH2 – Data Byte Length Register (MSB) (address offset: 0x08)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	MSB of the Byte Length to transfer

Table 22.10 – BLENGTH2 – Data Byte Length Register (MSB)

Note 1: *When the flash is the destination (write case), the byte length must be a multiple of 256 bytes (1 page of flash entry). There is no such restriction on byte length if the flash is the source (read case).*

Note 2: *The registers must be set to (byte length – 1). For example, if the byte length is 256, then BLENGTH0 is set to 255 and BLENGTH1 and BLENGTH2 are set to 0.*

Copyright © Bridgetek Pte Ltd

22.2.10 COMMAND – Command Register (address offset: 0x09)

Bit	Name	Type	Default Value	Description
7:0	-	WO	0x00	Command to perform. See section 23.3.

Table 22.11 – COMMAND – Command Register

22.2.11 SEMAPHORE – Semaphore Register (address offset: 0x0B)

Bit	Name	Type	Default Value	Description
7:0	-	RW	0x00	Semaphore value If a 0 is read, then the control unit's resource is allocated to whichever that does the test. This register is then automatically set to 1. If a 1 is read, then the resource is being used and not free. This semaphore can be released by only the interface which acquires the semaphore with writing a 1 to this register. All other registers in this control unit are not readable or writeable until this semaphore is acquired. If both the CPU and Debugger test this semaphore register at the same time when it is free, priority is given to the Debugger and the CPU will read a 1 instead.

Table 22.12 – SEMAPHORE – Semaphore Register

22.2.12 CONFIG – Configuration Register (address offset: 0x0C)

Bit	Name	Type	Default Value	Description
7:2	Reserved	-	-	Reserved bits
1:0	SPI_CLK	WO	2'b0	The serial SPI clock speed to the serial flash 0x0: Flash SPI clock speed = ½ System clock speed 0x1: Flash SPI clock speed = 1/3 System clock speed 0x2: Flash SPI clock speed = ¼ System clock speed 0x3: Flash SPI clock speed = 1/5 System clock speed

Table 22.13 – CONFIG – Configuration Register

22.2.13 STATUS – Status Register (address offset: 0x0D)

Bit	Name	Type	Default Value	Description
7	Reserved	-	-	-
6	Control Busy	RO	1'b0	1: The control unit is busy. This means no other command should be issued. When this bit is set, bits 7, 3-0 may not be valid until this bit is cleared.
5	Data Read Ready	RO	1'b0	1: Data for read is available at data read port
4	Data Write Ready	RO	1'b0	1: Data for write can be written to data write port
3:1	Reserved	-	-	-
0	Write in Progress	RO	1'b0	1: Flash Write Operations in Progress

Table 22.14 – STATUS – Status Register

Copyright © Bridgetek Pte Ltd

22.2.14 CRCL – Flash Content CRC Register (LSB) (address offset: 0x0E)

Bit	Name	Type	Default Value	Description
7:0	-	RO	0xXX	LSB of the CRC16 of the flash content. The CRC is calculated upon reset when flash content is transferred to the programme memory, and the CRC is based on the polynomial X16+X15+X2+1

Table 22.15 – CRCL – Flash Content CRC Register (LSB)

22.2.15 CRCH – Flash Content CRC Register (MSB) (address offset: 0x0F)

Bit	Name	Type	Default Value	Description
7:0	-	RO	0xXX	MSB of the CRC16 of the flash content. The CRC is calculated upon reset when flash content is transferred to the programme memory, and the CRC is based on the polynomial X16+X15+X2+1

Table 22.16 – CRCH – Flash Content CRC Register (MSB)

22.2.16 CHIPID0 – Chip ID Register (LSB) (address offset: 0x7C)

Bit	Name	Type	Default Value	Description
7:0	-	RO	0xXX	LSB of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address.

Table 22.17 – CHIPID0 – Chip ID Register (LSB)

22.2.17 CHIPID1 – Chip ID Register (Byte 1) (address offset: 0x7D)

Bit	Name	Type	Default Value	Description
7:0	-	RO	0xXX	Byte 1 of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address.

Table 22.18 – CHIPID1 – Chip ID Register (Byte 1)

22.2.18 CHIPID2 – Chip ID Register (Byte 2) (address offset: 0x7E)

Bit	Name	Type	Default Value	Description
7:0	-	RO	0xXX	Byte 2 of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address.

Table 22.19 – CHIPID2 – Chip ID Register (Byte 2)

22.2.19 CHIPID3 – Chip ID Register (MSB) (address offset: 0x7F)

Bit	Name	Type	Default Value	Description
7:0	-	RO	0xXX	MSB of the 32-bit chip ID, only accessible via the one-wire debugger. The same ID is available to the CPU in another register address.

Table 22.20 – CHIPID3 – Chip ID Register (MSB)

Copyright © Bridgetek Pte Ltd

22.2.20 DRWDATA – Data Register (address offset: 0x80)

Bit	Name	Type	Default Value	Description
7:0	-	RW	0x00	This is the data read or write port used to transfer data in and out of this control unit. Up to 64 addresses may be used. When bit 4 of STATUS is not set, writing to this port will have no effect. When bit 5 of STATUS is not set, reading this port will return 0.

Table 22.21 – DRWDATA – Data Register

22.3 Flash Controller Commands

The following commands are supported by the control unit. The command is initiated once the command register is updated. No further command can be entered until the current one is completed. If two successive commands, regardless what they are, are entered while one is on-going, the current operation will be aborted. The two commands will not be executed either.

The commands can be divided into 2 groups. The first group consists of commands that are supported directly by the serial flash. The second group consists of commands that initiate data transfer between the debugger/CPU interface and flash, debugger/CPU interface and program/data memory, or flash and program/data memory.

The following table lists the first group of commands. If executing any of these commands that require data input, write the required data to the DRWDATA register.

Command	Code	Description
CMDWREN	0x06	The Write Enable (WREN) instruction is for setting the Write Enable Latch (WEL) bit. Those instructions such as PP, SE, BE, CE, and WRSR, which are intended to change the device content, should be set every time after the WREN instruction sets the WEL bit.
CMDWRDI	0x04	The Write Disable (WRDI) instruction is for resetting the Write Enable Latch (WEL) bit. The WEL bit is reset by the following situations: - Power-up - Write Disable (WRDI) instruction completion - Write Status Register (WRSR) instruction completion - Page Program (PP) instruction completion - Sector Erase (SE) instruction completion - Block Erase (BE) instruction completion - Chip Erase (CE) instruction completion
CMDWRSR	0x01	The WRSR instruction is for changing the values of Status Register Bits. Before sending WRSR instruction, the Write Enable (WREN) instruction must be executed first. The WRSR instruction can change the value of Block Protect (BP1 – bit3, BP0 – bit2) bits to define the protected area of memory (as shown in table 2). The WRSR also can set or reset the Status Register Write Disable (SRWD – bit 7) bit in accordance with Write Protection (WP#) pin signal. The WRSR instruction cannot be executed once the Hardware Protected Mode (HPM) is entered. This instruction has no effect on bits 6, 5, 1 and 0.

Copyright © Bridgetek Pte Ltd

Command	Code	Description
CMDRDID	0x9F	The RDID instruction is for reading the manufacturer ID of 1-byte and followed by Device ID of 2-byte. The MXIC Manufacturer ID and Device ID are listed as table of "ID Definitions" of the MXIC specification. While Program/Erase operation is in progress, it will not decode the RDID instruction, so there's no effect on the cycle of program/erase operation which is currently in progress.
CMDRDSR	0x05	The RDSR instruction is for reading Status Register Bits. The Read Status Register can be read at any time (even in program/erase/write status register condition) and continuously. It is recommended to check the Write in Progress (WIP) bit before sending a new instruction when a program, erase, or write status register operation is in progress. The status data can be read back from the STATUS register. The bits 7, 3-0 of STATUS reflect the content from this flash command.
CMDSE	0x20	Sectors erase; Sector address can be set at FSADDRx.
CMDBE1 / CMDBE2	0x52 / 0xD8	Block erase; Block address can be set at FSADDRx.
CMDCE1 / CMDCE2	0x60 / 0xC7	Chip erase.
CMDDP	0xB9	The Deep Power-down (DP) instruction is for setting the device on the minimizing the power consumption. During the Deep Power-down mode, the device is not active and all Write/Program/Erase instruction are ignored.
CMDRDP	0xAB	The RDP instruction is for releasing from Deep Power Down Mode.

Table 22.22 – Flash Controller Command Group 1

The following table lists the commands that initiate data transfer between Debugger/CPU interface and flash/program/data memory, or between flash and program/data. Each command allows the option to reset the CPU, or reboot the system, or allow the system to continue.

Command	Code	Description
CMDDBG2P1	0xE0	Initiates data transfer from Debugger Interface to Program Memory. The start address of the Program Memory destination will be dictated by RSADDRx which must be 32-bit aligned. The number of bytes to transfer will be dictated by BLENGTHx. Data to be transferred will be in DRWDATA.
CMDDBG2P2	0xE1	Similar to CMDDBG2P1 except at the end of the transfer, a CPU reset will be performed.
CMDDBG2P3	0xE2	Similar to CMDDBG2P1 except at the end of the transfer, a system reboot will be performed.
CMDDBG2D1	0xE4	Similar to CMDDBG2P1 except the destination is the Data Memory.
CMDDBG2D2	0xE5	Similar to CMDDBG2P2 except the destination is the Data Memory.
CMDDBG2D3	0xE6	Similar to CMDDBG2P3 except the destination is the Data Memory.
CMDDBG2F1	0xE8	Initiates data transfer from Debugger Interface to Flash Memory.

Copyright © Bridgetek Pte Ltd

Command	Code	Description
		The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned. The number of bytes to transfer will be dictated by BLENGTHx and must be multiples of 256 bytes. Data to be transferred will be in DRWDATA.
CMDDBG2F2	0xE9	Similar to CMDDBG2F1 except at the end of the transfer, a CPU reset will be performed.
CMDDBG2F3	0xEA	Similar to CMDDBG2F1 except at the end of the transfer, a system reboot will be performed.
CMDP2F1	0xF0	Initiates data transfer from Program Memory to Flash Memory. The start address of the Program Memory destination will be dictated by RSADDRx which must be 32-bit aligned. The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned. The number of bytes to transfer will be dictated by BLENGTHx and must be multiples of 256 bytes.
CMDP2F2	0xF1	Similar to CMDP2F1 except at the end of the transfer, a CPU reset will be performed.
CMDP2F3	0xF2	Similar to CMDP2F1 except at the end of the transfer, a system reboot will be performed.
CMDF2P1	0xF4	Initiates data transfer from Flash Memory to Program Memory. The start address of the Program Memory destination will be dictated by RSADDRx which must be 32-bit aligned. The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned. The number of bytes to transfer will be dictated by BLENGTHx.
CMDF2P2	0xF5	Similar to CMDF2P1 except at the end of the transfer, a CPU reset will be performed.
CMDF2P3	0xF6	Similar to CMDF2P1 except at the end of the transfer, a system reboot will be performed.
CMDD2F1	0xF8	Initiates data transfer from Data Memory to Flash Memory. The start address of the Data Memory destination will be dictated by RSADDRx which must be 32-bit aligned. The start address of the Flash Memory destination will be dictated by FSADDRx which must be 256-byte aligned. The number of bytes to transfer will be dictated by BLENGTHx and must be multiples of 256 bytes.
CMDD2F2	0xF9	Similar to CMDD2F1 except at the end of the transfer, a CPU reset will be performed.
CMDD2F3	0xFA	Similar to CMDD2F1 except at the end of the transfer, a system reboot will be performed.
CMDF2D1	0xFC	Initiates data transfer from Flash Memory to Data Memory. The start address of the Data Memory destination will be dictated by RSADDRx which must be 32-bit aligned. The start address of the Flash Memory destination

Copyright © Bridgetek Pte Ltd

Command	Code	Description
		will be dictated by FSADDRx which must be 256-byte aligned. The number of bytes to transfer will be dictated by BLENGTHx.
CMDF2D2	0xFD	Similar to CMDF2D1 except at the end of the transfer, a CPU reset will be performed.
CMDF2D3	0xFE	Similar to CMDF2D1 except at the end of the transfer, a system reboot will be performed.
CMDHALT	0xFF	This command forces halt immediately to the MCU. The use should be avoided as this cannot be undone. Only a system reset will remove the halt effect.

Table 22.23 – Flash Controller Command Group 2

Copyright © Bridgetek Pte Ltd

23 Analog Interfaces

23.1 ADC

This is the interface used to access the 8-bit/10-bit ADC functionality.

The ADC supports up to 8 multiplexed channels. The default channel is 0. This channel is grounded. The channel can be selected via the **ADC_CHANNEL** bits in the **ADC/DAC Configuration/Status Register**.

Upon reset, the ADC is powered down. This is controlled by **ADC_PDB** in the **ADC/DAC Configuration/Status Register**. Powering down the ADC also resets the ADC control to its default state.

Two modes of operations are supported by the interface, the default one-shot and continuous. This is configurable by setting **ADC_CONT** in the **ADC/DAC Configuration/Status Register**.

In the one-shot mode, data can be read from **ADC_DATA** in the **ADC/DAC Data Register.**

For the continuous mode, there is a 128-level FIFO to hold the incoming conversion data. When the FIFO is full, incoming data will be lost and a buffer error status will be issued. Data can be read off the data port **ADC_DATA** in the **ADC/DAC Data Register**. When the FIFO is empty, further read from **ADC_DATA** will return the last data, and a buffer error status will also be issued. The error is reflected in **ADC_BUF_ERROR** in the **ADC/DAC Configuration/Status Register**. It's sticky and can only be cleared by **ADC_PDB** or reset and has no meaning in the one-shot mode.

ADC_DATA_COUNT in the **ADC/DAC Configuration/Status Register** shows the number of valid samples in the FIFO at the most recent interrupt.

ADC_DATA, **ADC_BUF_ERROR**, **ADC_DATA_COUNT** is read-only.

The ADC conversion operation is started by setting **ADC_START** in the **ADC/DAC Configuration/Status Register**. This should only be triggered after the ADC modes, channel selection etc. have been configured properly. Once triggered, in the continuous mode, the FIFO pointers are all reset.

ADC_START will be automatically cleared in the one-shot mode after the conversion is completed. In the continuous mode, it must be cleared in order to stop the conversion process.

An interrupt will be generated if it is enabled and one of the following conditions occurs:

- A conversion has been completed in the one-shot mode.
- 64 conversions have been performed in the continuous mode.
- The FIFO is full in the continuous mode.
- The last conversion after **ADC_START** is cleared in continuous mode has been completed.

ADC_IRQ_EN in the **ADC/DAC Configuration/Status register** is used to enable the interrupt. **ADC_IRQ_PEND** in the same register denotes there is an ADC interrupt. This bit can be polled to check for ADC interrupt as well. To clear this pending bit, a 1 should be written to the **ADC_IRQ_PEND** bit. Writing a 0 to this bit has no effect. **ADC_IRQ_PEND** does not indicate the type of interrupt; **ADC_DATA_COUNT** should be used to determine the appropriate action.

23.2 DAC

This is the interface used to access the 10-bit DAC functionality.

FT90X supports 2 DAC channels, denoted as DAC0 and DAC1 respectively.

Upon reset, both DACs are powered down. This is controlled by **DACx_PDB** in the **ADC/DAC Configuration/Status Register**. Powering down a DAC also resets its corresponding DAC control to its default state.

Two modes of operations are supported by the interface, the default one-shot and continuous. This is configurable by setting **DACx_CONT** in the **ADC/DAC Configuration/Status Register**.

Copyright © Bridgetek Pte Ltd

In the one-shot mode, data can be written to **DACx_DATA** in the **ADC/DAC Data Register.**

For the continuous mode, there is a 128-level FIFO to hold the out-going data. When the FIFO is full, further data written will be lost and a buffer error status will be issued. Data can be written through the data port **DACx_DATA** in the **ADC/DAC Data Register**. When the FIFO is empty, the last data will be sent to the corresponding DAC for conversion. The error is reflected in **DACx_BUFFER_ERROR** in the **ADC/DAC Configuration/Status register**. It's sticky and can only be cleared by the reset or peripheral soft reset and has no meaning in the one-shot mode.

DACx_DATA_COUNT in the **ADC/DAC Count Register** shows the number of valid samples in the FIFO at any time.

DACx_DATA, **DACx_BUFFER_ERROR**, **DACx_DATA_COUNT** is read-only.

The DAC conversion operation is started by setting **DACx_START** in the **ADC/DAC Configuration/Status Register**. This should only be triggered after the DAC settings such as the conversion mode have been configured properly.

DACx_START will be automatically cleared in the one-shot mode after the conversion is completed. In the continuous mode, it must be cleared in order to stop the conversion process.

An interrupt will be generated if it is enabled and one of the following conditions occurs:

- A conversion has been completed in the one-shot mode.
- 64 conversions have been performed in the continuous mode.
- The FIFO is empty in the continuous mode.
- The last conversion after **DACx_START** is cleared in continuous mode has been completed.

DACx_IRQ_EN in the **ADC/DAC Configuration/Status register** dictate this. **DACx_IRQ_PEND** in the same register denote there is a DAC interrupt. This bit can be polled to check for DAC interrupt as well. To clear this pending bit, a 1 should be written to the corresponding **DACx_IRQ_PEND** bit. Writing a 0 to this bit has no effect. **DACx_IRQ_PEND** do not indicate the type of interrupt; **DACx_DATA_COUNT** should be used to determine the appropriate action.

The DAC conversion rate default is set to the maximum, 1 MS/s. **DAC_DIVIDER** in the **ADC/DAC Count Register** controls this and it affects both DAC0 and DAC1.

Copyright © Bridgetek Pte Ltd

24 Security via OTP

The FT90X flash and program memory can be protected from further reading/writing by clearing several bits in the OTP section of the flash. This chapter provides details about these OTP bits, as well as the method to access them.

24.1 OTP Bitmap & Definition

79	**78**	**77**	**76**	**75**	**74**	**73**	**72**
< BLKnRD >							
71	**70**	**69**	**68**	**67**	**66**	**65**	**64**
< BLKnRD >				BLK3RD	BLK2RD	BLK1RD	BLK0RD
63	**62**	**61**	**60**	**59**	**58**	**57**	**56**
Reserved							
55	**54**	**53**	**52**	**51**	**50**	**49**	**48**
Reserved				BLK3WR	BLK2WR	BLK1WR	BLK0WR
47	**46**	**45**	**44**	**43**	**42**	**41**	**40**
Reserved							
39	**38**	**37**	**36**	**35**	**34**	**33**	**32**
Reserved							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved							

OTP Bit Definition:

Bit [48..51] - BLKnWR (n = 0, 1, 2, 3): Write protection for each block of the flash memory. The block size is equal to ¼ of the flash size, i.e., 64 kB. Block 0 starts at flash address 0, block 1 starts at flash address 0x10000, and so on.

Bit [64 ..] - BLKnRD (n = 0, 1, 2..): Read protection for each 64-kB block of the flash memory. Depending on the flash size, as many bits can be used as needed. Since the flash size of FT90x is 256 kB, only 4 bits (64 to 67) are used. Block 0 starts at flash address 0, block 1 starts at flash address 0x10000, and so on.

Note: All reserved bits must not be written to.

24.2 OTP Access

The OTP section of the flash can only be written to via the one-wire interface. The one-wire protocol is available in BRT_AN_002 Using the FT9XX Debug Interface. This document is available under NDA. Please approach Bridgetek support team for more information. The contact information can be found in Section 25.

Copyright © Bridgetek Pte Ltd

25 Contact Information

Refer to https://brtchip.com/contact-us/ for contact information.

Distributor and Sales Representatives

Please visit the Sales Network page for the contact details of our distributor(s) and sales representative(s) in your country.

System and equipment manufacturers and designers are responsible to ensure that their systems, and any Bridgetek Pte Ltd (BRTChip) devices incorporated in their systems, meet all applicable safety, regulatory and system-level performance requirements. All application-related information in this document (including application descriptions, suggested Bridgetek devices and other materials) is provided for reference only. While Bridgetek has taken care to assure it is accurate, this information is subject to customer confirmation, and Bridgetek disclaims all liability for system designs and for any applications assistance provided by Bridgetek. Use of Bridgetek devices in life support and/or safety applications is entirely at the user's risk, and the user agrees to defend, indemnify and hold harmless Bridgetek from any and all damages, claims, suits or expense resulting from such use. This document is subject to change without notice. No freedom to use patents or other intellectual property rights is implied by the publication of this document. Neither the whole nor any part of the information contained in, or the product described in this document, may be adapted or reproduced in any material or electronic form without the prior written consent of the copyright holder. Bridgetek Pte Ltd, 1 Tai Seng Avenue, Tower A, #03-05, Singapore 536464. Singapore Registered Company Number: 201542387H.

Copyright © Bridgetek Pte Ltd

Appendix A – References

Document References

AN_324 FT900 User Manual

DS_FT900_1_2_3_Revision_C

DS_FT905_6_7_8_Revision_C

Acronyms and Abbreviations

Terms	Description
ADC	Analogue to Digital Converter
BCD	Battery Charger Detection
CAN	Controller Area Network
CPRM	Content Protection for Recordable Media
CPU	Central Processing unit
CRC	Cyclic Redundancy Check
DAC	Digital to Analogue Converter
FIFO	First In First Out
GPIO	General Purpose Input Output
I^2C	Inter-Integrated Circuit
IP	Intellectual Property
ISR	Interrupt Service Routine
LSB	Least Significant Byte
MAC	Media Access Controller
MDC	Management Data Clock
MII	Media Independent Interface
MMC	Multi Media Card
MSB	Most Significant Byte
SPI	Serial Peripheral Interface
PHY	Physical Layer

Copyright © Bridgetek Pte Ltd

Terms	Description
PWM	Pulse Width Modulation
RAM	Random Access Memory
SD	Secure Digital
STA	Used in Ethernet as the "Station Management Controller"
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
USB-IF	USB Implementers Forum

Copyright © Bridgetek Pte Ltd

Appendix B – List of Tables & Figures

List of Tables

Table 2.1 - FT900 Program Memory Organization 19
Table 3.1 - Peripheral Availability in FT90X Series Revision C Models 21
Table 3.2 - Register Map for FT90X Series Revision C 22
Table 4.1 - Notations used in Register Description 23
Table 5.1 - Overview of General System Registers 26
Table 5.2 - CHIPID - Chip ID Register 27
Table 5.3 - FT90X Series Revision B Configuration 27
Table 5.4 - FT90X Series Revision C Configuration 27
Table 5.5 - CHIPCFG - Chip Configuration Register 28
Table 5.6 - CLKCFG - Clock Configuration Register 29
Table 5.7 - PMCFG - Power Management Register 31
Table 5.8 - PTSTNSET - Test & Set Register 32
Table 5.9 - PTSTNSETR - Test & Set Shadow Register 32
Table 5.10 - MSC0CFG - Miscellaneous Configuration Register 34
Table 5.11 – GPIO Function Register Description 35
Table 5.12 - GPIO 00 – 03 Function Register 35
Table 5.13 - GPIO 04 – 07 Function Register 35
Table 5.14 - GPIO 08 – 11 Function Register 35
Table 5.15 - GPIO 12 – 15 Function Register 36
Table 5.16 - GPIO 16 – 19 Function Register 36
Table 5.17 - GPIO 20 – 23 Function Register 36
Table 5.18 - GPIO 24 – 27 Function Register 36
Table 5.19 - GPIO 28 – 31 Function Register 36
Table 5.20 - GPIO 32 – 35 Function Register 37
Table 5.21 - GPIO 36 – 39 Function Register 37
Table 5.22 - GPIO 40 – 43 Function Register 37
Table 5.23 - GPIO 44 – 47 Function Register 37
Table 5.24 - GPIO 48 – 51 Function Register 38
Table 5.25 - GPIO 52 – 55 Function Register 38
Table 5.26 - GPIO 56 – 59 Function Register 38
Table 5.27 - GPIO 60 – 63 Function Register 39
Table 5.28 - GPIO 64 – 66 Function Register 39
Table 5.29 - GPIO Configuration Register Description 40
Table 5.30 - GPIO 00 – 07 Configuration Register 40
Table 5.31 - GPIO 08 – 15 Configuration Register 40

Copyright © Bridgetek Pte Ltd

Table 5.32 - GPIO 16 – 23 Configuration Register 40
Table 5.33 - GPIO 24 – 31 Configuration Register 41
Table 5.34 - GPIO 32 – 39 Configuration Register 41
Table 5.35 - GPIO 40 – 47 Configuration Register 42
Table 5.36 - GPIO 48 – 55 Configuration Register 42
Table 5.37 - GPIO 56 – 63 Configuration Register 42
Table 5.38 - GPIO 64 – 66 Configuration Register 42
Table 5.39 - GPIO 00 – 31 Value Register 43
Table 5.40 - GPIO 32 – 63 Value Register 43
Table 5.41 - GPIO 64 – 66 Value Register 43
Table 5.42 - GPIO 00 – 31 Interrupt Enable Register 43
Table 5.43 - GPIO 32 – 63 Interrupt Enable Register 43
Table 5.44 - GPIO 64 – 66 Interrupt Enable Register 43
Table 5.45 - GPIO 00 – 31 Interrupt Pending Register 44
Table 5.46 - GPIO 32 – 63 Interrupt Pending Register 44
Table 5.47 - GPIO 64 – 66 Interrupt Pending Register 44
Table 5.48 - ETH_PHY_CFG - Ethernet PHY Miscellaneous Configuration Register 45
Table 5.49 - ETH_PHY_ID - Ethernet PHY ID Register 45
Table 5.50 - DAC_ADC_CONF - ADC/DAC Configuration/Status Register 47
Table 5.51 - DAC_ADC_CNT - ADC/DAC Count Register 48
Table 5.52 - DAC_ADC_DATA - ADC/DAC Data Register 48
Table 5.53 - RESET_STATUS – Global Reset/Status Register 48
Table 6.1 – Interrupt Assignment Table 49
Table 6.2 – Overview of Interrupt Control Registers 50
Table 6.3 – Interrupt 00-03 Priority Assignment Register 50
Table 6.4 – Interrupt 04-07 Priority Assignment Register 50
Table 6.5 – Interrupt 08-11 Priority Assignment Register 51
Table 6.6 – Interrupt 12-15 Priority Assignment Register 51
Table 6.7 – Interrupt 16-19 Priority Assignment Register 51
Table 6.8 – Interrupt 20-23 Priority Assignment Register 51
Table 6.9 – Interrupt 24-27 Priority Assignment Register 52
Table 6.10 – Interrupt 28-31 Priority Assignment Register 52
Table 6.11 – Interrupt Control Register 52
Table 7.1 – Overview of USB Host Controller Registers 53
Table 7.2 – HC Capability Register 54
Table 7.3 – HCSPARAMS – HC Structural Parameters 54
Table 7.4 – HCCPARAMS – HC Capability Parameters 54
Table 7.5 – USBCMD – HC USB Command Register 55
Table 7.6 – USBSTS – HC USB Status Register 56

Copyright © Bridgetek Pte Ltd

Table 7.7 – USBINTR – HC USB Interrupt Enable Register 57
Table 7.8 – FRINDEX – HC Frame Index Register 57
Table 7.9 – PERIODICLISTBASE – HC Periodic Frame List Base Address Register 57
Table 7.10 – ASYNCLISTADDR – HC Current Asynchronous List Address Register 58
Table 7.11 – PORTSC – HC Port Status and Control Register 59
Table 7.12 – EOF Time & Asynchronous Schedule Sleep Timer Register 60
Table 7.13 – Bus Monitor Control / Status Register 61
Table 7.14 – HPROT – Master Protection Information Setting Register 61
Table 7.15 – Vendor Specific IO Control Register 61
Table 7.16 – Vendor Specific Status Register 61
Table 7.17 – Test Register 62
Table 8.1 – Overview of USB Device Registers 65
Table 8.2 – DC_ADDRESS_ENABLE – Address Register 65
Table 8.3 – DC_MODE – Mode Register 66
Table 8.4 – DC_INT_ENABLE – Interrupt Enable Register 66
Table 8.5 – DC_EP_INT_ENABLE – Endpoints Interrupt Enable Register 66
Table 8.6 – DC_EP0_CONTROL – Endpoint 0 Control Register 67
Table 8.7 – DC_EP0_STATUS – Endpoint 0 Status Register 67
Table 8.8 – DC_EP0_BUFFER_LENGTH – Endpoint 0 Buffer Length Register 68
Table 8.9 – DC_EP0_BUFFER – Endpoint 0 Buffer Register 68
Table 8.10 – DC_EP(x)_CONTROL – Endpoint Control Registers 69
Table 8.11 – DC_EP(x)_STATUS – Endpoint Status Registers 70
Table 8.12 – DC_EP(x)_BUFFER_LENGTH_LSB – Endpoint Buffer Length LSB Registers 70
Table 8.13 – DC_EP(x)_BUFFER_LENGTH_MSB – Endpoint Buffer Length MSB Registers 70
Table 8.14 – DC_EP(x)_BUFFER – Endpoint Buffer Registers 70
Table 8.15 – DC_INT_STATUS – Interrupt Status Register 71
Table 8.16 – DC_EP_INT_STATUS – Endpoints Interrupt Status Register 71
Table 8.17 – DC_FRAME_NUMBER_LSB – Frame Number LSB Register 71
Table 8.18 – DC_FRAME_NUMBER_MSB – Frame Number MSB Register 71
Table 8.19 – HBWCTRL0 – High-bandwidth Isochronous Control Register 0 72
Table 8.20 – HBWCTRL1 – High-bandwidth Isochronous Control Register 1 72
Table 8.21 – HBWCTRL2 – High-bandwidth Isochronous Control Register 2 73
Table 8.22 – HBWCTRL3 – High-bandwidth Isochronous Control Register 3 73
Table 8.23 – HBW_INT_ENABLE – High-bandwidth Interrupt Enable Register 74
Table 8.24 – HBW_INT_STATUS – High-bandwidth Interrupt Status Register 74
Table 9.1 – Memory Organization of TX/RX RAM 77
Table 9.2 – Overview of Ethernet Registers 78
Table 9.3 – ETH_INT_STATUS – Interrupt Status Register 79
Table 9.4 – ETH_INT_ENABLE – Interrupt Enable Register 79

Copyright © Bridgetek Pte Ltd

Table 9.5 – ETH_RX_CNTL – Receive Control Register 80
Table 9.6 – ETH_TX_CNTL – Transmit Control Register 80
Table 9.7 – ETH_DATA_N0 – Data Register (octet n) 80
Table 9.8 – ETH_DATA_N1 – Data Register (octet n+1) 81
Table 9.9 – ETH_DATA_N2 – Data Register (octet n+2) 81
Table 9.10 – ETH_DATA_N3 – Data Register (octet n+3) 81
Table 9.11 – ETH_ADDR_1 – Address Register (octet 1) 81
Table 9.12 – ETH_ADDR_2 – Address Register (octet 2) 81
Table 9.13 – ETH_ADDR_3 – Address Register (octet 3) 81
Table 9.14 – ETH_ADDR_4 – Address Register (octet 4) 81
Table 9.15 – ETH_ADDR_5 – Address Register (octet 5) 81
Table 9.16 – ETH_ADDR_6 – Address Register (octet 6) 81
Table 9.17 – ETH_THRESHOLD – Threshold Register 82
Table 9.18 – ETH_MNG_CNTL – Management Control Register 82
Table 9.19 – ETH_MNG_DIV – Management Divider Register 82
Table 9.20 – ETH_MNG_ADDR – Management Address Register 82
Table 9.21 – ETH_MNG_TX0 – Management Transmit Data 0 Register 83
Table 9.22 – ETH_MNG_TX1 – Management Transmit Data 1 Register 83
Table 9.23 – ETH_MNG_RX0 – Management Receive Data 0 Register 83
Table 9.24 – ETH_MNG_RX1 – Management Receive Data 1 Register 83
Table 9.25 – ETH_NUM_PKT – Number of Packets Register 83
Table 9.26 – ETH_TR_REQ – Transmission Request Register 83
Table 10.1 – Symbols used in the CAN Frame buffer 84
Table 10.2 – Standard Frames Memory Buffer Layout 84
Table 10.3 – Extended Frames Memory Buffer Layout 85
Table 10.4 – Overview of CAN Registers 86
Table 10.5 – CAN_MODE – Mode Register 86
Table 10.6 – CAN_CMD – Command Register 87
Table 10.7 – CAN_STATUS – Status Register 88
Table 10.8 – CAN_INT_STATUS – Interrupt Status Register 88
Table 10.9 – CAN_INT_ENABLE – Interrupt Enable Register 89
Table 10.10 – CAN_RX_MSG – Receive Message Register 89
Table 10.11 – CAN_BUS_TIM_0 – Bus Timing 0 Register 89
Table 10.12 – CAN_BUS_TIM_1 – Bus Timing 1 Register 90
Table 10.13 – CAN_TX_BUF_0 – Transmit Buffer 0 Register 90
Table 10.14 – CAN_TX_BUF_1 – Transmit Buffer 1 Register 90
Table 10.15 – CAN_TX_BUF_2 – Transmit Buffer 2 Register 91
Table 10.16 – CAN_TX_BUF_3 – Transmit Buffer 3 Register 91
Table 10.17 – CAN_RX_BUF_0 – Receive Buffer 0 Register 91

Copyright © Bridgetek Pte Ltd

Table 10.18 – CAN_RX_BUF_1 – Receive Buffer 1 Register 91
Table 10.19 – CAN_RX_BUF_2 – Receive Buffer 2 Register 91
Table 10.20 – CAN_RX_BUF_3 – Receive Buffer 3 Register 92
Table 10.21- CAN_ACC_CODE_0 – Acceptance Code 0 Register 93
Table 10.22 – CAN_ACC_CODE_1 – Acceptance Code 1 Register 93
Table 10.23 - CAN_ACC_CODE_2 – Acceptance Code 2 Register 94
Table 10.24 – CAN_ACC_CODE_3 – Acceptance Code 3 Register 94
Table 10.25 – CAN_ACC_MASK_0 – Acceptance Mask 0 Register 94
Table 10.26 – CAN_ACC_MASK_1 – Acceptance Mask 1 Register 94
Table 10.27 – CAN_ACC_MASK_2 – Acceptance Mask 2 Register 94
Table 10.28 – CAN_ACC_MASK_3 – Acceptance Mask 3 Register 95
Table 10.29 – CAN_ERR_CODE – Error Code Capture Register 96
Table 10.30 – CAN_RX_ERR_CNTR – Receive Error Counter Register 96
Table 10.31 – CAN_TX_ERR_CNTR – Transmit Error Counter Register 96
Table 10.32 – CAN_ARB_LOST_CODE – Arbitration Lost Code Capture Register 96
Table 11.1 – Overview of SD Host Registers 99
Table 11.2 – SDH_AUTO_CMD23_ARG2 – Auto CMD23 Argument 2 Register 99
Table 11.3 – SDH_BLK_SIZE – Block Size Register 100
Table 11.4 – SDH_BLK_COUNT – Block Count Register 100
Table 11.5 – SDH_ARG_1 – Argument 1 Register 100
Table 11.6 – SDH_TNSFER_MODE – Transfer Mode Register 101
Table 11.7 – SDH_CMD – Command Register 102
Table 11.8 – SDH_RESPONSE – Response Register 102
Table 11.9 – 11.2.8 SDH_BUF_DATA – Buffer Data Port Register 102
Table 11.10 – SDH_PRESENT_STATE – Present State Register 104
Table 11.11 – SDH_HST_CNTL_1 – Host Control 1 Register 104
Table 11.12 – SDH_PWR_CNTL – Power Control Register 105
Table 11.13 – SDH_BLK_GAP_CNTL – Block Gap Control Register 105
Table 11.14 – SDH_CLK_CNTL – Clock Control Register 106
Table 11.15 – SDH_TIMEOUT_CNTL – Timeout Control Register 106
Table 11.16 – SDH_SW_RST – Software Reset Register 106
Table 11.17 – SDH_NRML_INT_STATUS – Normal Interrupt Status Register 107
Table 11.18 – SDH_ERR_INT_STATUS – Error Interrupt Status Register 108
Table 11.19 – SDH_NRML_INT_ENABLE – Normal Interrupt Status Enable Register 108
Table 11.20 – SDH_ERR_INT_ENABLE – Error Interrupt Status Enable Register 109
Table 11.21 – SDH_NRML_INT_SGNL_ENABLE – Normal Interrupt Signal Enable Register 109
Table 11.22 – SDH_ERR_INT_SGNL_ENABLE – Error Interrupt Signal Enable Register 110
Table 11.23 – SDH_AUTO_CMD12_ERR_STATUS – Auto CMD12 Error Status Register 110
Table 11.24 – SDH_HOST_CNTL_2 – Host Control 2 Register 110

Copyright © Bridgetek Pte Ltd

Table 11.25 – SDH_CAP_1 – Capabilities Register 1 111
Table 11.26 – SDH_CAP_2 – Capabilities Register 2 112
Table 11.27 – SDH_RSRV_1 – Reserved 1 Register 112
Table 11.28 – SDH_RSRV_2 – Reserved 2 Register 112
Table 11.29 – SDH_FORCE_EVT_CMD_ERR_STATUS – Force Event Register for Auto CMD Error Status 113
Table 11.30 – SDH_FORCE_EVT_ERR_INT_STATUS – Force Event for Error Interrupt Status Register 113
Table 11.31- SDH_RSRV_3 – Reserved 3 Register 113
Table 11.32 – SDH_RSRV_4 – Reserved 4 Register 114
Table 11.33 – SDH_PRST_INIT – Preset value for initialization 114
Table 11.34 – SDH_PRST_DFLT_SPD – Preset value for default speed 114
Table 11.35 – SDH_PRST_HIGH_SPD – Preset value for the high speed 114
Table 11.36 – SDH_PRST_SDR12 – Preset value for SDR12 115
Table 11.37 – SDH_PRST_SDR25 – Preset value for SDR25 116
Table 11.38 – SDH_PRST_SDR50 – Preset value for SDR50 116
Table 11.39 – SDH_PRST_SDR104 – Preset value for SDR104 116
Table 11.40 – SDH_PRST_DDR50 – Preset value for DDR50 117
Table 11.41 – SDH_RSRV_5 – Reserved 5 Register 117
Table 11.42 – SDH_HC_VER – Host Controller Version Register 117
Table 11.43 – SDH_VNDR_0 – Vendor-defined 0 Register 118
Table 11.44 – SDH_VNDR_1 – Vendor-defined 1 Register 120
Table 11.45 – SDH_VNDR_2 – Vendor-defined 2 Register 120
Table 11.46 – SDH_VNDR_3 – Vendor-defined 3 Register 121
Table 11.47 – SDH_VNDR_4 – Vendor-defined 4 Register 121
Table 11.48 – SDH_VNDR_5 – Vendor-defined 5 Register 122
Table 11.49 – SDH_VNDR_6 – Vendor-defined 6 Register 122
Table 11.50 – SDH_VNDR_7 – Vendor-defined 7 Register 122
Table 11.51 – SDH_VNDR_8 – Vendor-defined 8 Register 122
Table 11.52 – SDH_VNDR_9 – Vendor-defined 9 Register 122
Table 11.53 – SDH_RSRV_6 – Reserved 6 Register 122
Table 11.54 – SDH_HW_ATTR – Hardware Attributes Register 122
Table 11.55 – SDH_CPR_MOD_CNTL – Cipher Mode Control Register 123
Table 11.56 – SDH_CPR_MOD_STATUS – Cipher Mode Status Register 123
Table 11.57 – SDH_CPR_MOD_STATUS_EN – Cipher Mode Status Enable Register 124
Table 11.58 – SDH_CPR_MOD_SIG_EN – Cipher Mode Signal Enable Register 124
Table 11.59 – SDH_IN_DATA_LSB –Input Data LSB Register 124
Table 11.60 – SDH_IN_DATA_MSB –Input Data MSB Register 124
Table 11.61 – SDH_IN_KEY_LSB – Input Key LSB Register 124
Table 11.62 – SDH_IN_KEY_MSB – Input Key MSB Register 124

Copyright © Bridgetek Pte Ltd

Table 11.63 – SDH_OUT_DATA_LSB – Output Data LSB Register 124
Table 11.64 – SDH_OUT_DATA_MSB – Output Data MSB Register 125
Table 11.65 – SDH_SCRT_CONS_DATA – Secret Constant Table Data Port....125
Table 12.1 – Overview of UART Registers....128
Table 12.2 – UART Mode Selection 129
Table 12.3 – FIFOSEL, CLKSEL and INTSEL configuration 131
Table 12.4 – Sample Clock Values....131
Table 12.5 – UART_RBR – Receiver Buffer Register 131
Table 12.6 – UART_THR – Transmitter Holding Register....132
Table 12.7 – UART_DIV_LSB – Divisor LSB Register 132
Table 12.8 – UART_DIV_MSB – Divisor MSB Register 132
Table 12.9 – UART_IER – Interrupt Enable Register 132
Table 12.10 – UART_ ISR – Interrupt Status Register 132
Table 12.11 – Interrupt Status Register Software Handling 133
Table 12.12 – UART_FCR – FIFO Control Register – 16550/16750 mode 134
Table 12.13 – UART_RCVR – FIFO Trigger Level – 16550/16750 mode 134
Table 12.14 - UART_FCR – FIFO Control Register – 16650 mode....134
Table 12.15 - Receiver FIFO Trigger Level in 16650 Mode 135
Table 12.16 - Transmitter FIFO Trigger Level in 16650 Mode 135
Table 12.17 – UART_LCR – Line Control Register 136
Table 12.18 – UART_MCR – Modem Control Register 137
Table 12.19 – UART_LSR – Line Status Register 138
Table 12.20 – UART_MSR – Modem Status Register 139
Table 12.21 – UART_SPR – SPR Register 140
Table 12.22 – UART_EFR – Enhanced Feature Register 140
Table 12.23 – UART_XON1 – XON1 Register....140
Table 12.24 – UART_XON2 – XON2 Register....140
Table 12.25 – UART_XOFF1 – XOFF1 Register 140
Table 12.26 – UART_XOFF2 – XOFF2 Register 141
Table 12.27 - UART_ASR – Additional Status Register....141
Table 12.28 - UART_RFL – Receiver FIFO Level Register 141
Table 12.29 - UART_TFL – Transmitter FIFO Level Register 141
Table 12.30 - UART_ICR – ICR Register 142
Table 12.31 – UART_ACR- Additional Control Register 144
Table 12.32 – UART_CPR – Clock Pre-scaler Register....145
Table 12.33 – UART_TCR – Time Clock Register....145
Table 12.34 – UART_CKS Clock Select Register....145
Table 12.35 – UART_TTL – Transmitter Trigger Level Register 146
Table 12.36 – UART_RTL – Receiver Trigger Level Register 146

Copyright © Bridgetek Pte Ltd

Table 12.37 – UART_FCL – Flow Control Level LSB Register 146
Table 12.38 – UART_FCH – Flow Control Level Register MSB 146
Table 12.39 – UART_ID1 – Identification 1 Register 147
Table 12.40 – UART_ID2 – Identification 2 Register 147
Table 12.41 UART_ID3 – Identification 3 Register 147
Table 12.42 – UART_REV – Revision Register 147
Table 12.43 – UART_CSR – Channel Software Reset Register 147
Table 12.44 – UART_NMR – Nine Bit Mode Register.....148
Table 12.45 – UART_MDM – Modem Disable Mask Register 148
Table 12.46 – UART_RFC – Readable FCR Register.....148
Table 12.47 – UART_GDS – Good Data Status Register.....149
Table 12.48 – UART_RSRV_1 – Reserved Register 1.....149
Table 12.49 – UART_PIDX – Port Index Register 149
Table 12.50 – UART_RSRV_2 – Reserved Register 2.....149
Table 13.1 – Timers/Watchdog Operation 151
Table 13.2 – Overview of Timers/Watchdog Registers.....152
Table 13.3 – TIMER_CONTROL_0 – Timers Control Register 0.....152
Table 13.4 – TIMER_CONTROL_1 – Timers Control Register 1.....152
Table 13.5 – TIMER_CONTROL_2 – Timers Control Register 2.....152
Table 13.6 – TIMER_CONTROL_3 – Timers Control Register 3.....153
Table 13.7 – TIMER_CONTROL_4 – Timers Control Register 4.....153
Table 13.8 – TIMER_INT – Timers Interrupt Register.....153
Table 13.9 – TIMER_SELECT – Timers A..D Select Register 153
Table 13.10 – TIMER_WDG – Watchdog Start Value 154
Table 13.11 – TIMER_WRITE_LSB – Timer A..D Start Value 7:0.....154
Table 13.12 – TIMER_WRITE_MSB – Timer A..D Start Value 15:8 154
Table 13.13 – TIMER_PRESC_LSB – Pre-scaler Start Value 7:0 154
Table 13.14 – TIMER_PRESC_MSB – Pre-scaler Start Value 15:8 154
Table 13.15 – TIMER_READ_LSB – Timer A..D Current Value 7:0.....154
Table 13.16 – TIMER_READ_MSB – Timer A..D Current Value 15:8 155
Table 13.17 – TIMER_RESERVED – Reserved Register 155
Table 13.18 – WDT_STATUS – Watchdog Status Register.....155
Table 14.1 – Oversampling rates supported by FT90x Revision C I^2S 157
Table 14.2 – FT90x Revision C I^2S settings.....157
Table 14.3 – Overview of I^2S Registers 158
Table 14.4 – I2SCR – Configuration Register 1 159
Table 14.5 – I2SCR2 – Configuration Register 2.....159
Table 14.6 – I2SIRQEN – Interrupt Enable Register.....160
Table 14.7 – I2SIRQPEND – Interrupt Pending Register 160

Copyright © Bridgetek Pte Ltd

Table 14.8 – I2SRWDATA – Transmit / Receive Data Register....160
Table 14.9 – I2SRXCOUNT – RX Count Register....161
Table 14.10 – I2STXCOUNT – TX Count Register....161
Table 15.1 – Overview of SPI Master Registers....162
Table 15.2 – Overview of SPI Master Extended Registers....163
Table 15.3 – SPI Master FIFO Size....163
Table 15.4 – SPIM_CNTL – Control Register....167
Table 15.5 – SPIM_STATUS – Status Register....168
Table 15.6 – SPIM_DATA – Receiver and Transmitter Data Registers....168
Table 15.7 – SPIM_SLV_SEL_CNTL – Slave Select Control Register....168
Table 15.8 – SPIM_FIFO_CNTL – FIFO Control Register....169
Table 15.9 – SPIM_TNSFR_FRMT_CNTL – Transfer Format Control Register....169
Table 15.10 – SPIM_ALT_DATA – Alternative SPI Master Data Register....169
Table 15.11 – SPIM_RX_FIFO_COUNT – SPI Master RX FIFO Count Register....170
Table 15.12 – SPIM_CNTL_2 – Control Register 2....170
Table 15.13 – SPIM_STATUS_2 – Status Register 2....171
Table 15.14 – SPIM_FIFO_CNTL_2 – FIFO Control Register 2....171
Table 15.15 – SPIM_TX_FIFO_COUNT – SPI Master TX FIFO Count Register....171
Table 15.16 – SPIM_BAUD – SPI Master Baud Register....172
Table 16.1 – Overview of SPI Slave Registers....173
Table 16.2 – Overview of SPI Slave Extended Registers....174
Table 16.3 – SPI Slave FIFO Size....174
Table 16.4 – SPIS_CNTL – Control Register....177
Table 16.5 – SPIS_STATUS – Status Register....178
Table 16.6 – SPIS_DATA – Receiver and Transmitter Data Registers....178
Table 16.7 – SPIS_RESERVED – Reserved Register....178
Table 16.8 – SPIS_FIFO_CNTL – FIFO Control Register....179
Table 16.9 – SPIS_TNSFR_FRMT_CNTL – Transfer Format Control Register....180
Table 16.10 – SPIS_ALT_DATA – Alternative SPI Slave Data Register....180
Table 16.11 – SPIS_RX_FIFO_COUNT – SPI Slave RX FIFO Count Register....181
Table 16.12 – SPIS_CNTL_2 – Control Register 2....181
Table 16.13 – SPIS_STATUS_2 – Status Register 2....183
Table 16.14 – SPIS_FIFO_CNTL_2 – FIFO Control Register 2....183
Table 16.15 – SPIS_TX_FIFO_COUNT – SPI Slave TX FIFO Count Register....183
Table 17.1 – Overview of I^2C Master Registers....184
Table 17.2 – I2CM_SLV_ADDR – Slave Address Register....185
Table 17.3 – I2CM_CNTL – Control Register....186
Table 17.4 – Control bit combinations permitted in idle state....186
Table 17.5 – Control bit combinations permitted in Master Transmitter mode....187

Copyright © Bridgetek Pte Ltd

Table 17.6 – Control bit combinations permitted in Master Receiver mode 187
Table 17.7 – I2CM_STATUS – Status Register 188
Table 17.8 – I2CM_DATA – Receive / Transmit Data Register 188
Table 17.9 – I2CM_TIMER_PERIOD – Timer Period Register 189
Table 17.10 – I2CM_HS_TIMER_PERIOD – High-speed Timer Period Register 189
Table 17.11 – I2CM_FIFO_LEN – FIFO Mode Byte Length 189
Table 17.12 – I2CM_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable 190
Table 17.13 – I2CM_FIFO_INT_PEND – FIFO Mode Interrupt Pending 190
Table 17.14 – I2CM_FIFO_DATA – FIFO Data Register 190
Table 17.15 – I2CM_TRIG – FIFO Operation Trigger Register 190
Table 18.1 – Overview of I2C Master Registers 191
Table 18.2 – I2CS_OWN_ADDR – Own Address Register 192
Table 18.3 – I2CS_CNTL – Control Register 193
Table 18.4 – I2CS_STATUS – Status Register 194
Table 18.5 – I2CS_DATA – Receive / Transmit Data Register 194
Table 18.6 – I2CS_FIFO_STATUS – FIFO Mode Status Register 194
Table 18.7 – I2CS_FIFO_INT_ENABLE – FIFO Mode Interrupt Enable 194
Table 18.8 – I2CS_FIFO_INT_PEND – FIFO Mode Interrupt Pending 195
Table 18.9 – I2CS_FIFO_DATA – FIFO Data Register 195
Table 19.1 – Overview of RTC Registers 196
Table 19.2 – Calibration Time 198
Table 19.3 – Calibration Time/Accuracy vs. Frequency Range 198
Table 19.4 – RTC_TIME – Time Register 198
Table 19.5 – RTC_DATE – Date Register 199
Table 19.6 – RTC_AALARM1 – Auto Alarm 1 Register 199
Table 19.7 – Alarm 1 Mask Bits 199
Table 19.8 – RTC_AALARM2 – Auto Alarm 2 Register 200
Table 19.9 – Alarm 2 Mask Bits 200
Table 19.10 – RTC_CTRL – Control Register 201
Table 19.11 – RTC_STAT – Status Register 201
Table 19.12 – RTC_TRIM – Trimming Register 202
Table 19.13 – RTC_COMP_VERSION – Component Version Register 202
Table 19.14 – RTC_SLEEP_DATE - Sleep Date Register 202
Table 19.15 – RTC_COMP_PER – Compare Period Register 203
Table 19.16 – RTC_MISC – Miscellaneous Register 203
Table 20.1 – Overview of PWM Registers 205
Table 20.2 – Comparators values to generate the example waveform 207
Table 20.3 – Comparators values to generate 50% duty cycle 207
Table 20.4 – PWM_CTRL0 – PCM Control Register 208

Copyright © Bridgetek Pte Ltd

Table 20.5 – PWM_CTRL1 – PWM Control Register 208
Table 20.6 – PWM_PRESCALER – PWM Pre-scaler Register 209
Table 20.7 – PWM_CNTL – PWM Counter Register (LSB) 209
Table 20.8 – PWM_CNTH – PWM Counter Register (MSB) 209
Table 20.9 – PWM_CMP0L – Comparator 0 Value Register (LSB) 209
Table 20.10 – PWM_CMP0H – Comparator 0 Value Register (MSB) 209
Table 20.11 – PWM_CMP1L – Comparator 1 Value Register (LSB) 209
Table 20.12 – PWM_CMP1H – Comparator 1 Value Register (MSB) 209
Table 20.13 – PWM_CMP2L – Comparator 2 Value Register (LSB) 210
Table 20.14 – PWM_CMP2H – Comparator 2 Value Register (MSB) 210
Table 20.15 – PWM_CMP3L – Comparator 3 Value Register (LSB) 210
Table 20.16 – PWM_CMP3H – Comparator 3 Value Register (MSB) 210
Table 20.17 – PWM_CMP4L – Comparator 4 Value Register (LSB) 210
Table 20.18 – PWM_CMP4H – Comparator 4 Value Register (MSB) 210
Table 20.19 – PWM_CMP5L – Comparator 5 Value Register (LSB) 210
Table 20.20 – PWM_CMP5H – Comparator 5 Value Register (MSB) 211
Table 20.21 – PWM_CMP6L – Comparator 6 Value Register (LSB) 211
Table 20.22 – PWM_CMP6H – Comparator 6 Value Register (MSB) 211
Table 20.23 – PWM_CMP7L – Comparator 7 Value Register (LSB) 211
Table 20.24 – PWM_CMP7H – Comparator 7 Value Register (MSB) 211
Table 20.25 – PWM_TOGGLE0 – Channel 0 OUT Toggle Comparator Mask.....211
Table 20.26 – PWM_TOGGLE1 – Channel 1 OUT Toggle Comparator Mask.....212
Table 20.27 – PWM_TOGGLE2 – Channel 2 OUT Toggle Comparator Mask Register 212
Table 20.28 – PWM_TOGGLE3 – Channel 3 OUT Toggle Comparator Mask Register 212
Table 20.29 – PWM_TOGGLE4 – Channel 4 OUT Toggle Comparator Mask Register 212
Table 20.30 – PWM_TOGGLE5 – Channel 5 OUT Toggle Comparator Mask Register 212
Table 20.31 – PWM_TOGGLE6 – Channel 6 OUT Toggle Comparator Mask Register 213
Table 20.32 – PWM_TOGGLE7 – Channel 7 OUT Toggle Comparator Mask Register 213
Table 20.33 – PWM_OUT_CLR_EN – PWM OUT Clear Enable Register 213
Table 20.34 – PWM_CTRL_BL_CMP8 – Control Block CMP8 Value Register 213
Table 20.35 – PWM_INIT – PWM Initialization Register 213
Table 20.36 – PWM_INTMASK – PWM Interrupt Mask Register.....213
Table 20.37 – PWM_INTSTATUS – PWM Interrupt Status Register.....214
Table 20.38 – PWM_SAMPLE_FREQ_H – PWM Data Sampling Frequency High Byte Register.....214
Table 20.39 – PWM_SAMPLE_FREQ_L – PWM Data Sampling Frequency Low Byte Register 214
Table 20.40 – PCM_VOLUME – PCM Volume Register.....215
Table 20.41 – PWM_BUFFER – PCM Buffer Register 215
Table 21.1 – Overview of Data Capture Interface Registers 216
Table 21.2 – DCAP_REG1 – Data Capture Interface Register 1.....216

Copyright © Bridgetek Pte Ltd

Table 21.3 – DCAP_REG2 – Data Capture Interface Register 2 216
Table 21.4 – DCAP_REG3 – Data Capture Interface Register 3 217
Table 21.5 – DCAP_REG4 – Data Capture Interface Register 4 217
Table 22.1 – Overview of Data Capture Interface Registers 219
Table 22.2 – RSADDR1 – Memory Start Address Register (LSB) 219
Table 22.3 – RSADDR1 – Memory Start Address Register (Byte 1) 219
Table 22.4 – RSADDR2 – Memory Start Address Register (MSB) 219
Table 22.5 – FSADDR0 – Flash Start Address Register (LSB) 220
Table 22.6 – FSADDR1 – Flash Start Address Register (Byte 1) 220
Table 22.7 – FSADDR2 – Flash Start Address Register (MSB) 220
Table 22.8 – BLENGTH0 – Data Byte Length Register (LSB) (LSB) 220
Table 22.9 – BLENGTH1 – Data Byte Length Register (Byte 1) 220
Table 22.10 – BLENGTH2 – Data Byte Length Register (MSB) 220
Table 22.11 – COMMAND – Command Register 221
Table 22.12 – SEMAPHORE – Semaphore Register 221
Table 22.13 – CONFIG – Configuration Register 221
Table 22.14 – STATUS – Status Register 221
Table 22.15 – CRCL – Flash Content CRC Register (LSB) 222
Table 22.16 – CRCH – Flash Content CRC Register (MSB) 222
Table 22.17 – CHIPID0 – Chip ID Register (LSB) 222
Table 22.18 – CHIPID1 – Chip ID Register (Byte 1) 222
Table 22.19 – CHIPID2 – Chip ID Register (Byte 2) 222
Table 22.20 – CHIPID3 – Chip ID Register (MSB) 222
Table 22.21 – DRWDATA – Data Register 223
Table 22.22 – Flash Controller Command Group 1 224
Table 22.23 – Flash Controller Command Group 2 226

Copyright © Bridgetek Pte Ltd

List of Figures

Figure 2.1 - FT900 Revision C System Architecture 19
Figure 2.2 - FT90X Boot Control 20
Figure 2.3 - FT90X Debugging Support 20
Figure 8.1 - UVC Payload Header transferred over an IN endpoint 75
Figure 8.2 - Format of the UVC Payload Header 75
Figure 8.3 - Example of an UVC Payload Header on the USB Bus 75
Figure 10.1 – CAN Acceptance Filter 92
Figure 12.1 – ICR registers write access 142
Figure 12.2 – ICR registers read access 143
Figure 15.1 – SPI Master Register Access 163
Figure 15.2 – SPI Master 8-bit FIFO Access 163
Figure 15.3 – SPI Master 32-bit FIFO Access 163
Figure 15.4 – SPI Master Block Diagram 164
Figure 16.1 – SPI Slave Register Access 174
Figure 16.2 – SPI Slave 8-bit FIFO Access 174
Figure 16.3 – SPI Slave 32-bit FIFO Access 174
Figure 16.4 – SPI Slave Block Diagram 175
Figure 20.1 – PWM Block Diagram 206
Figure 20.2 – PWM Waveform generated by 4 Comparators 207

Copyright © Bridgetek Pte Ltd

Appendix C – Revision History

Document Title: BRT_AN_020 FT90X Revision C User Manual

Document Reference No.: BRT_000174

Clearance No.: BRT#124

Product Page: https://brtchip.com/ft9xx-toolchain/

Document Feedback: Send Feedback

Revision	Changes	Date
1.0	Initial Release	19-01-2018
1.1	Updated watchdog information & Company address Updated for v2.7.0 toolchain release	29-08-2023

Copyright © Bridgetek Pte Ltd