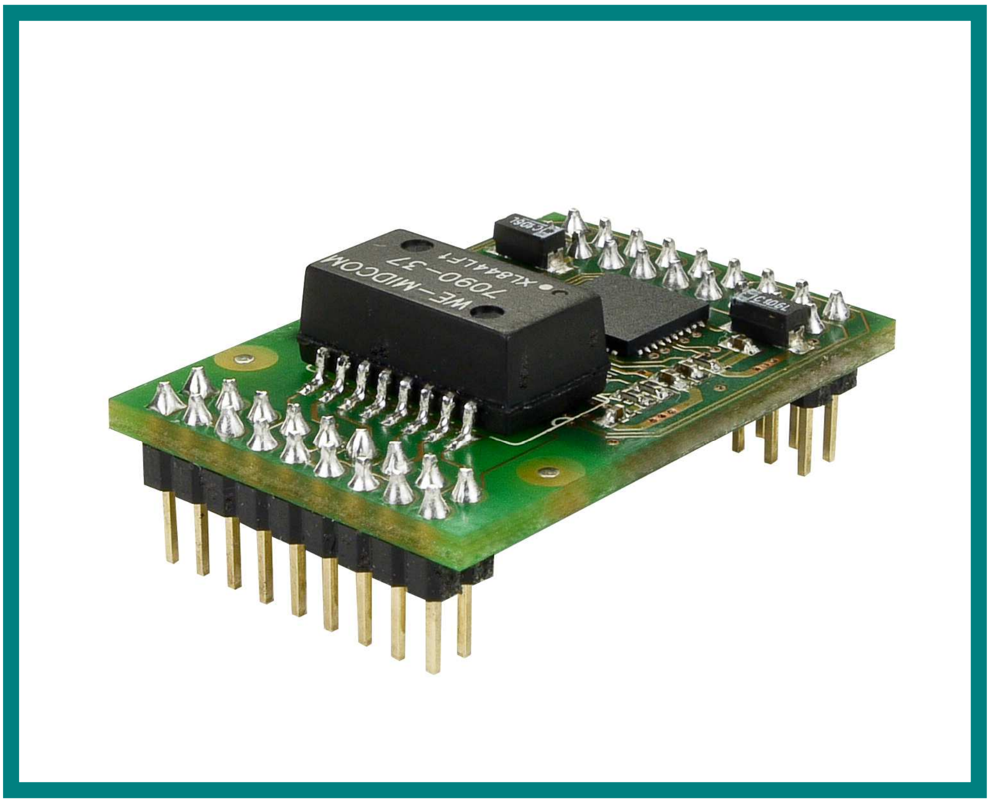
XT- PICO -XXL



DesignGuide V1.0

1 Description

Thanks to its extremely little dimensions of only **22 x 32 mm** the new **XT-PICO–XXL** module is very well appropriate to be integrated also to very small terminals. A total of **two** bus systems are made available of each four or five switchable interfaces such as **RS232**, **RS485**, **I2C, SPI** as well as **TTL-IO**. Even a **POE** (Power over Ethernet) supply is possible since all necessary connections are executed. Lots of functions such as e.g. a 4-bit or SPI LCD display as well as the direct connection of the SD card are directly selectable.

2 Interfaces and functions

Interface (TTL level (3.3V)**)**

RS232	Baud rate	: up to 1M baud
	Data bits	: 7,8
	Parity	: Odd,Even,None Mark,Space
RS485	Baud rate	: bis 1MBaud
	Data bits	: 7,8
	Parity	: Odd,Even,None Mark,Space
I2C	Data rate	: up to 400KHz
SPI	Data rate	: up to 25 M bit
TTL-IO	7 or 8 pin for digital input/output	

RS232(TTL)

It is possible to use up to 2 independent individually operating serial interfaces. Each interface can be individually set and it is possible to transfer data rates of up to 1.000.000 bauds. Furthermore, it is possible to additionally set emulations such as modem, Auto-Connect, Connect-On-Data, TCP / UDP - Client with up to 10 parallel connections, TCP/UDP server, tunnel mode with transfers of the signal modes as well as settings, E-Mail Client with sending and receiving of e-mails.

RS485(TTL)

It is possible to use up to 2 independently operating RS485 interfaces. This mode also supports so-called 2-wire components such as e.g. the MAX3072E, since it does not possess a proper control wire. Each interface can be individually set and it is possible to transfer data rates of up to 1.000.000 bauds. Furthermore, it is possible to additionally set emulations such as modem, AutoConnect, Conntect-On-Data, TCP/UDP-Client with up to 10 parallel connections, TCP/UDP server, tunnel mode with transfer of the signal modes as well as settings, E-Mail Client with sending and receiving of e-mails.

I2C

Indeed it is only possible to operate one I2C interface in the master mode; however it is possible to control it via two different ports with e.g. different addresses. A data mode had also been implemented in order to achieve a maximum of flexibility. The interface can be individually set and it is possible to transfer data rates of up to 400.000 bits/sec. Furthermore, it is possible to additionally set emulations such as modem, AutoConnect, Conntect-On-Data, TCP/UDP-Client with up to 10 parallel connections, TCP/UDP server, E-Mail Client with sending and receiving of e-mails.

SPI

It is possible to use up to 2 independently operating SPI interfaces in the master or slave mode. Each interface can be individually set and it is possible to transfer data rates of up to 25.000.000 bits. Furthermore, it is possible to additionally set emulations such as AutoConnect, Conntect-On-Data, TCP/UDP-Client with up to 10 parallel connections, TCP/UDP server, E-Mail-Client with sending and receiving of e-mails.

By the setting in the Setup, the SPI interface can directly operate SD cards, DF cards or Data-Flash components. An implemented Flash-File system with FAT16 structure support the FTP server in order to save data on it or to read data from it. Now it is possible to save own homepages or JAVA – Applets in order to present a proper and individual look to the customers via the WEB server.

SD CARDS:

- Up to 4 GByte
- FAT16
- PC-compatible

DF cards / components:

- Up to 4 GByte
- FAT16
- AT45DB011B,AT45DB021B
  AT45DB041B,AT45DB081B
  AT45DB0161B,AT45DB0321B AT45DB0642, AT45DB1282
  will be directly identified.

The SPI interface can also directly control as SPI display, e.g. the EA DOGM162B-A, which you can directly use via TCP/IP.

TTL IO

It is possible to directly control up to 15 pins via two interfaces. To do so, there is a proper control mode which can read, switch on or switch off the signals. A tunnel mode allows the automatic transfer of the signal modes.

Ethernet

It can directly be used without any additional components. RJ45 Connector on board.

MDIX (Automatic cable recognition)
10 Half Duplex
10 Full Duplex
100 Half Duplex
100 Full Duplex
AutoSensing

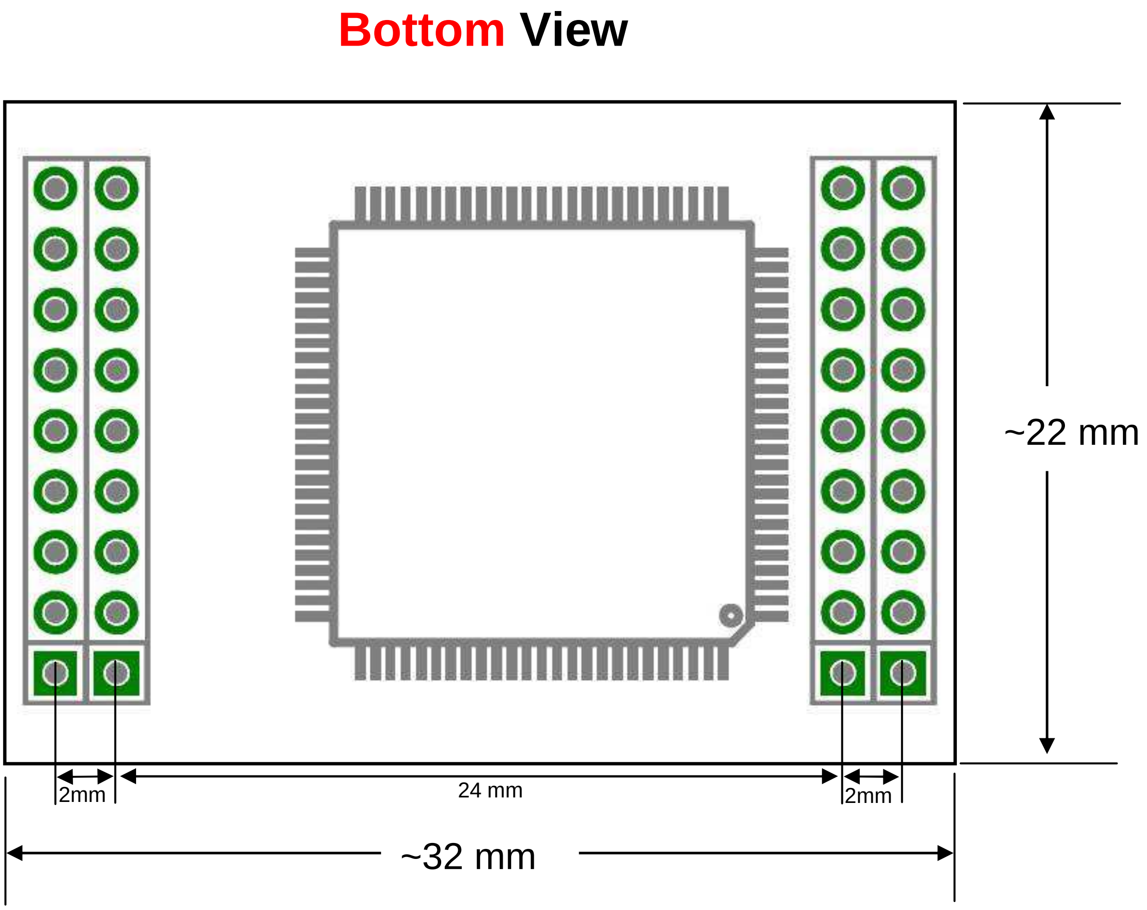
3 Power supply

- 3.3 volts +- 1% low noise, the supply voltage is directly used (0 Ohm)
- 170 mA

4 Network protocols

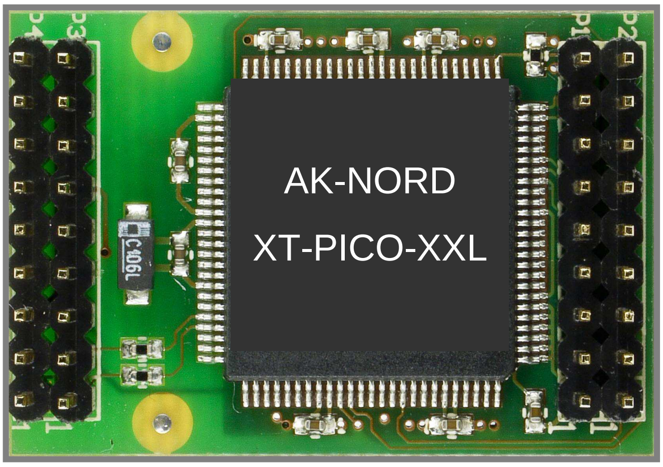
1. IP
2. TCP
3. UDP
4. FTP
5. TFTP
6. ICMP
7. ARP
8. SNMP
9. LPR
10. DHCP
11. BOOTP
12. DNS
13. TELNET
14. HTML
15. http
16. DYNDNS
17. SMTP
18. POP3
19. SYSLOG
20. SSL (in preparation)

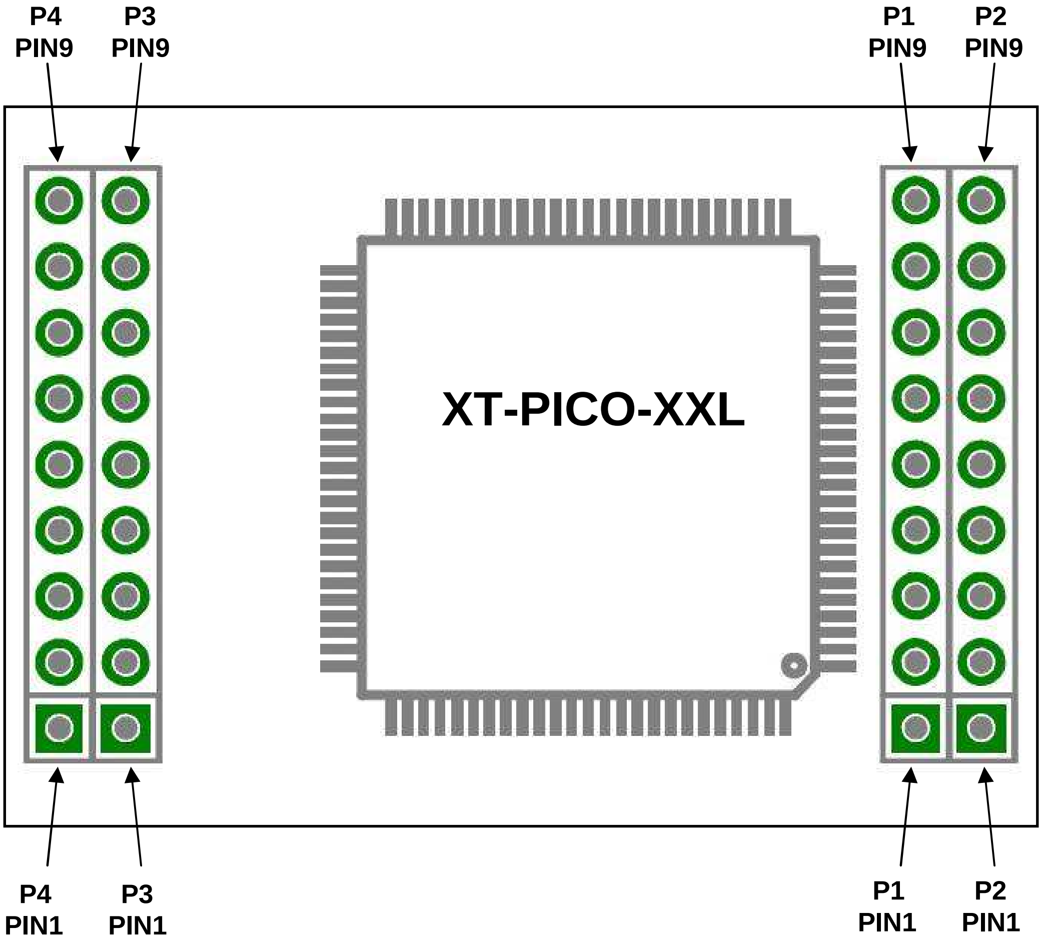
5 Dimensions

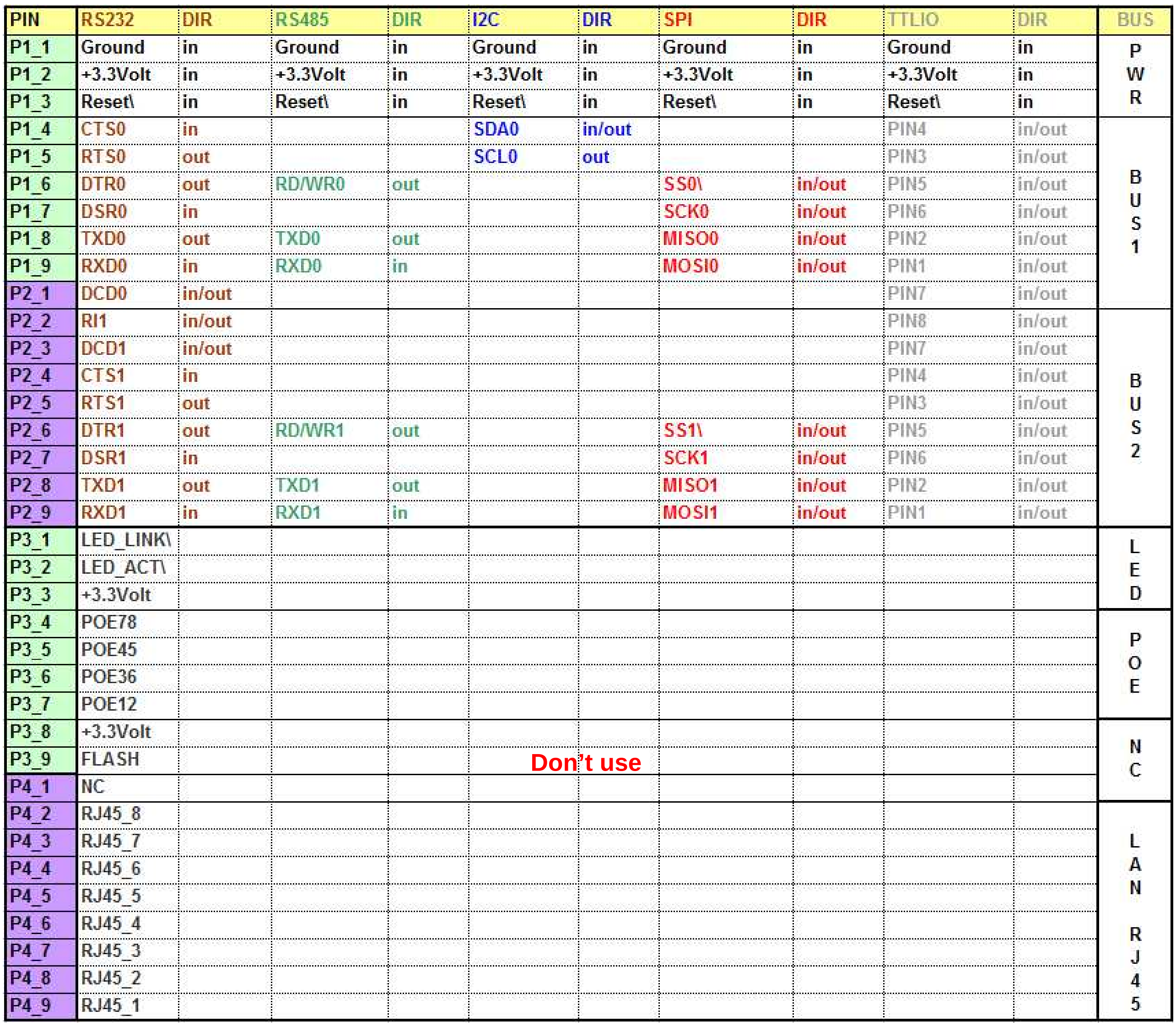


6 Pin description

Bottom View







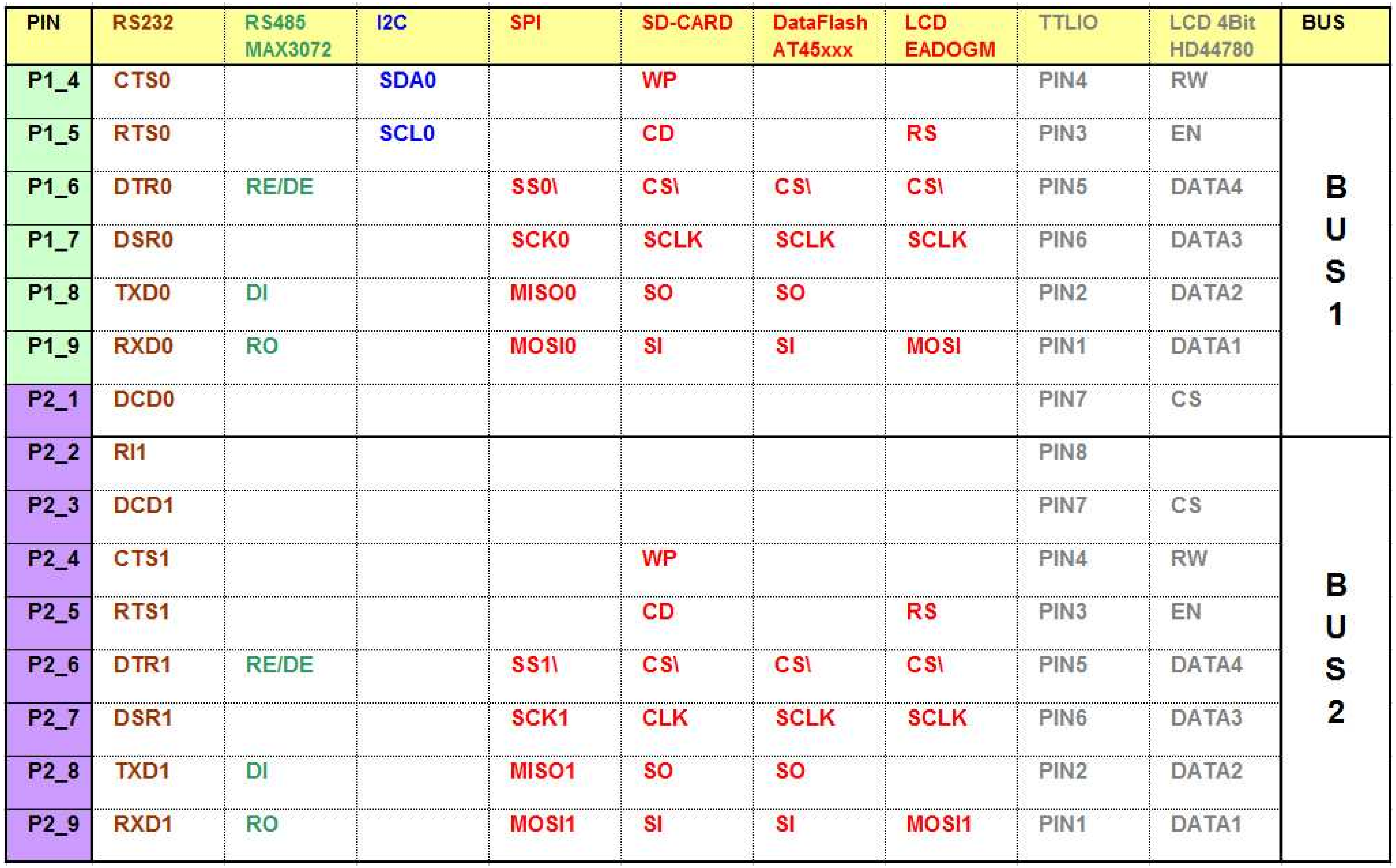
PIN	RS232	DIR	RS485	DIR	I2C	DIR	SPI	DIR	TTLIO	DIR	BUS
P1_1	Ground	in	Ground	in	Ground	in	Ground	in	Ground	in	PWR
P1_2	+3.3Volt	in	+3.3Volt	in	+3.3Volt	in	+3.3Volt	in	+3.3Volt	in	
P1_3	Reset\	in	Reset\	in	Reset\	in	Reset\	in	Reset\	in	
P1_4	CTS0	in			SDA0	in/out			PIN4	in/out	BUS1
P1_5	RTS0	out			SCL0	out			PIN3	in/out	
P1_6	DTR0	out	RD/WR0	out			SS0\	in/out	PIN5	in/out	
P1_7	DSR0	in					SCK0	in/out	PIN6	in/out	
P1_8	TXD0	out	TXD0	out			MISO0	in/out	PIN2	in/out	
P1_9	RXD0	in	RXD0	in			MOSI0	in/out	PIN1	in/out	
P2_1	DCD0	in/out							PIN7	in/out	
P2_2	RI1	in/out							PIN8	in/out	BUS2
P2_3	DCD1	in/out							PIN7	in/out	
P2_4	CTS1	in							PIN4	in/out	
P2_5	RTS1	out							PIN3	in/out	
P2_6	DTR1	out	RD/WR1	out			SS1\	in/out	PIN5	in/out	
P2_7	DSR1	in					SCK1	in/out	PIN6	in/out	
P2_8	TXD1	out	TXD1	out			MISO1	in/out	PIN2	in/out	
P2_9	RXD1	in	RXD1	in			MOSI1	in/out	PIN1	in/out	
P3_1	LED_LINK\										LED
P3_2	LED_ACT\										
P3_3	+3.3Volt										
P3_4	POE78										POE
P3_5	POE45										
P3_6	POE36										
P3_7	POE12										
P3_8	+3.3Volt										NC
P3_9	FLASH				Don't use						
P4_1	NC										
P4_2	RJ45_8										LAN RJ45
P4_3	RJ45_7										
P4_4	RJ45_6										
P4_5	RJ45_5										
P4_6	RJ45_4										
P4_7	RJ45_3										
P4_8	RJ45_2										
P4_9	RJ45_1										

Alles was Sie noch benötigen, ist eine RJ45 – Buchse und zwei LED's

Hinweis:

Die LED – Anschlüsse können direkt verwendet werden. Die Vorwiderstände sind schon auf dem **XT-PICO-XXL** vorhanden. Auch die RJ45 – Anschlüsse können direkt an eine RJ45 – Buchse angeschlossen werden. Der erforderliche Übertrager ist auf dem **XT-PICO-XXL** ebenfalls vorhanden.

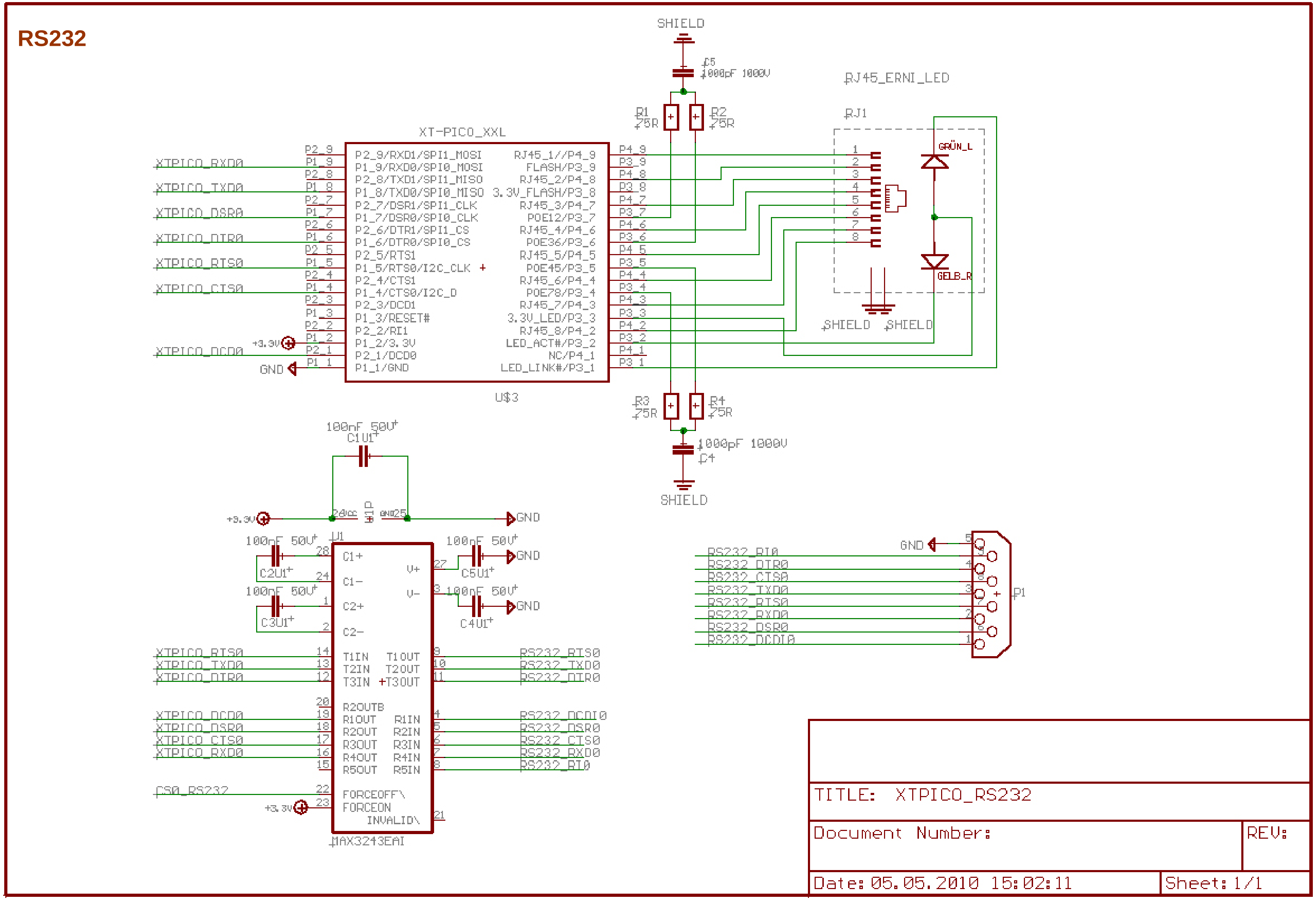
Connecting plan



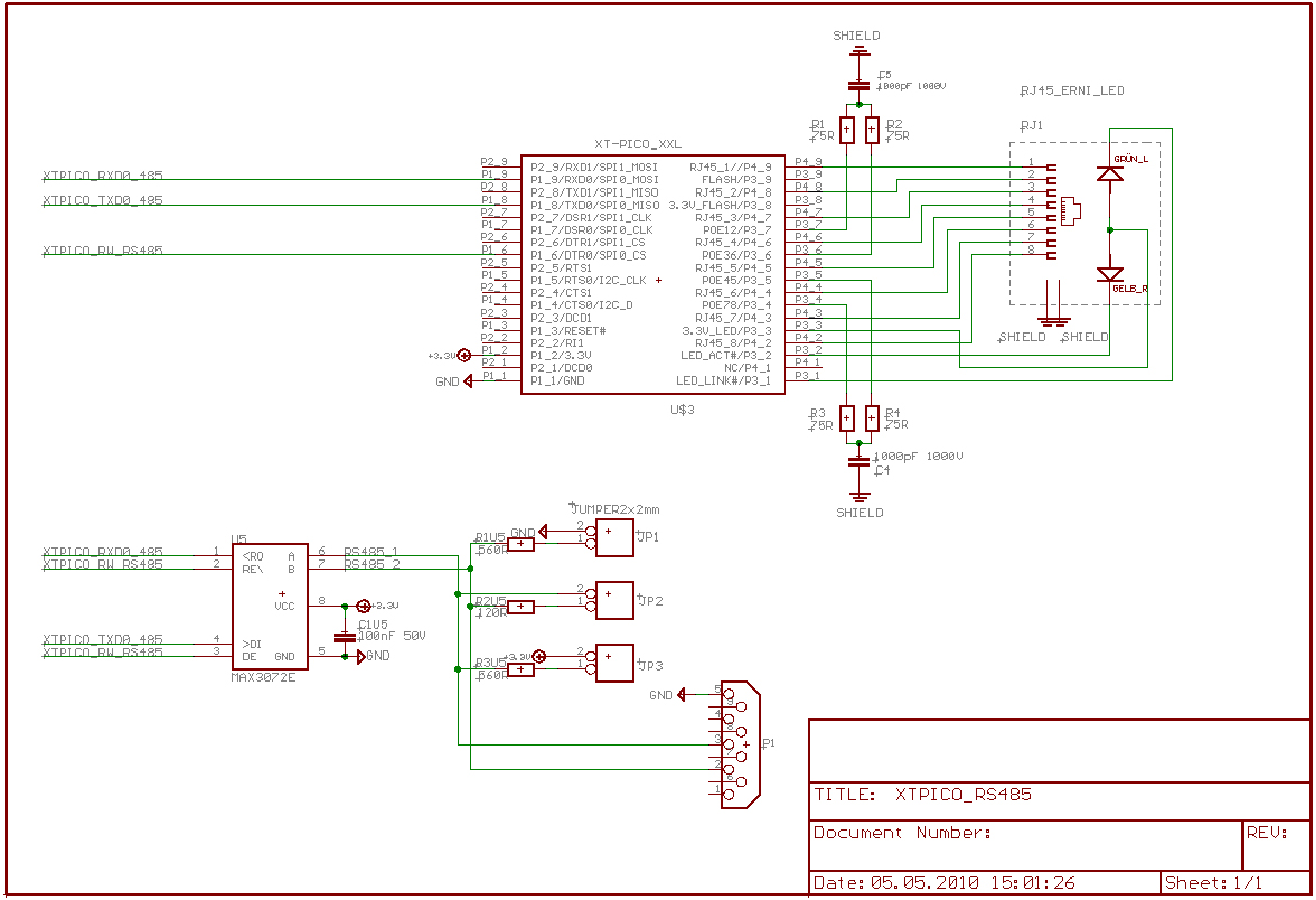
PIN	RS232	RS485 MAX3072	I2C	SPI	SD-CARD	DataFlash AT45xxx	LCD EADOGM	TTLIO	LCD 4Bit HD44780	BUS
P1_4	CTS0		SDA0		WP			PIN4	RW	BUS 1
P1_5	RTS0		SCL0		CD		RS	PIN3	EN	
P1_6	DTR0	RE/DE		SS0\	CS\	CS\	CS\	PIN5	DATA4	
P1_7	DSR0			SCK0	SCLK	SCLK	SCLK	PIN6	DATA3	
P1_8	TXD0	DI		MISO0	SO	SO		PIN2	DATA2	
P1_9	RXD0	RO		MOSI0	SI	SI	MOSI	PIN1	DATA1	
P2_1	DCD0							PIN7	CS	
P2_2	RI1							PIN8		BUS 2
P2_3	DCD1							PIN7	CS	
P2_4	CTS1				WP			PIN4	RW	
P2_5	RTS1				CD		RS	PIN3	EN	
P2_6	DTR1	RE/DE		SS1\	CS\	CS\	CS\	PIN5	DATA4	
P2_7	DSR1			SCK1	CLK	SCLK	SCLK	PIN6	DATA3	
P2_8	TXD1	DI		MISO1	SO	SO		PIN2	DATA2	
P2_9	RXD1	RO		MOSI1	SI	SI	MOSI1	PIN1	DATA1	

7 Schematics

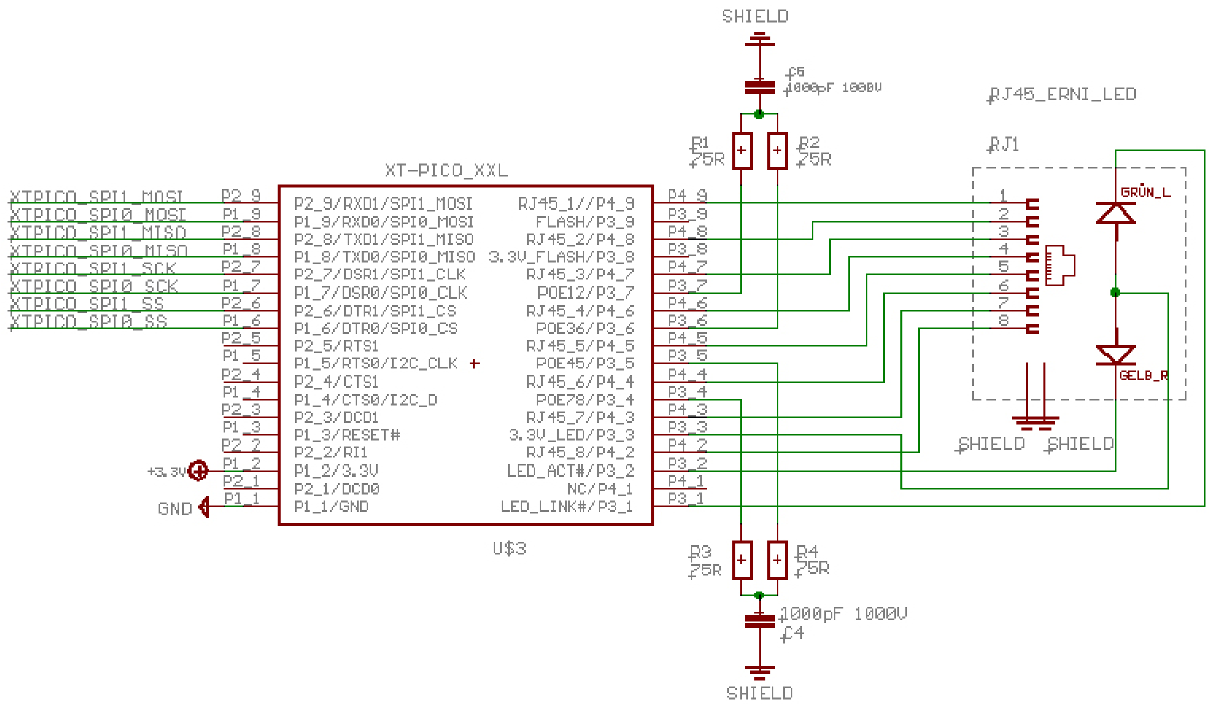
RS232



RS485


SHIELD
C5
1000pF 1000V
RJ45_ERNI_LED
RJ1
R1
75R
R2
75R
XT-PICO_XXL
XTPICO_RXD0_485
XTPICO_TXD0_485
XTPICO_RW_RS485
P2_9
P1_9
P2_8
P1_8
P2_7
P1_7
P2_6
P1_6
P2_5
P1_5
P2_4
P1_4
P2_3
P1_3
P2_2
P1_2
P2_1
P1_1
+3.3V
GND
P2_9/RXD1/SPI1_MOSI
P1_9/RXD0/SPI0_MOSI
P2_8/TXD1/SPI1_MISO
P1_8/TXD0/SPI0_MISO
P2_7/DSR1/SPI1_CLK
P1_7/DSR0/SPI0_CLK
P2_6/DTR1/SPI1_CS
P1_6/DTR0/SPI0_CS
P2_5/RTS1
P1_5/RTS0/I2C_CLK
P2_4/CTS1
P1_4/CTS0/I2C_D
P2_3/DCD1
P1_3/RESET#
P2_2/RI1
P1_2/3.3V
P2_1/DCD0
P1_1/GND
RJ45_1//P4_9
FLASH/P3_9
RJ45_2/P4_8
3.3V_FLASH/P3_8
RJ45_3/P4_7
POE12/P3_7
RJ45_4/P4_6
POE36/P3_6
RJ45_5/P4_5
POE45/P3_5
RJ45_6/P4_4
POE78/P3_4
RJ45_7/P4_3
3.3V_LED/P3_3
RJ45_8/P4_2
LED_ACT#/P3_2
NC/P4_1
LED_LINK#/P3_1
P4_9
P3_9
P4_8
P3_8
P4_7
P3_7
P4_6
P3_6
P4_5
P3_5
P4_4
P3_4
P4_3
P3_3
P4_2
P3_2
P4_1
P3_1
1
2
3
4
5
6
7
8
GRÜN_L
GELB_R
SHIELD
SHIELD
U$3
R3
75R
R4
75R
1000pF 1000V
C4
SHIELD
JUMPER2x2mm
GND
2
1
JP1
R1U5
560R
U5
XTPICO_RXD0_485
XTPICO_RW_RS485
1
2
<RO
RE\
A
B
6
7
RS485_1
RS485_2
VCC
8
+3.3V
C1U5
100nF 50V
XTPICO_TXD0_485
XTPICO_RW_RS485
4
3
>DI
DE
GND
5
GND
MAX3072E
2
1
JP2
R2U5
120R
R3U5
560R
+3.3V
2
1
JP3
GND
5
9
4
8
3
7
2
6
1
P1
TITLE: XTPICO_RS485
Document Number:
REV:
Date: 05.05.2010 15:01:26
Sheet: 1/1

SPI



TITLE: XTPICO_SPI

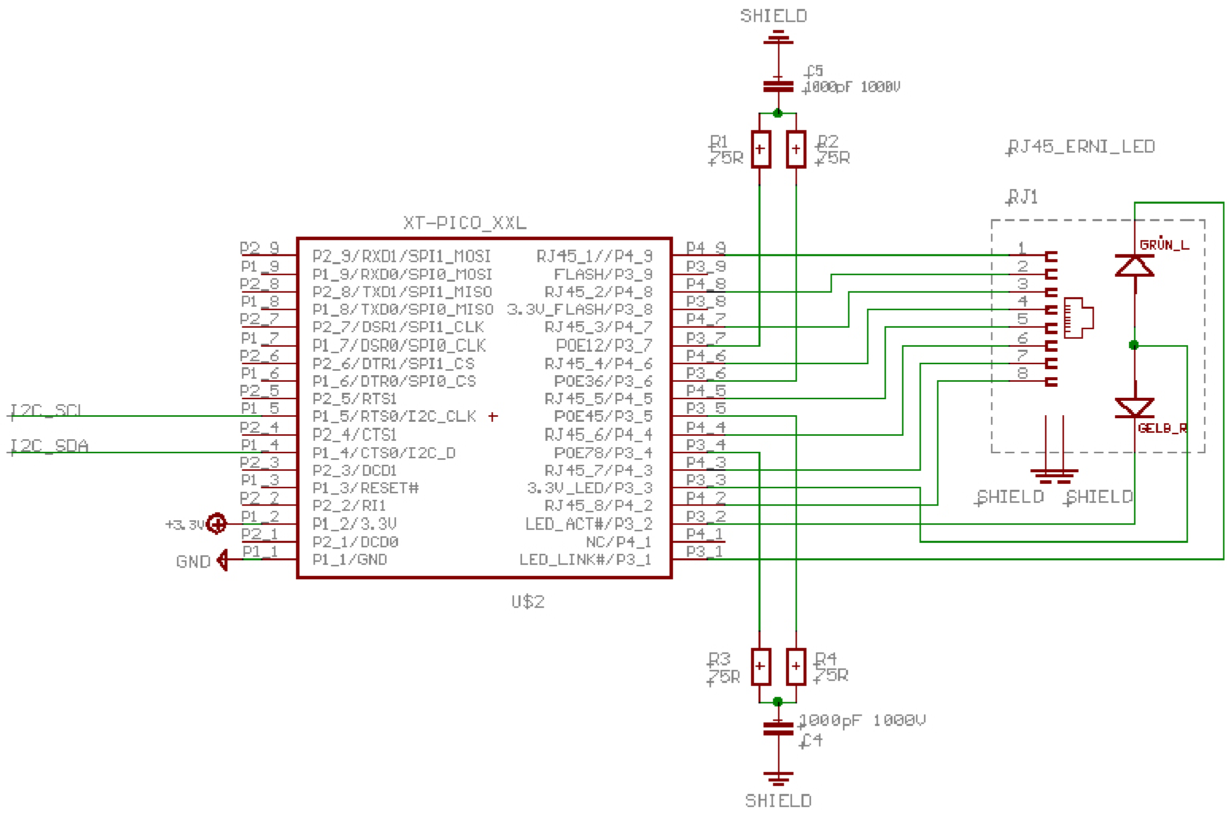
Document Number:

REV:

Date: 05.05.2010 15:00:16

Sheet: 1/1

I2C



TITLE: XTPICO_I2C

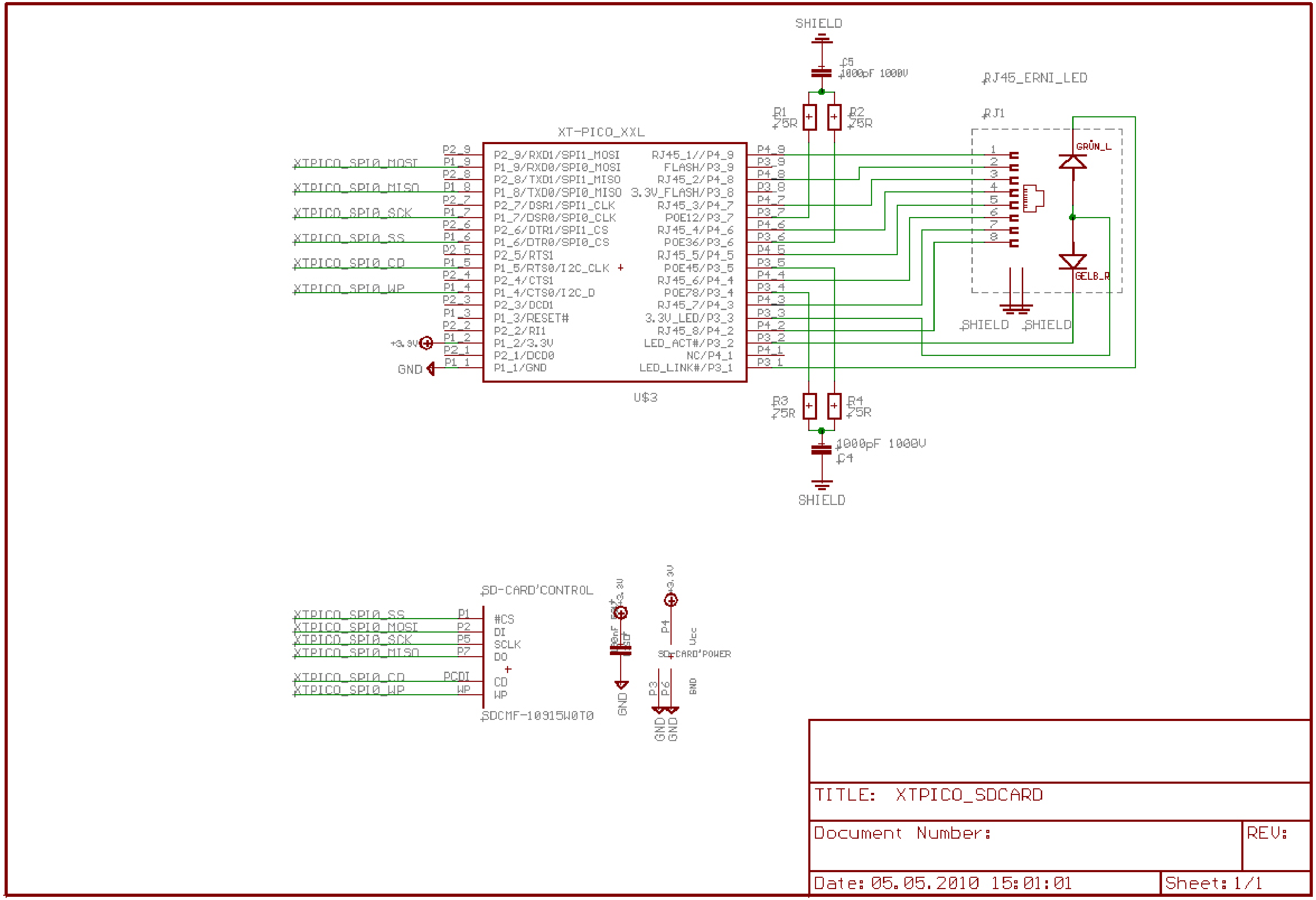
Document Number:

REV:

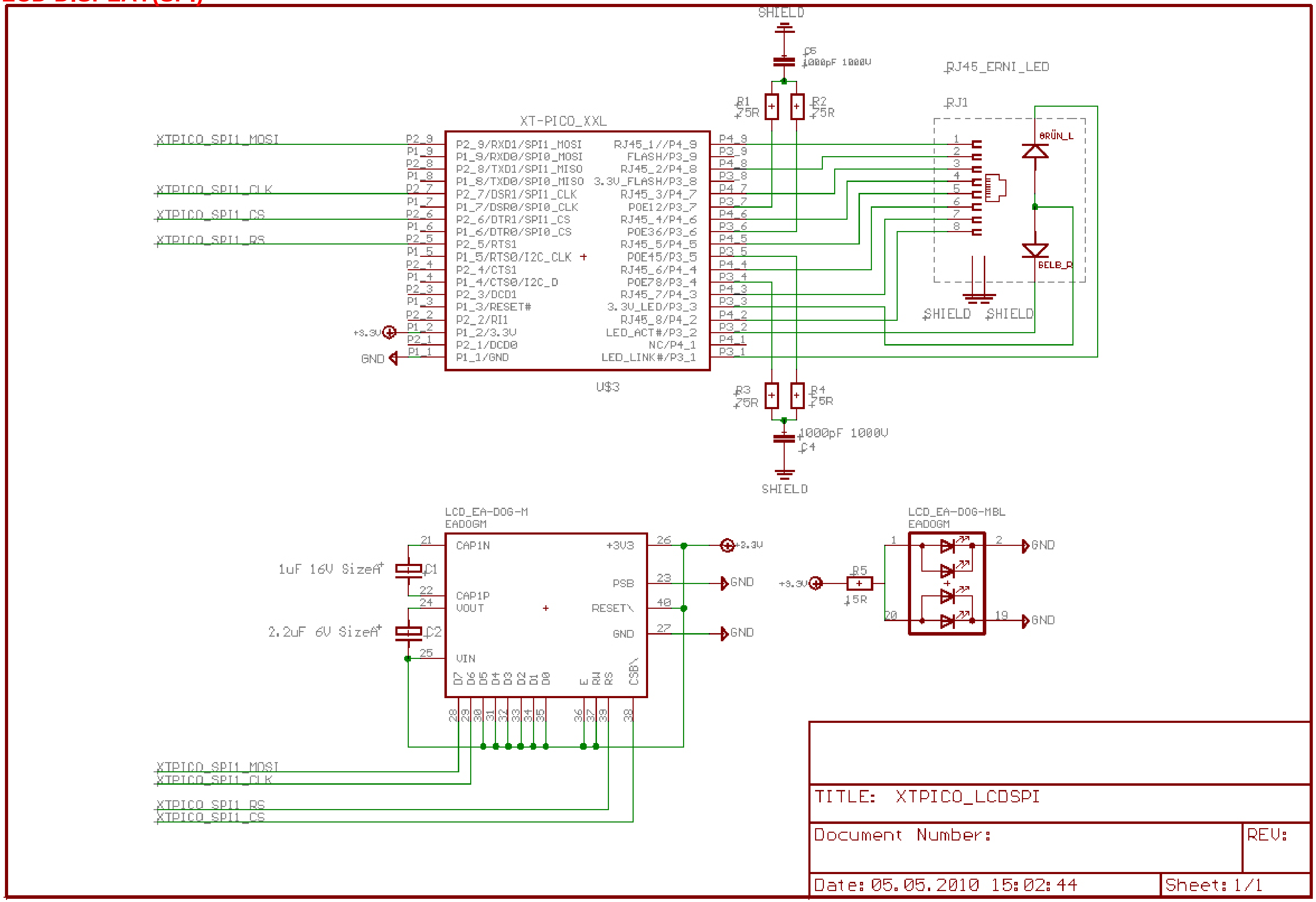
Date: 05.05.2010 14:19:07

Sheet: 1/1

SD CARD


SHIELD
C5
1000pF 1000V
RJ45_ERNI_LED
RJ1
R1
75R
R2
75R
XT-PICO_XXL
XTPICO_SPI0_MOSI
XTPICO_SPI0_MISO
XTPICO_SPI0_SCK
XTPICO_SPI0_SS
XTPICO_SPI0_CD
XTPICO_SPI0_WP
+3.3V
GND
P2_9/RXD1/SPI1_MOSI
P1_9/RXD0/SPI0_MOSI
P2_8/TXD1/SPI1_MISO
P1_8/TXD0/SPI0_MISO
P2_7/DSR1/SPI1_CLK
P1_7/DSR0/SPI0_CLK
P2_6/DTR1/SPI1_CS
P1_6/DTR0/SPI0_CS
P2_5/RTS1
P1_5/RTS0/I2C_CLK
P2_4/CTS1
P1_4/CTS0/I2C_D
P2_3/DCD1
P1_3/RESET#
P2_2/RI1
P1_2/3.3V
P2_1/DCD0
P1_1/GND
RJ45_1//P4_9
FLASH/P3_9
RJ45_2/P4_8
3.3V_FLASH/P3_8
RJ45_3/P4_7
POE12/P3_7
RJ45_4/P4_6
POE36/P3_6
RJ45_5/P4_5
POE45/P3_5
RJ45_6/P4_4
POE78/P3_4
RJ45_7/P4_3
3.3V_LED/P3_3
RJ45_8/P4_2
LED_ACT#/P3_2
NC/P4_1
LED_LINK#/P3_1
U$3
GRÜN_L
GELB_R
SHIELD
SHIELD
R3
75R
R4
75R
1000pF 1000V
C4
SHIELD
SD-CARD'CONTROL
XTPICO_SPI0_SS
XTPICO_SPI0_MOSI
XTPICO_SPI0_SCK
XTPICO_SPI0_MISO
XTPICO_SPI0_CD
XTPICO_SPI0_WP
P1
P2
P5
P7
PCDI
WP
#CS
DI
SCLK
DO
CD
WP
SDCMF-10915W0T0
+3.3V
100nF
GND
+3.3V
P4
Vcc
SD-CARD'POWER
P3
P6
GND
GND
GND
TITLE: XTPICO_SDCARD
Document Number:
REV:
Date: 05.05.2010 15:01:01
Sheet: 1/1

LCD DISPLAY(SPI)



TITLE: XTPICO_LCDSPI

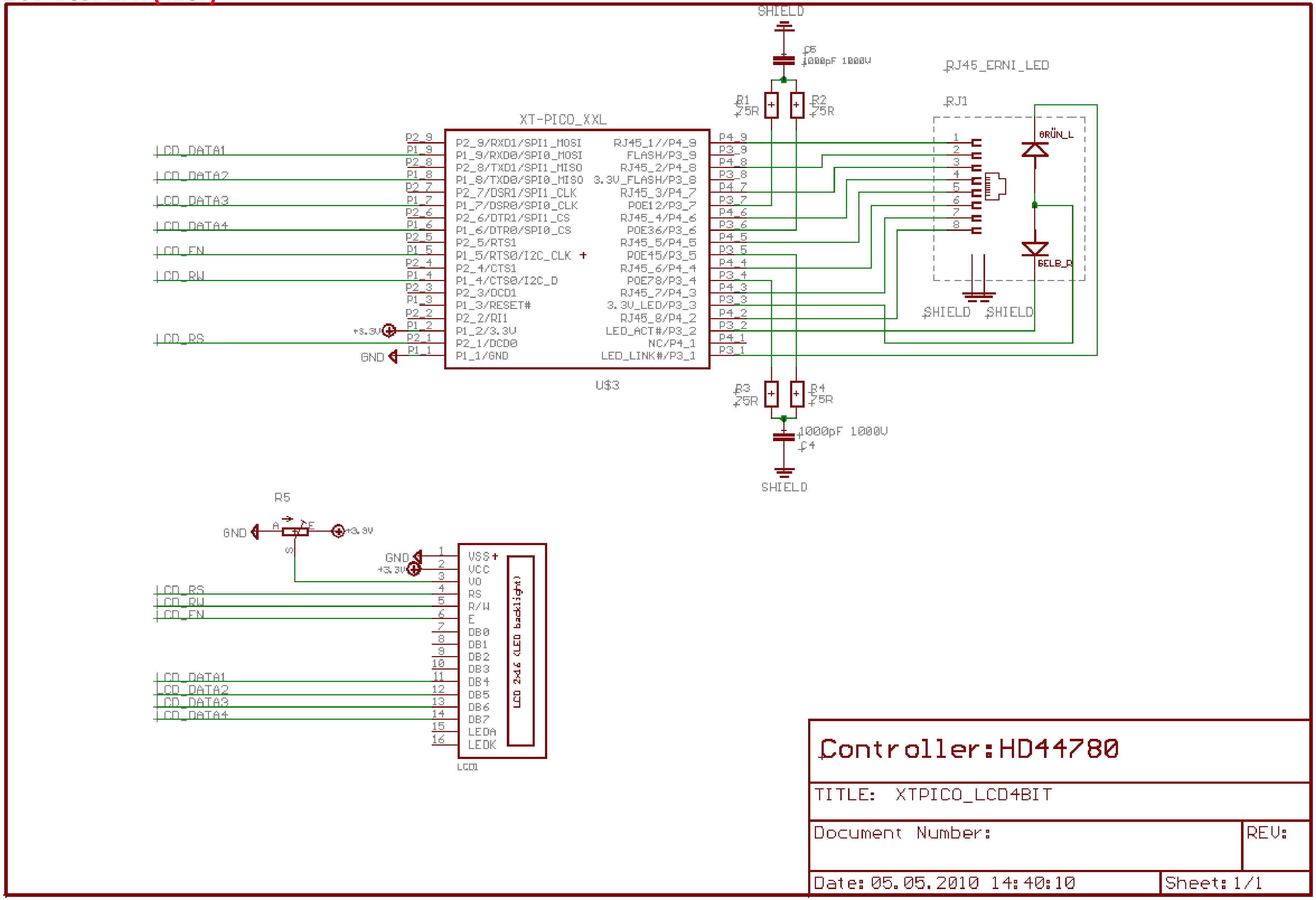
Document Number:

REV:

Date: 05.05.2010 15:02:44

Sheet: 1/1

LCD DISPLAY(4BIT)



Controller:HD44780

TITLE: XTPICO_LCD4BIT

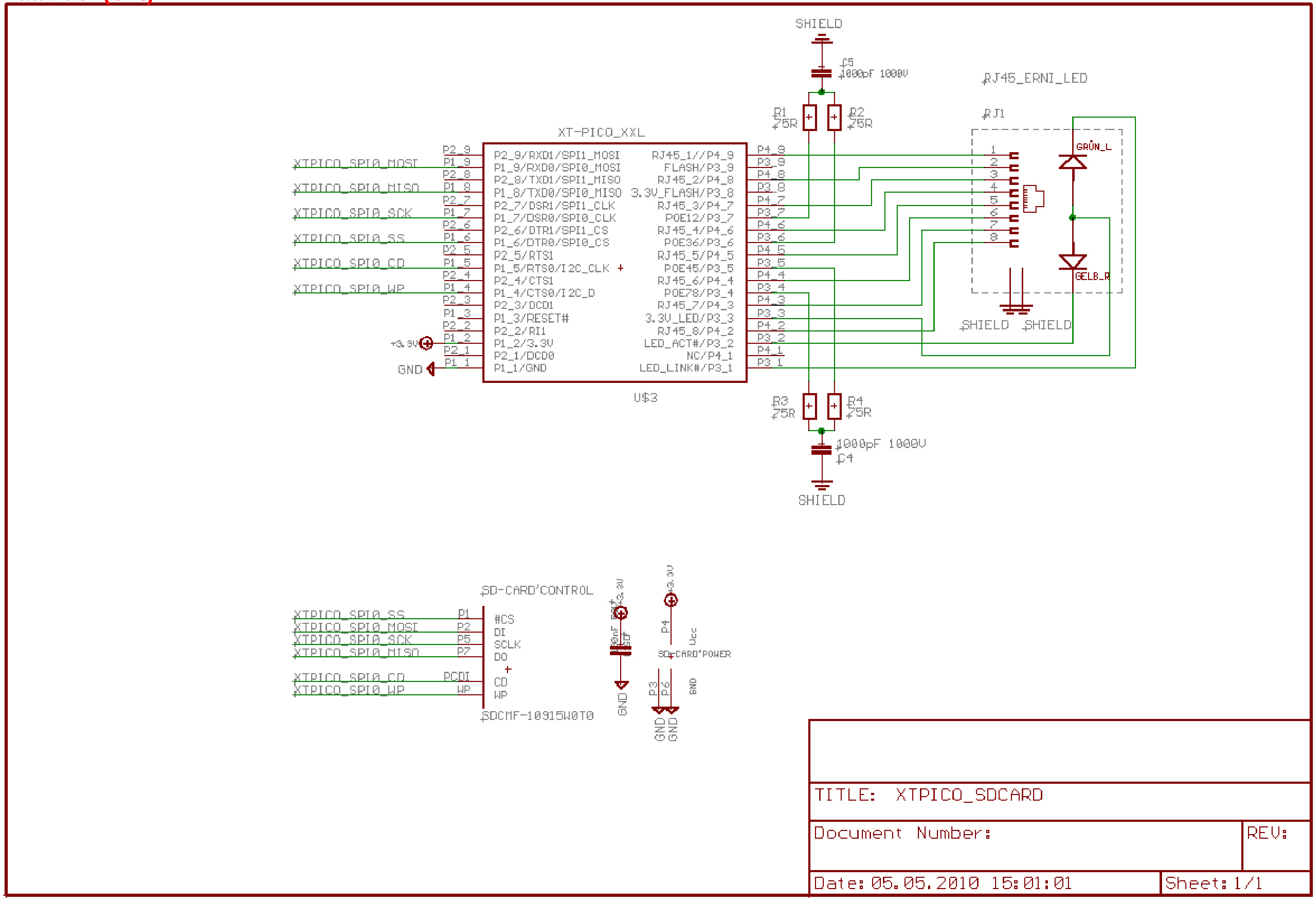
Document Number:

REV:

Date: 05.05.2010 14:40:10

Sheet: 1/1

Dataflash(SPI)



TITLE: XTPICO_SDCARD

Document Number:

REV:

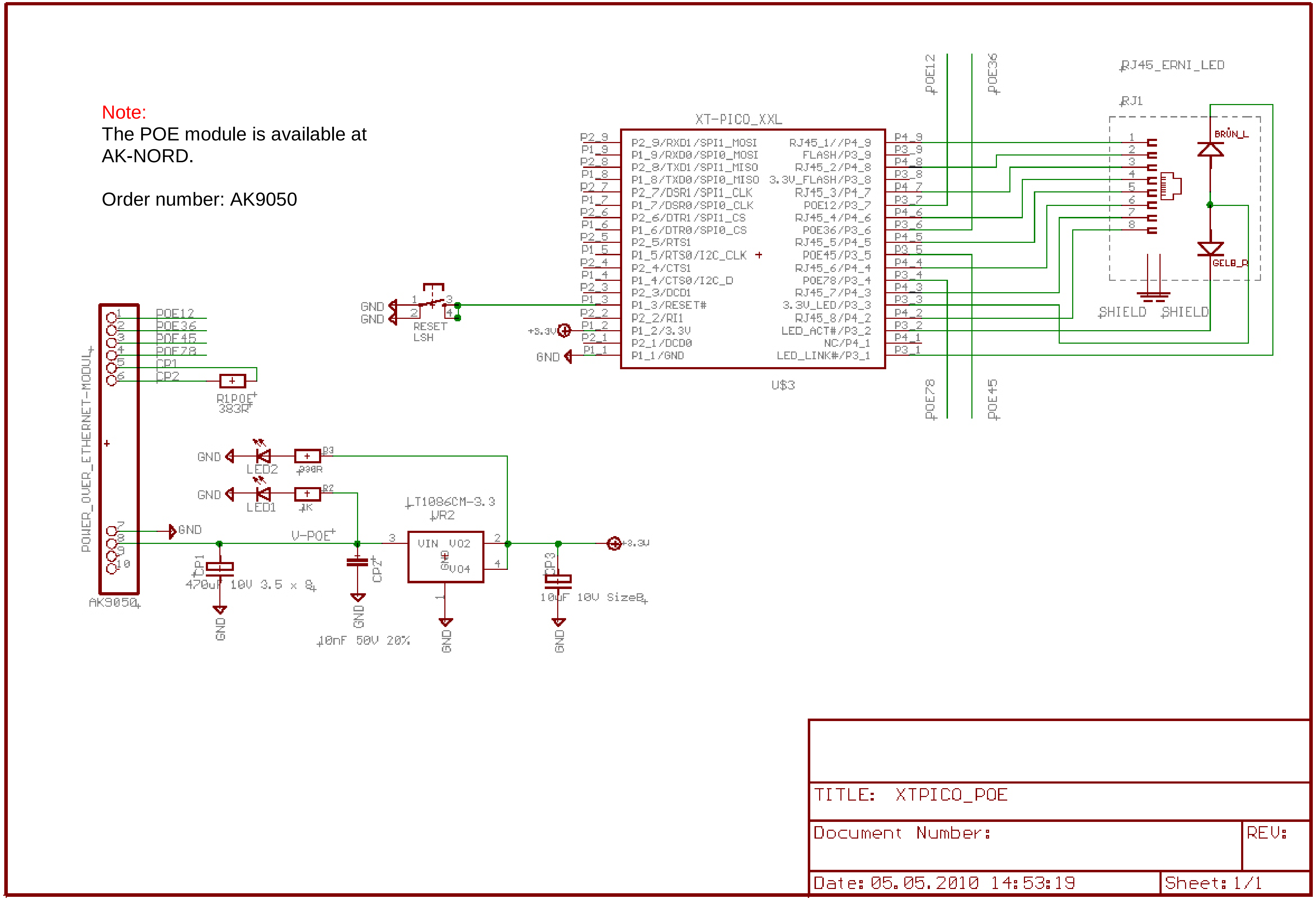
Date: 05.05.2010 15:01:01

Sheet: 1/1

POE (PowerOverEthernet)

Note:
The POE module is available at AK-NORD.

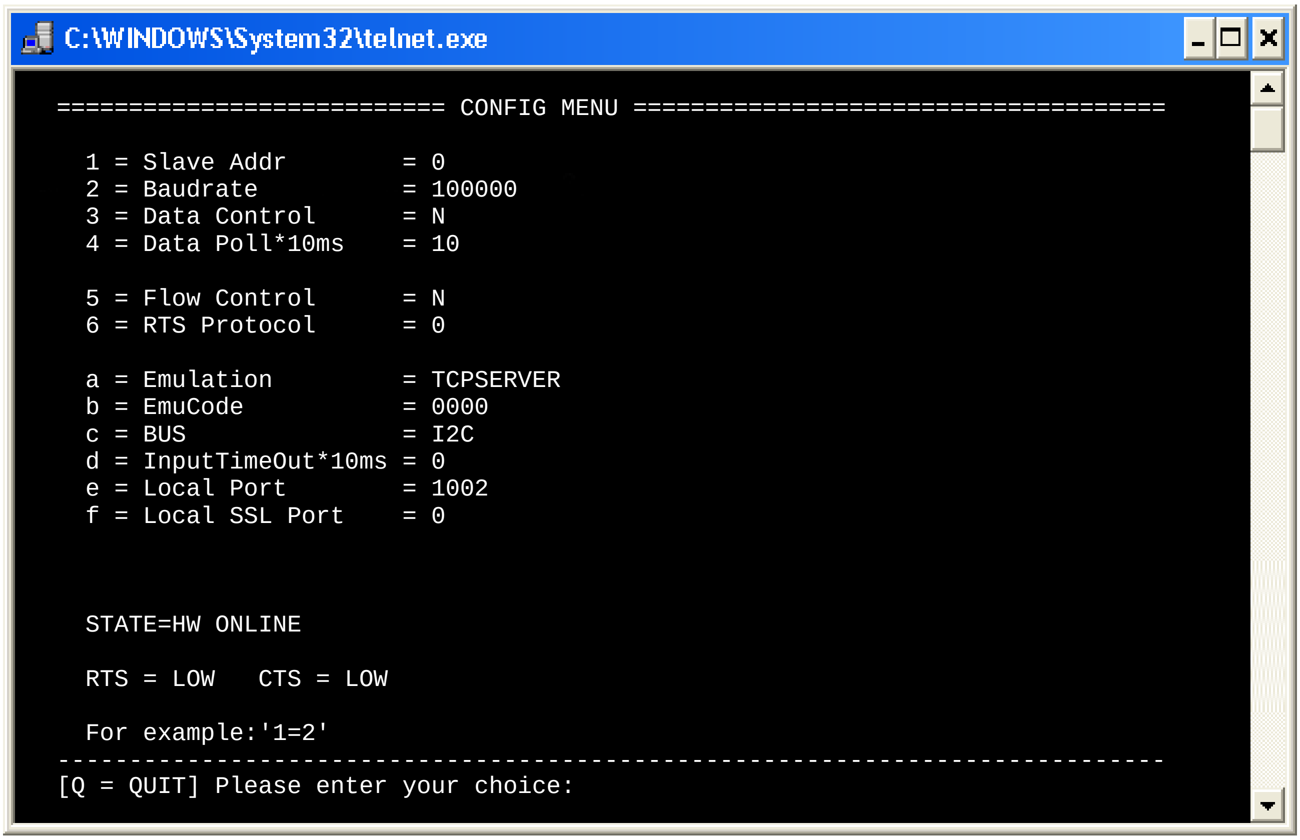
Order number: AK9050



8 BUS = I2C

Inter-Integrated Circuit (IIC)
The inter-IC bus (IIC) is a two-wire, bidirectional serial bus that provides a simple, efficient method of data exchange between devices. Being a two-wire device, the IIC bus minimizes the need for large numbers of connections between devices, and eliminates the need for an address decoder.

SETUP I2C



```
========================== CONFIG MENU ====================================

  1 = Slave Addr          = 0
  2 = Baudrate            = 100000
  3 = Data Control        = N
  4 = Data Poll*10ms      = 10

  5 = Flow Control        = N
  6 = RTS Protocol        = 0

  a = Emulation           = TCPSERVER
  b = EmuCode             = 0000
  c = BUS                 = I2C
  d = InputTimeOut*10ms   = 0
  e = Local Port          = 1002
  f = Local SSL Port      = 0



  STATE=HW ONLINE

  RTS = LOW   CTS = LOW

  For example:'1=2'
-----------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

`1 = Slave Addr` = Address of slave. Possible values 0-127

`2 = Baudrate` = Baud rate on the I2C bus. Possible values 1-400000

`3 = Data Control` = N = no data control (please refer to next page)
D = protocol mode (please refer to next page)

`4 = Data Poll*10ms` = only if DataControl=N is active. Then every xxx ms it is controlled, if there are data available on the Slave.

`5 = Flow Control` = No function

`6 = RTS Protocol` = No function

`a=Emulation:` = The following emulations can be used:
MODEM,EMAIL,TCPCLIENT,UDPCLIENT,UDPSERVER,TCPSERVER
Please find examples of applications and detailed explanations regarding emulations in the main manual of the AK XXL products.

`b=Emucode:` = for customer-specific functions

`c=BUS` = **I2C**
Each bus can be reconfigured by entering an admissible value as for example "RS232, RS485, I2C, SPI, LCD, SDCARD or DFCARD" in the menu item "BUS". However, it is only possible to activate and configure the selected BUS after resetting.

`e = Local Port:` = Here you can define the TCP/IP or UDP port, which is approved for the corresponding serial interface.
If you enter here 23 (Telnet) or 80 (Browser), the device can no longer be configured via this service. But the output of the data (requests) of browser or Telnet will be performed directly via the serial interface.

`f = Local SSL Port:` = Has got the same function as the Local Port, but it is used for encoded SSL communication.

All other menus are described in the main manual of the AK XXL products.

Standard mode:

Settings:

```
1 = Slave Addr         = 0
2 = Baudrate           = 100000
3 = Data Control       = N
4 = Data Poll*10ms     = 10
5 = Flow Control       = N
```

In this setting the XT-PICO module polls the I2C-Slave on the address **0** every **100ms** and tries to obtain data from it. Depending on the other setting such as TCP-Client, modem etc., such data is being processed. As soon as a connection is available via TCP/IP as soon as data are received, it will try to send the data to the I2C-Slave to the address **0**.

Protocol mode:

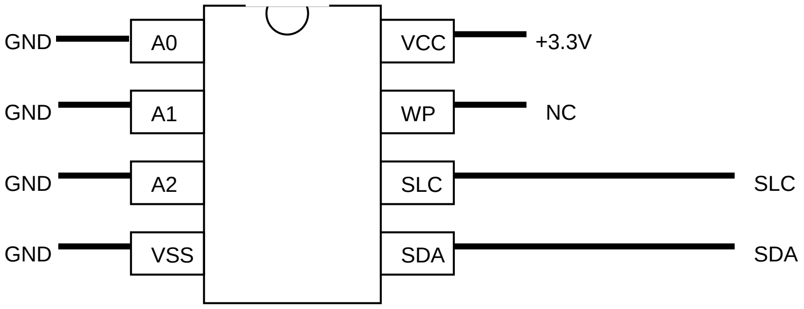
Settings:

```
1 = Slave Addr         = 0
2 = Baudrate           = 400000
3 = Data Control       = P
4 = Data Poll*10ms     = 10
5 = Flow Control       = N
```

Using this setting it will be possible to directly access the interface via TCP/IP and to directly address the I2C-Slave by means of corresponding commands such as WRITE and READ.

```
WRITE:      0x02,0x00,0x0A,0x03,0x50,0x02,0x00,0x00,0x57,0x00,0x01,0xnn,0x03
            0x02            STX
            0x00,0x0A       Len (10 Bytes) follows              always 2 Byte
            0x03            function code (with all messages)
            0x50            Slave Address
            0x02            Count Internal Address
            0x00,0x00       Internal Address 0x00,0x00          Count(0-4 Byte)
            0x57            W = WRITE
            0x00,0x01       write 1 byte                        always 2 Byte
            0xnn            Byte to write
            0x03            ETX

READ:       0x02,0x00,0x09,0x03,0x50,0x02,0x00,0x00,0x52,0x00,0x02,0x03
            0x02            STX
            0x00,0x09       Len (9 Bytes) follows               always 2 Byte
            0x03            function code (with all messages)
            0x50            Slave Address
            0x02            Count Internal Address
            0x00,0x00       Internal Address 0x00,0x00          Count(0-4 Byte)
            0x52            R = READ
            0x00,0x02       read 2 byte                         always 2 Byte
            0x03            ETX


MSG:        no ACK/NAK  (function code = 0x00)
            NAK only    (function code = 0x01)
            ACK only    (function code = 0x02)
            ACK and NAK (function code = 0x03)

Answers:

NAK         0x15,'S'        NAK STX
            0x15,'E'        NAK ETX
            0x15,'A'        NAK Slave Address
            0x15,'C'        NAK Command
            0x15,'L'        NAK Len
            0x15,'B'        NAK Buffer
            0x15,'R',..     NAK Read and Data we could read
            0x15,'W',nn,nn  NAK Write and nn,nn = Data we could write

ACK         0x06,'R'...     ACK Read and Data
            0x06,'W'        ACK Write
```

Example: EEPROM 24LC16



Settings:

```
1 = Slave Addr         = 0
2 = Baudrate           = 400000
3 = Data Control       = P
4 = Data Poll*10ms     = 10
5 = Flow Control       = N
```

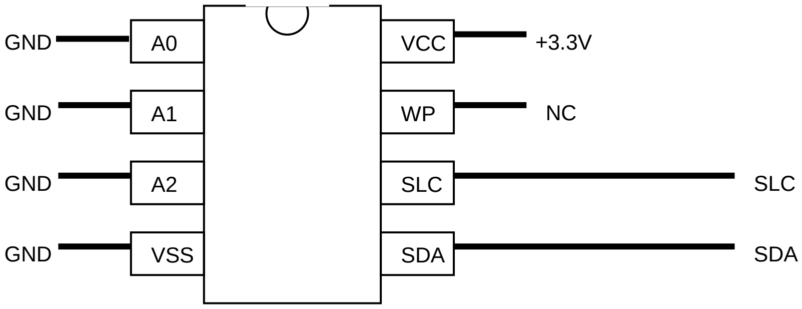
Write “HALLO”

```
WRITE:      0x02,0x00,0x0D,0x03,0x50,0x01,0x00,0x57,0x00,0x05,0x48,0x41,0x4C,
            0x4C,0x4F,0x03
ANSWER:     0x06,0x57
```

Read “HALLO”

```
WRITE:      0x02,0x00,0x08,0x03,0x50,0x01,0x00,0x52,0x00,0x05,0x03
ANSWER:     0x06,0x52,0x48,0x41,0x4C,0x4C,0x4F
```

Example: EEPROM AT24C16



Settings:

```
1 = Slave Addr         = 0
2 = Baudrate           = 400000
3 = Data Control       = P
4 = Data Poll*10ms     = 10
5 = Flow Control       = N
```

Write “HALLO”

```
WRITE:      0x02,0x00,0x0E,0x03,0x50,0x02,0x00,0x00,0x57,0x00,0x05,0x48,0x41,
            0x4C,0x4C,0x4F,0x03
ANSWER:     0x06,0x57
```

Read “HALLO”

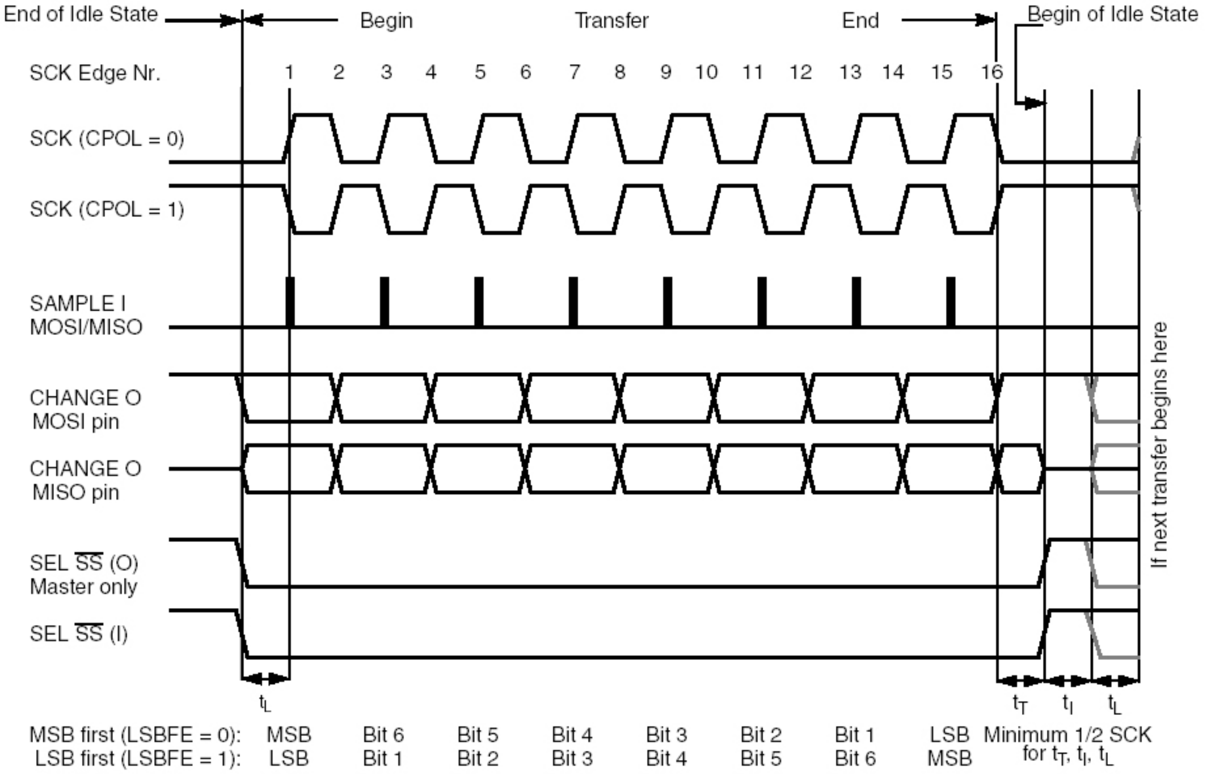
```
WRITE:      0x02,0x00,0x09,0x03,0x50,0x02,0x00,0x00,0x52,0x00,0x05,0x03
ANSWER:     0x06,0x52,0x48,0x41,0x4C,0x4C,0x4F
```

9 BUS = SPI

Serial Peripheral Interface (SPI)

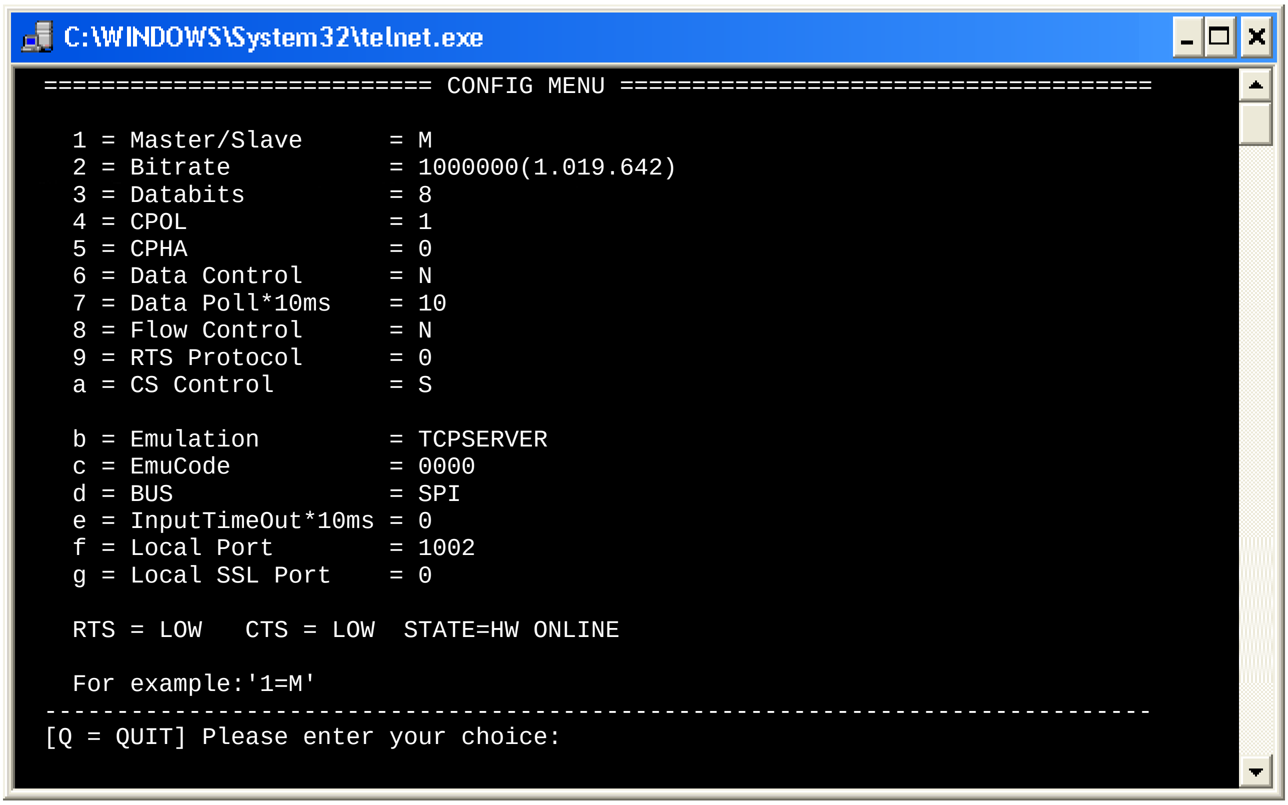
The SPI-bus is a 4-wire serial communications interface which is used by many microprocessor peripheral chips. The Serial Peripheral Interface (SPI) circuit is a synchronous serial data link that is standard for many microprocessors and other peripheral chips. It provides support for a bandwidth (**25MBit**) network connection amongst CPUs and other devices supporting the SPI.

SPI bus is basically a relatively simple synchronous serial interface which is used to connect low speed external devices using a quite little number of wires. SPI (serial peripheral interface) is an interface standard defined by Motorola. A synchronous clock shifts serial data into and out of the microcontrollers in blocks of 8 bits.



t_L = Minimum leading time before the first SCK edge
t_T = Minimum trailing time after the last SCK edge
t_I = Minimum idling time between transfers (minimum $\overline{SS}$ high time)

SETUP SPI



```
 =========================== CONFIG MENU ====================================

   1 = Master/Slave      = M
   2 = Bitrate           = 1000000(1.019.642)
   3 = Databits          = 8
   4 = CPOL              = 1
   5 = CPHA              = 0
   6 = Data Control      = N
   7 = Data Poll*10ms    = 10
   8 = Flow Control      = N
   9 = RTS Protocol      = 0
   a = CS Control        = S

   b = Emulation         = TCPSERVER
   c = EmuCode           = 0000
   d = BUS               = SPI
   e = InputTimeOut*10ms = 0
   f = Local Port        = 1002
   g = Local SSL Port    = 0

   RTS = LOW   CTS = LOW  STATE=HW ONLINE

   For example:'1=M'
 -----------------------------------------------------------------------------
 [Q = QUIT] Please enter your choice:
```

`1 = Master/Slave` = Mode
M = MASTER
S = SLAVE

`2 = Bit rate` = Cycle clock on the SPI bus. Possible values 1-25.000.000Bit/s

`3 = Databits` = Possible values 8-16

`4 = CPOL` = Clock Polarity possible values 0,1

`5 = CPHA` = Clock Phase possible values 0,1

`6 = Data Control` = N = no data control (please refer to next page)
P = protocol mode (please refer to next page)
D = Downstream. Protocol modes without return data.

`7 = Data Poll*10ms` = Only if DataControl=N is active. Then the slave is controlled every xxx ms if it has got data.

`8 = Flow Control` = Additional flow control via RTS/CTS hardware lines.
N = no flow control
H = hardware flow control.

`9 = RTS Protocol` = Signal statuses of RTS
0 = ALWAYS = RTS always low
1 = RTS_FOLLOWS_CTS = RTS follows CTS
3 = RTS_IND_CONNECTION_N = Shows a TCP/IP connection (LOW)
4 = RTS_IND_CONNECTION_I = Shows a TCP/IP connection (HIGH)

`a = CS Control` = Functionality of CS-Line in the master mode
S = CS = Low at start of the data stream.
B = CS = Low at every byte.

`a=Emulation:` = The following emulations can be used:
MODEM,EMAIL,TCPCLIENT,UDPCLIENT,UDPSERVER,TCPSERVER
Please find examples of applications and detailed explanations regarding emulations in the main manual of the AK XXL products.

`b=Emucode:` = for customer-specific functions

`c=BUS` = SPI
Each bus can be reconfigured by entering an admissible value as for example “RS232, RS485, I2C, SPI, LCD, SDCARD or DFCARD“ in the menu item “BUS”. However it is only possible to activate and configure the selected BUS after resetting.

`e = Local Port:` = Here you can define the TCP/IP or UDP port, which is approved for the corresponding serial interface.
If you enter here 23 (Telnet) or 80 (Browser), the device can no longer be configured via this service. But the output of the data (requests) of browser or Telnet will be performed directly via the serial interface.

`f = Local SSL Port:=` Has got the same function as the Local Port, but it is used for encoded SSL communication

All other menus are described in the main manual of the AK XXL products.

Standard mode:

Settings:

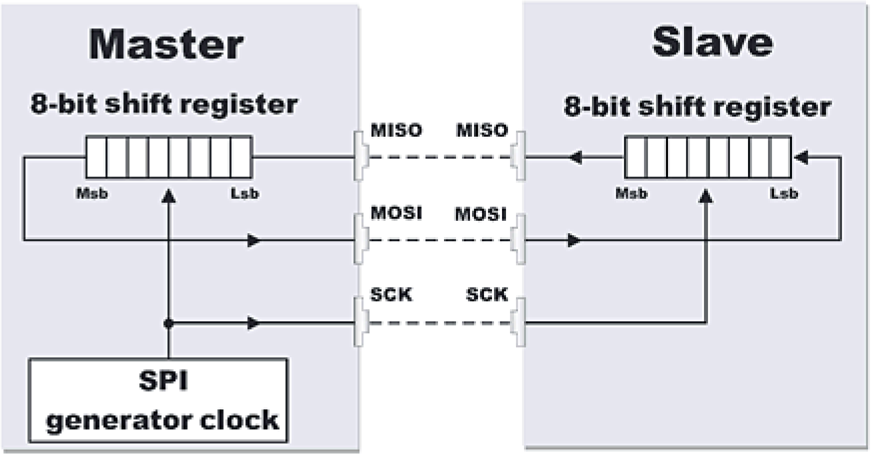
```
  1 = Master/Slave      = M
  2 = Bitrate           = 1000000(1.019.642)
  3 = Databits          = 8
  4 = CPOL              = 1
  5 = CPHA              = 0
  6 = Data Control      = N
  7 = Data Poll*10ms    = 10
  8 = Flow Control      = N
  9 = RTS Protocol      = 0
```

In this mode the SPI interface is absolutely transparent. I.e. no data will be sent to the SPI-Slave as long as no data are being received. The user has got the option to control the data flow by himself. Therefore, in order to receive data from the SPI-Slave, there must first be a connection available via the TCP/IP and it is necessary to send data from the Client (Remote). Such data will then be directly sent to the SPI–Slave.

Example:
There is a TCP/IP connection available and the SPI-Slave has to return on byte "0x33" in response to a 3 byte query e.g. "0x01, 0x00, 0x33".

Remote	XT-PICO	SPI-Slave
0x01, 0x00, 0x33 ->	0x01, 0x00, 0x33 ->	->
0xnn, 0xnn, 0xnn <-	0xnn, 0xnn, 0xnn <-	<-
0xnn ->	0xnn ->	->
0x33 <-	0x33 <-	<-
or (if the SPI–Slave is rapid enough)		
0x01, 0x00, 0x33, 0xnn->	0x01, 0x00, 0x33, 0xnn ->	->
0xnn, 0xnn, 0xnn ,0x33 <-	0xnn, 0xnn, 0xnn, 0x33 <-	<-

The data which are marked by "nn" do not include any relevant information. However, it is necessary to keep in mind that one bit is being received for each bit which is leaving the Master. (Refer to illustration)



Protocol mode:

Settings:

```
  1 = Master/Slave       = M
  2 = Bitrate            = 1000000(1.019.642)
  3 = Databits           = 8
  4 = CPOL               = 1
  5 = CPHA               = 0
  6 = Data Control       = P
  7 = Data Poll*10ms     = 10
  8 = Flow Control       = N
  9 = RTS Protocol       = 0
```

Using these settings, the XT-PICO module polls the SPI-Slave every **100ms** by sending a 0x00 and tries to obtain data from it. The slave as well as the master has the option to separate the data from non-relevant data "0xnn". Each data byte which is sent or received needs to be preceded by 0x01 in order to signalize that it is a correct data byte.

Example:

	Master	Slave	Data
Idle running	0x00 ->	<- 0x00	none
Master is sending	0x01,0x33 ->	<- 0x00, 0x00	Slave is receiving 0x33
Slave is sending	0x00,0x00 ->	<- 0x01, 0x44	Master is receiving 0x44
Master+Slave are sending	0x01,0x33,0x01,0x33 ->	<- 0x01, 0x44, 0x01, 0x44	

Note:

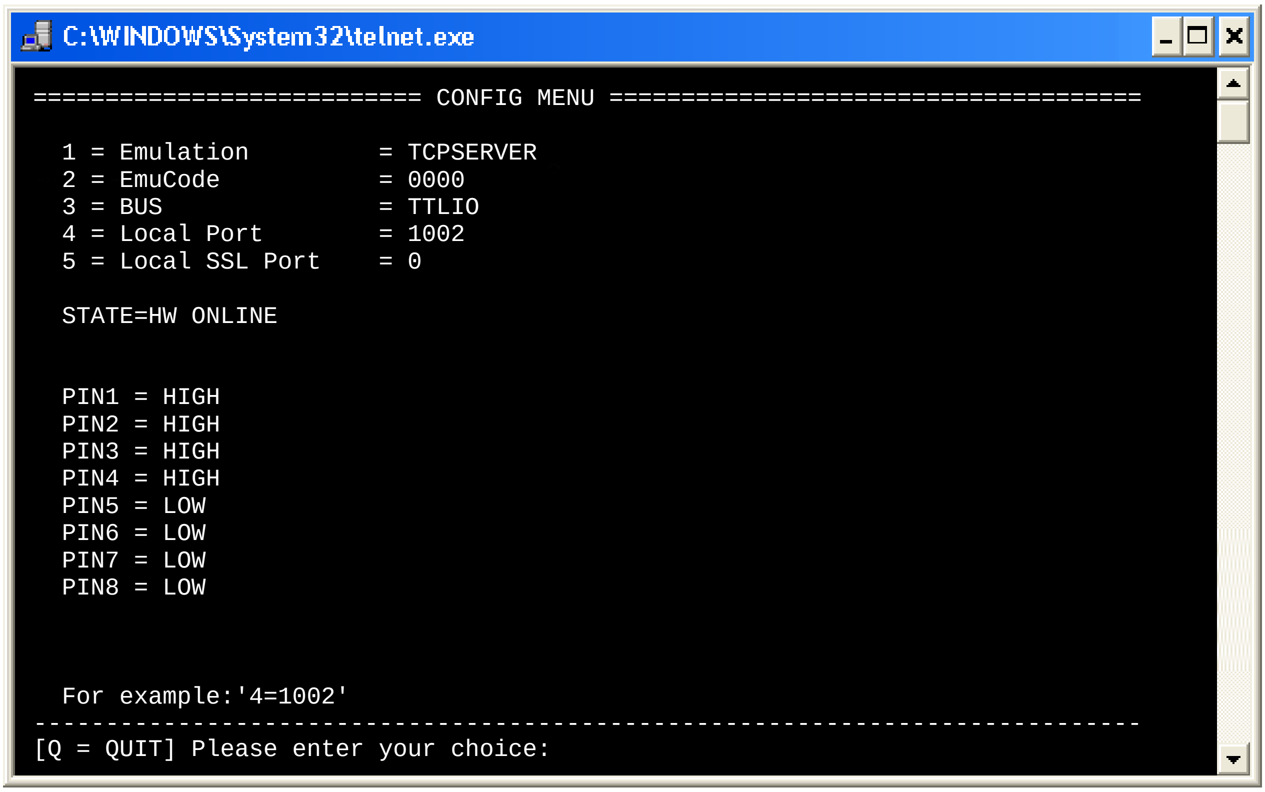
The Data-Poll-Timeout is only considered in the idle running mode. In the sending or receiving mode it tries as soon as possible to send or to receive data. Please keep in mind that the transmittance speed needs to be selected in a way that the Slave will also have the possibility to make the corresponding data available.

10 BUS = TTL IO

The TTL IO (input/output) mode makes available digital signal lines on the BUS1 7 and on the BUS2 8. It is possible to configure them either as output or as input. It is possible to switch them via a protocol which is described in detail in the annex.

Features:
15 i/o switchable digital signal lines
Input/output adjustable for each PIN.
Standard +3.3V TTL digital signals

SETUP TTL IO



```
=========================== CONFIG MENU =====================================

  1 = Emulation          = TCPSERVER
  2 = EmuCode            = 0000
  3 = BUS                = TTLIO
  4 = Local Port         = 1002
  5 = Local SSL Port     = 0

  STATE=HW ONLINE


  PIN1 = HIGH
  PIN2 = HIGH
  PIN3 = HIGH
  PIN4 = HIGH
  PIN5 = LOW
  PIN6 = LOW
  PIN7 = LOW
  PIN8 = LOW



  For example:'4=1002'
-----------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

`1=Emulation:` = The following emulations can be used: MODEM,EMAIL,TCPCLIENT,UDPCLIENT,UDPSERVER,TCPSERVER

Please find examples of applications and detailed explanations regarding emulations in the main manual of the AK XXL products.

`2=Emucode:` = for customer-specific functions

`3=BUS` = TTLIO

Each bus can be reconfigured by entering an admissible value as for example “RS232, RS485, I2C, SPI, LCD, SDCARD or DFCARD“ in the menu item “BUS”. However, it is only possible to activate and configure the selected BUS after resetting.

`4 = Local Port:` = Here you can define the TCP/IP or UDP port, which is approved for the corresponding serial interface.

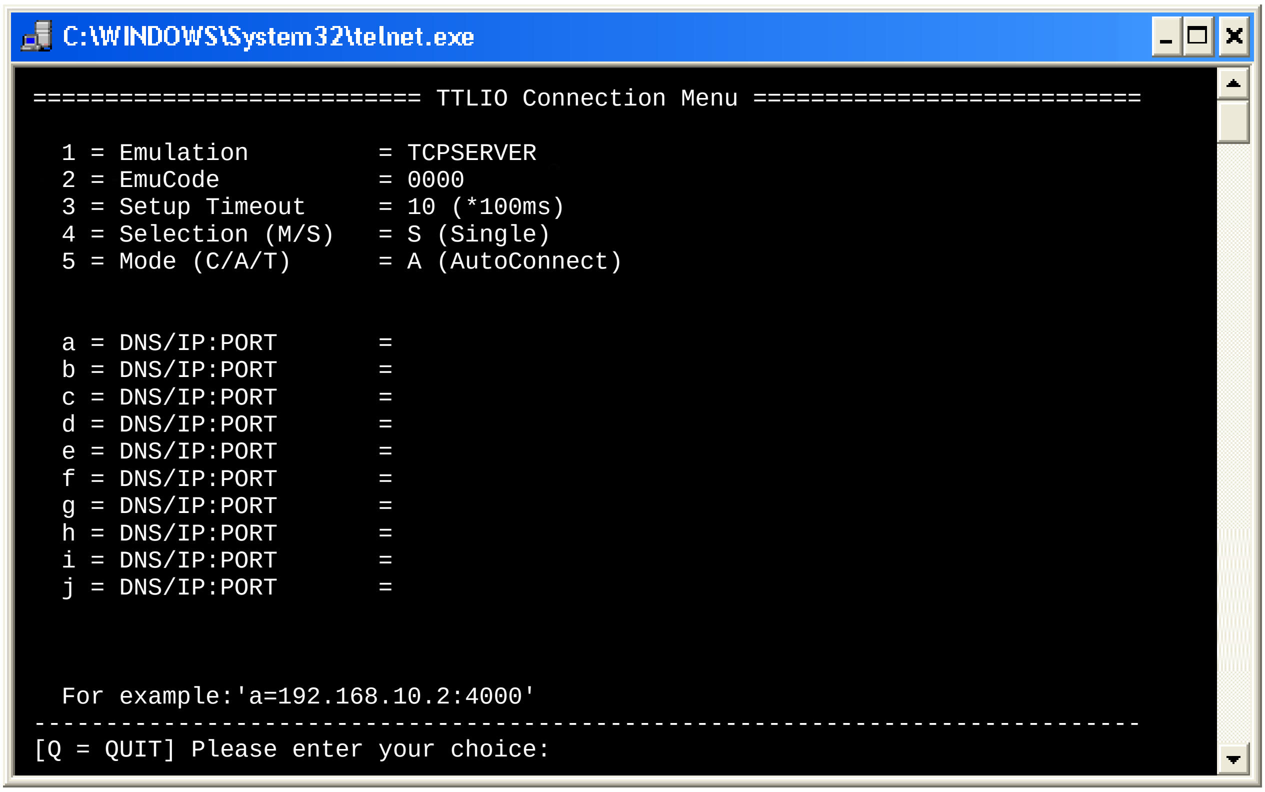
If you enter here 23 (Telnet) or 80 (Browser), the device can no longer be configured via this service. But the output of the data (requests) of browser or Telnet will be performed directly via the serial interface.

`5 = Local SSL Port:=` Has got the same function as the Local Port, but it is used for encoded SSL communication

All other menus are described in the main manual of the AK XXL products.

```
PIN1 = HIGH
PIN2 = HIGH
PIN3 = HIGH
PIN4 = HIGH
PIN5 = LOW
PIN6 = LOW
PIN7 = LOW
PIN8 = LOW
```

Here the current status of the TTL signal line is shown



```
========================== TTLIO Connection Menu ==========================

  1 = Emulation          = TCPSERVER
  2 = EmuCode            = 0000
  3 = Setup Timeout      = 10 (*100ms)
  4 = Selection (M/S)    = S (Single)
  5 = Mode (C/A/T)       = A (AutoConnect)


  a = DNS/IP:PORT        =
  b = DNS/IP:PORT        =
  c = DNS/IP:PORT        =
  d = DNS/IP:PORT        =
  e = DNS/IP:PORT        =
  f = DNS/IP:PORT        =
  g = DNS/IP:PORT        =
  h = DNS/IP:PORT        =
  i = DNS/IP:PORT        =
  j = DNS/IP:PORT        =



  For example:'a=192.168.10.2:4000'
----------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

`1 = Emulation` This is the same option as in the TTLIO Config menu

`2 = EmuCode` This is the same option as in the TTLIO Config menu

`3 = Setup Timeout` Determines how long it will be tried to establish a connection to the target if it is not immediately accessible
Example: 3=10

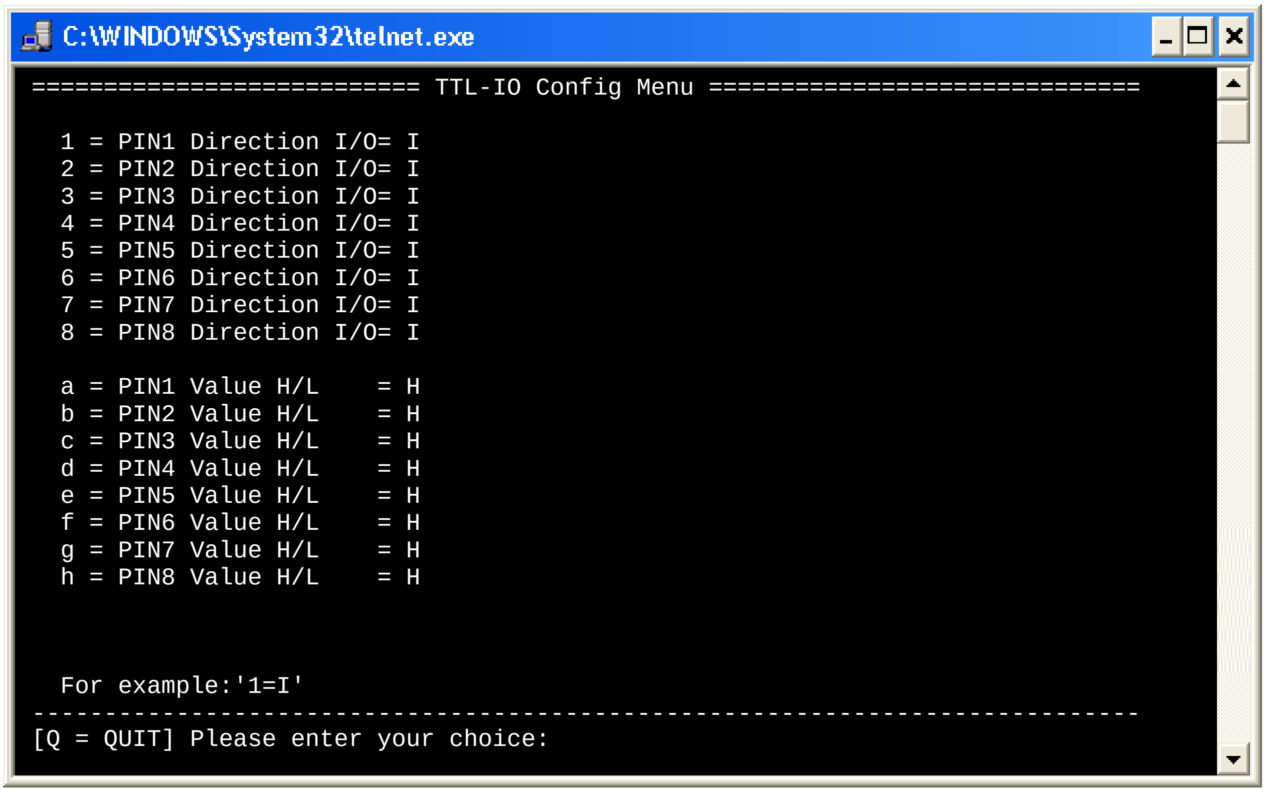
`4 = Selection (M/S)` Using this setting, it is possible to choose between the connection types SINGLE and MULTI. SINGLE means that the data will be transmitted to the first target which is entered in the list, if it is available. If the first target is not available, the device will try to connect itself to the second target and then to the third target, etc. In doing so, the Setup Timeout means how long it is being tried to connect the device to the corresponding target. In the operating mode MULTI, the data will be transmitted to all targets at a time.
Example: 4=M

`5 = Mode (C/A/T)` Here it is possible to select the operating mode. This will influence the mode only for the emulation TCP and UDPCLIENT. The peculiarities of the single options are described in the chapter “Emulations (operating modes)” of the main manual.
Example: 5=C

`a = DNS/IP:PORT` Using this setting, you can determine the 1st target address (IP or DNS) and the first TARGET PORT if you use the TCP or UDPCLIENT emulation. TARGET address and TARGET PORT will be separated by a colon.
All data which are incoming on the serial interface will be automatically sent to the entered target.
Example: a=192.168.0.2:6000

```
b = DNS/IP:PORT
c = DNS/IP:PORT
d = DNS/IP:PORT
e = DNS/IP:PORT
f = DNS/IP:PORT
g = DNS/IP:PORT
h = DNS/IP:PORT
i = DNS/IP:PORT
j = DNS/IP:PORT
```

With the options “b“ to “j“ it is possible to determine other TARGET addresses. In the MULTI mode all addresses will be opened at a time and the data will be sent. It is possible to process 10 targets at a time. In the SINGLE mode the other TARGETS will be addressed one after another if the first one is not available.



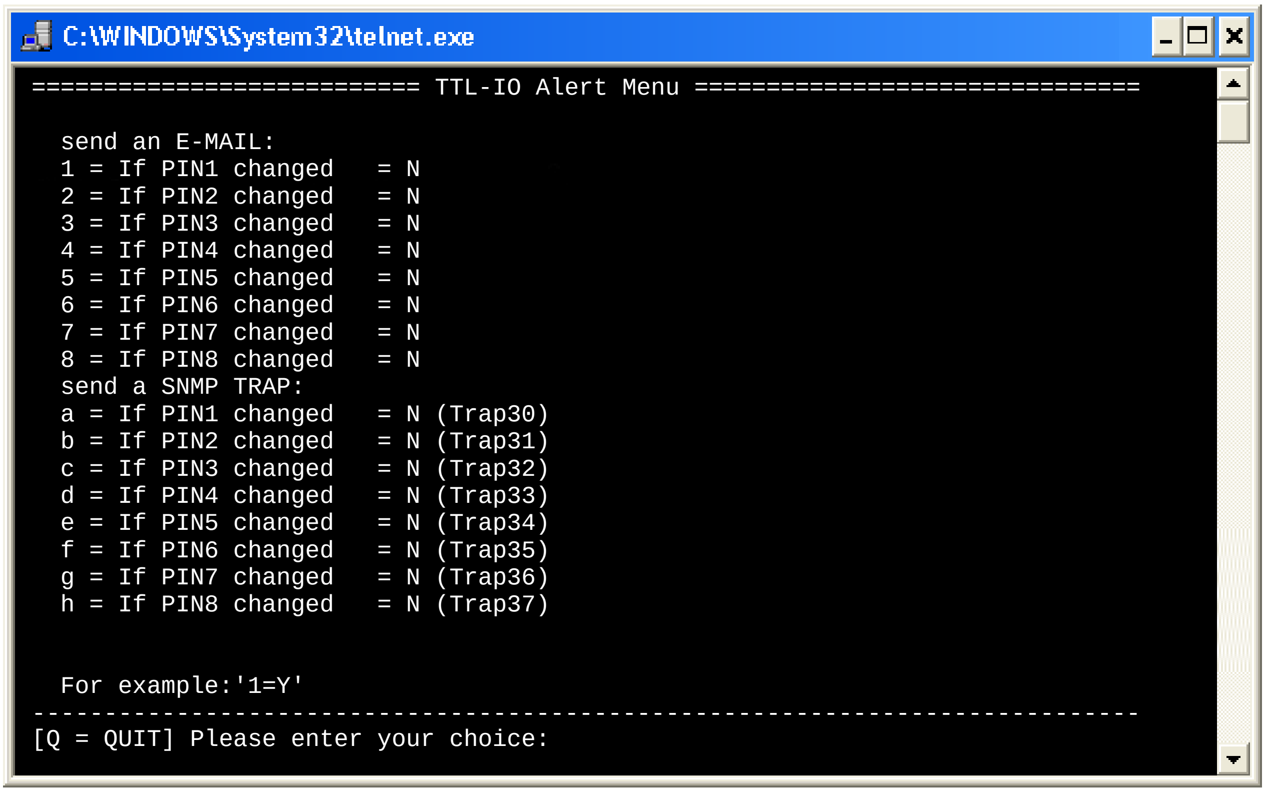
```
=========================== TTL-IO Config Menu ==============================

  1 = PIN1 Direction I/O= I
  2 = PIN2 Direction I/O= I
  3 = PIN3 Direction I/O= I
  4 = PIN4 Direction I/O= I
  5 = PIN5 Direction I/O= I
  6 = PIN6 Direction I/O= I
  7 = PIN7 Direction I/O= I
  8 = PIN8 Direction I/O= I

  a = PIN1 Value H/L    = H
  b = PIN2 Value H/L    = H
  c = PIN3 Value H/L    = H
  d = PIN4 Value H/L    = H
  e = PIN5 Value H/L    = H
  f = PIN6 Value H/L    = H
  g = PIN7 Value H/L    = H
  h = PIN8 Value H/L    = H



  For example:'1=I'
-----------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

```
1 = PIN1 Direction I/O= I
2 = PIN2 Direction I/O= I
3 = PIN3 Direction I/O= I
4 = PIN4 Direction I/O= I
5 = PIN5 Direction I/O= I
6 = PIN6 Direction I/O= I
7 = PIN7 Direction I/O= I
8 = PIN8 Direction I/O= I
```

Using these settings, it is possible to determine if the single PINs shall act as input or output when starting the interface. It is also possible to set this function via the protocol during operation. Please find an explanation on the following pages.

```
a = PIN1 Value H/L    = H
b = PIN2 Value H/L    = H
c = PIN3 Value H/L    = H
d = PIN4 Value H/L    = H
e = PIN5 Value H/L    = H
f = PIN6 Value H/L    = H
g = PIN7 Value H/L    = H
h = PIN8 Value H/L    = H
```

Using these settings, it is possible to determine if the single PINs are HIGH or LOW when starting the interface. This function can also be changed via the protocols during operation. Please find an explanation on the following pages.



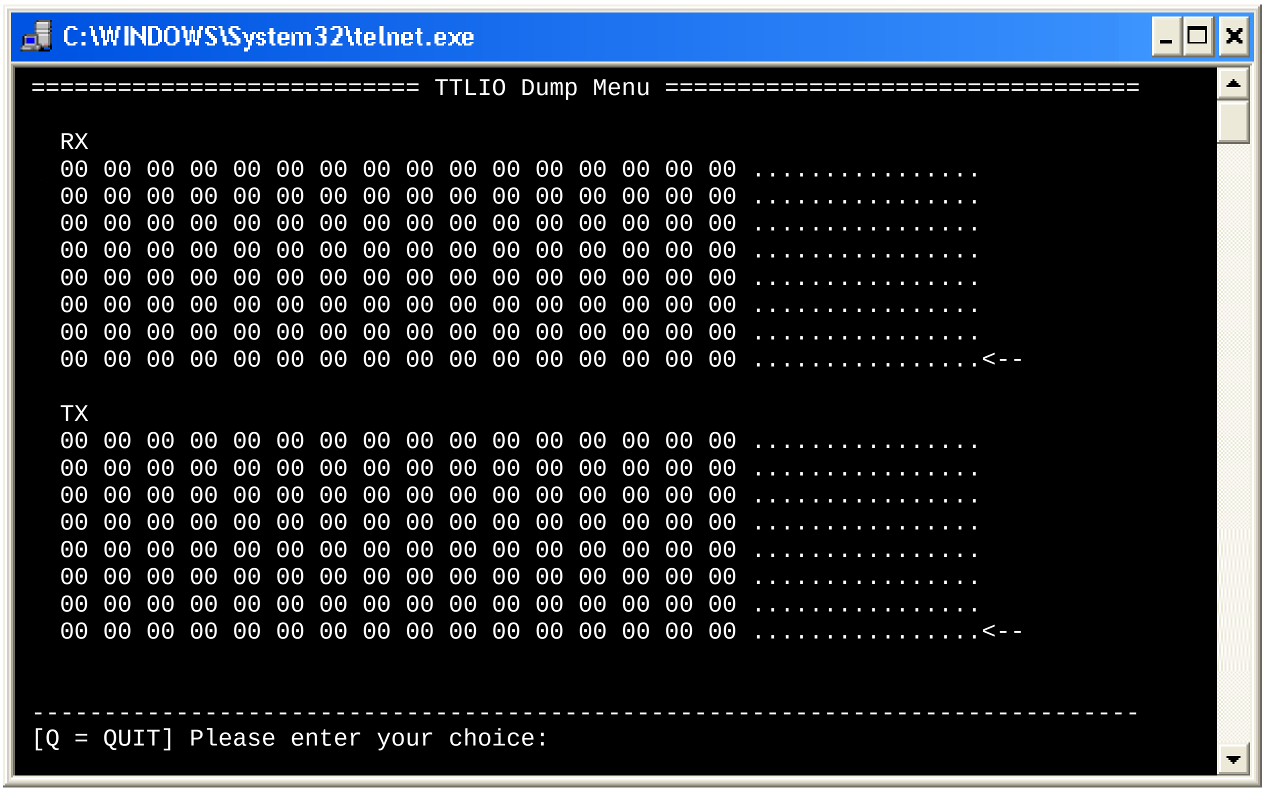
```
=========================== TTL-IO Alert Menu ================================

  send an E-MAIL:
  1 = If PIN1 changed   = N
  2 = If PIN2 changed   = N
  3 = If PIN3 changed   = N
  4 = If PIN4 changed   = N
  5 = If PIN5 changed   = N
  6 = If PIN6 changed   = N
  7 = If PIN7 changed   = N
  8 = If PIN8 changed   = N
  send a SNMP TRAP:
  a = If PIN1 changed   = N (Trap30)
  b = If PIN2 changed   = N (Trap31)
  c = If PIN3 changed   = N (Trap32)
  d = If PIN4 changed   = N (Trap33)
  e = If PIN5 changed   = N (Trap34)
  f = If PIN6 changed   = N (Trap35)
  g = If PIN7 changed   = N (Trap36)
  h = If PIN8 changed   = N (Trap37)


  For example:'1=Y'
------------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

```
send an E-MAIL:
  1 = If PIN1 changed   = N
  2 = If PIN2 changed   = N
  3 = If PIN3 changed   = N
  4 = If PIN4 changed   = N
  5 = If PIN5 changed   = N
  6 = If PIN6 changed   = N
  7 = If PIN7 changed   = N
  8 = If PIN8 changed   = N
```

If the sending data are correctly entered in the SMTP menu and if the SMTP function had been activated, an e-mail will be sent to the entered recipient, if different events are occurring.

```
send a SNMP TRAP:
  a = If PIN1 changed   = N (Trap30)
  b = If PIN2 changed   = N (Trap31)
  c = If PIN3 changed   = N (Trap32)
  d = If PIN4 changed   = N (Trap33)
  e = If PIN5 changed   = N (Trap34)
  f = If PIN6 changed   = N (Trap35)
  g = If PIN7 changed   = N (Trap36)
  h = If PIN8 changed   = N (Trap37)
```

If the SNMP-Traps are activated in the SNMP menu, SNMP-Traps will be sent to the set server when the described events are occurring.



```
=========================== TTLIO Dump Menu =================================

  RX
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................<--

  TX
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................<--


-----------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

In the Dump menu of the TTLIO-BUS only the item TX is relevant, since no data will be transferred using this BUS variant but only lines are switched. Under TX it is possible to check the protocol commands which are incoming via the network.

Control the IO – Pins of the XT-PICO-XXL:
You always have to send two bytes. The first is the command byte and the second byte is the pin value

For example:
In order to configure PIN9 from the XT-PICO to an output, you have to send following control characters:

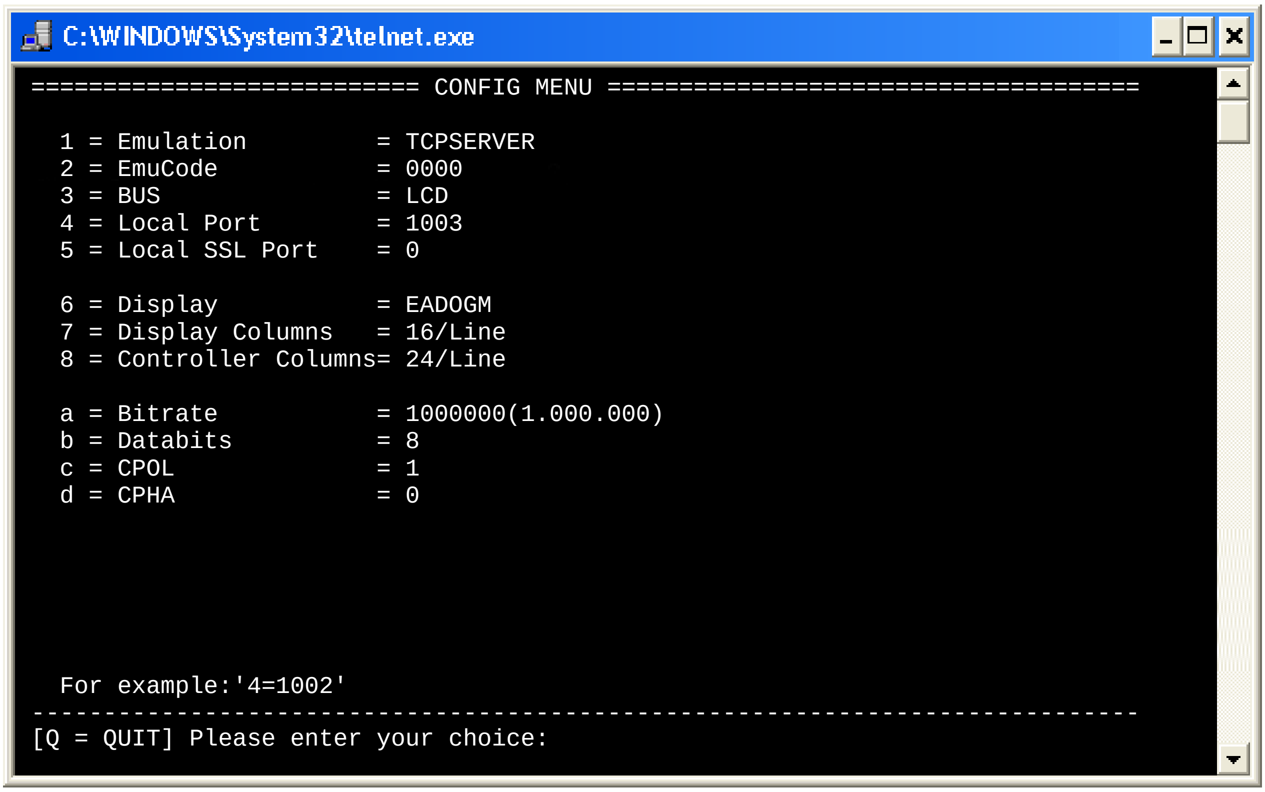
0x04 0x01	= Configure pin to Output
0x02 0x01	= set pin
0x01 0x01	= clear pin

Commands:

0x00	= Read Pin
0x01	= Clear Pin
0x02	= Set Pin
0x03	= Configure Pin to Input
0x04	= Configure pin to Output
0x05	= Clear Pullup
0x06	= Set Pullup

11 BUS = LCD

Using this special mode, it is possible to control LCD displays. Currently entered displays are the EADOGM and the HD44780.



```
=========================== CONFIG MENU ======================================

  1 = Emulation          = TCPSERVER
  2 = EmuCode            = 0000
  3 = BUS                = LCD
  4 = Local Port         = 1003
  5 = Local SSL Port     = 0

  6 = Display            = EADOGM
  7 = Display Columns    = 16/Line
  8 = Controller Columns= 24/Line

  a = Bitrate            = 1000000(1.000.000)
  b = Databits           = 8
  c = CPOL               = 1
  d = CPHA               = 0





  For example:'4=1002'
------------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

`1=Emulation:` = The following emulations can be used:
MODEM,EMAIL,TCPCLIENT,UDPCLIENT,UDPSERVER,TCPSERVER
Please find examples of applications and detailed explanations regarding emulations in the main manual of the AK XXL products.

`2=Emucode:` = for customer-specific functions

`3=BUS` = **LCD**
Each bus can be reconfigured by entering an admissible value as for example "RS232, RS485, I2C, SPI, LCD, SDCARD or DFCARD" in the menu item "BUS". However, it is only possible to activate and configure the selected BUS after resetting.

`4 = Local Port:` = Here you can define the TCP/IP or UDP port, which is approved for the corresponding serial interface.
If you enter here 23 (Telnet) or 80 (Browser), the device can no longer be configured via this service. But the output of the data (requests) of browser or Telnet will be performed directly via the serial interface.

`5 = Local SSL Port:=` Has got the same function as the Local Port, but it is used for encoded SSL communication

All other menus are described in the main manual of the AK XXL products.

6 = Display Columns = EADOGM or HD44780

7 = Display Columns = Using this setting, it is possible to indicate the display fields existing on the display per row

8 = Controller Columns= Using this setting, it is possible to enter the display elements which are made available by the display controller

The values are all preset and compliant to the display. However, these values are only relevant for the SPI display EADOGM.

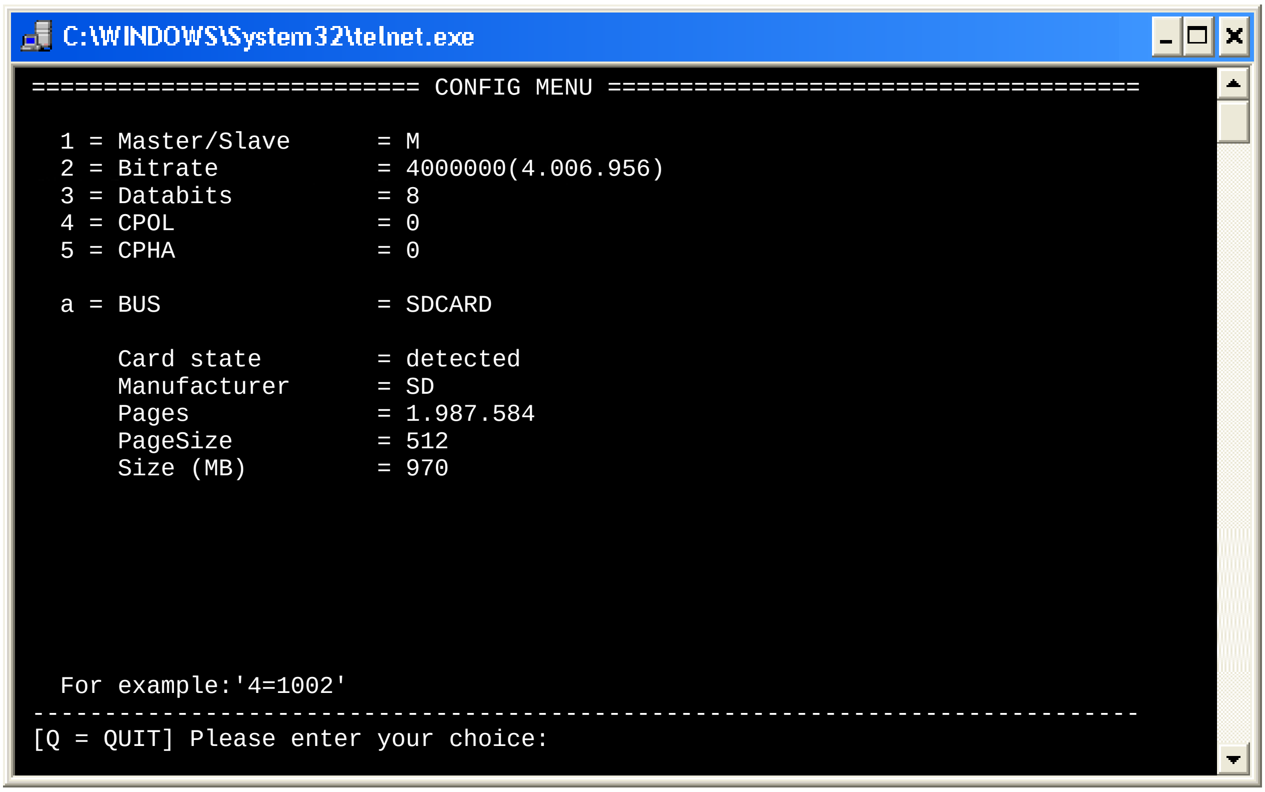
a = Bitrate = Bit rate on the SPI bus. Possible values 1-25.000.000 bit/sec.

b = Databits = Possible values 8-16

c = CPOL = Clock Polarity possible values 0,1

5 = CPHA = Clock Phase possible values 0,1

12 BUS = SDCARD / DFCARD



C:\WINDOWS\System32\telnet.exe

```
=========================== CONFIG MENU =====================================

  1 = Master/Slave      = M
  2 = Bitrate           = 4000000(4.006.956)
  3 = Databits          = 8
  4 = CPOL              = 0
  5 = CPHA              = 0

  a = BUS               = SDCARD

      Card state        = detected
      Manufacturer      = SD
      Pages             = 1.987.584
      PageSize          = 512
      Size (MB)         = 970







  For example:'4=1002'
-----------------------------------------------------------------------------
[Q = QUIT] Please enter your choice:
```

If no card is inserted into the slot:

```
      Card state         = not detected
      Card info          = not identified
```

If a card is inserted into the slot:

```
      Card state         = detected
      Manufacturer       = SD
      Pages              = 1.987.584
      PageSize           = 512
      Size (MB)          = 970
```

The values are all preset and compliant to most cards.

`1 = Master/Slave` = An SD card / DF card can only be operated in the Master mode

`2 = Bitrate` = Bit rate on the SPI bus. Possible values 1-25.000.000 bit/sec.
Most cards can be operated with a speed of 4 M bit.

`3 = Databits` = Possible values 8

`4 = CPOL` = Clock Polarity possible values 0,1

`5 = CPHA` = Clock Phase possible values 0,1

`a = BUS` = **SDCARD**
Each bus can be reconfigured by entering an admissible value as for example “RS232, RS485, I2C, SPI, LCD, SDCARD or DFCARD“ in the menu item “BUS”. However, it is only possible to activate and configure the selected BUS after resetting.