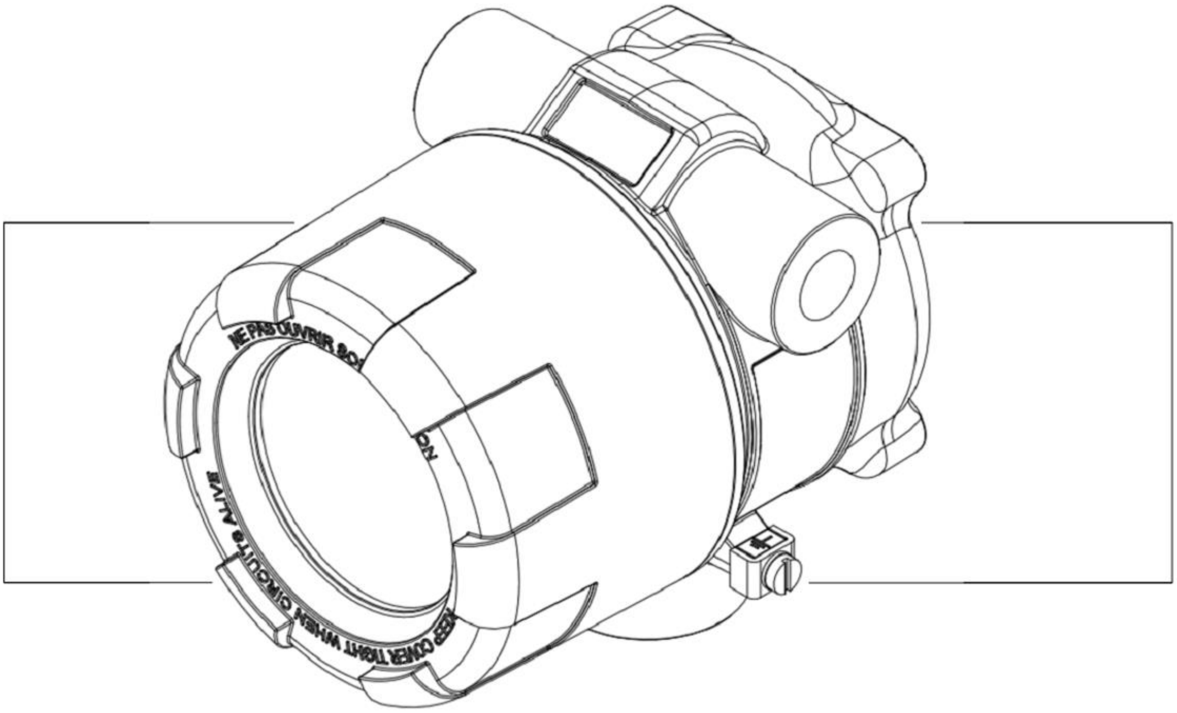
MODBUS Communication

with Fluidwell E-Series



Troubleshooting Guide

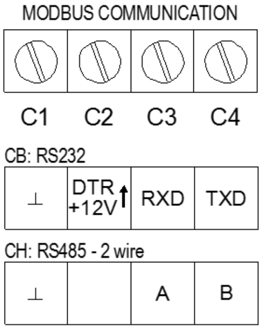
Table of content

Communication settings

For all E-Series the *communication* settings are: **8 databits, no parity, 1 stopbit.**
The factory default settings are: **9600 BAUD, Modbus address 1**
Communication mode can be either **ASCII** or **RTU**, depending on the complexity of the unit. Please verify the various settings menu.

Physical connection

For the connection lay-out and function, see the instruction manual of the related model.
As a reference, a generic connector lay-out is given



CB: RS232

⊥	DTR +12V ↑	RXD	TXD

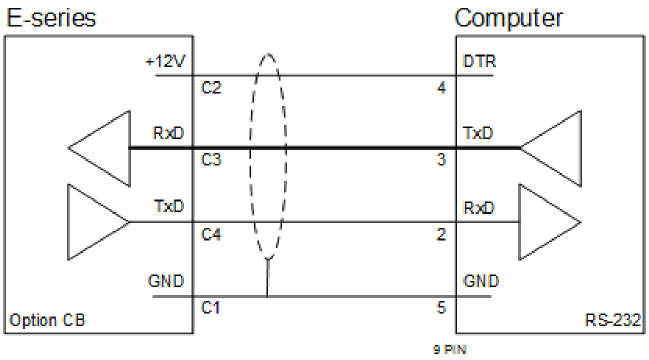
CH: RS485 - 2 wire

⊥		A	B

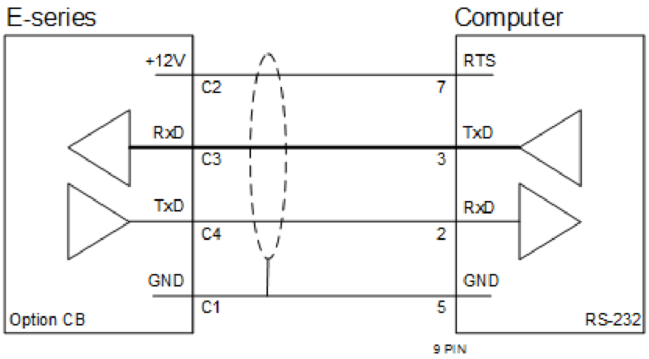
Fig. 1 Communication connections - typical

RS-232 communication

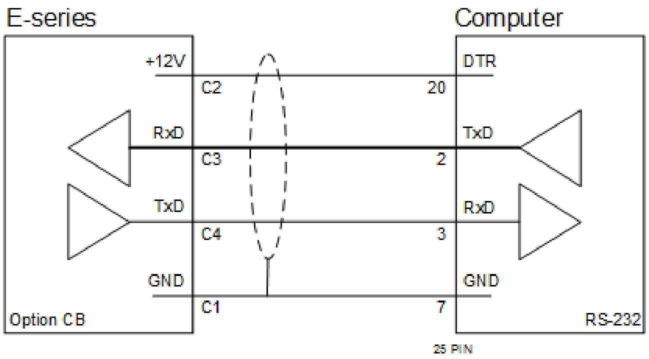
When using the RS-232 communication (option CB), terminal C2 is used for supplying the E-Series interface. Please connect the DTR (or the RTS) signal of the interface to this terminal and set it active (+12V). If no active signal is available it is possible to connect a separate supply between terminals C1 and C2 with a voltage of 8V to 12V DC.



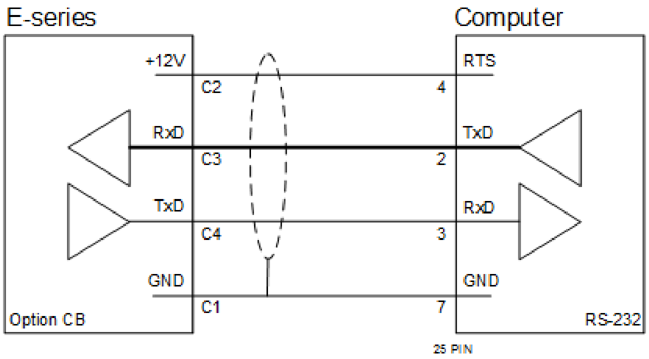
RS-232 – DTR – 9 pin



RS-232 – RTS – 9 pin



RS-232 – DTR – 25 pin



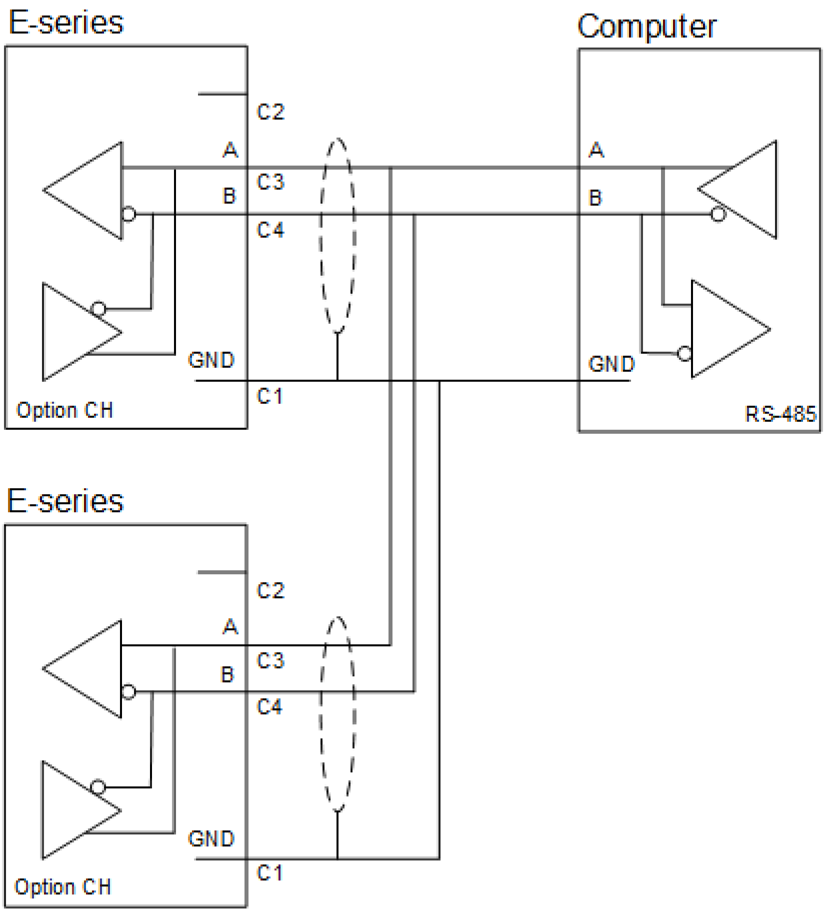
RS-232 – RTS – 25 pin

RS-485 communication

For battery and loop-powered applications communicating through RS-485, it is important to disconnect all termination resistors.
The E-Series are designed to work without them for power saving reasons.

RS-485 - 2 wire communication

With **2-wire** communication (option CH) you need to use RS-485 drivers. It is possible to connect up to 32 slaves in a multidrop configuration. There is no need to supply power to the interface of the E-Series. The grounds must be connected to all drivers. Optional shielding should be connected on one side only.



Register addressing

In MODBUS, every register is categorised in a group.
In the E-series **all registers** are located in the **‘Holding registers’** group.
By doing this, all registers can be accessed by just two Modbus function codes.
Currently, the function codes supported are:

- function code 3 “Read Holding Registers” (4x references);
- function code 16 “Preset Multiple Registers” (4x references).

Within a PLC or application with MODBUS interface every Holding registers is assigned an address somewhere between 40001 and 50000.
In the actual physical Modbus data bytes these addresses are converted to a value between **0** and **9999** plus an indication of the group.

The registers addresses are available from the appendix of the related instruction manual which came with the E-series.

When the PLC address range is required (4x references typically used by PLCs), please add a value of 40001 to the Modbus PDU address. E.g. reading the serial number with PLC-based addressing means reading 165 + 40001 = register 40166.
Variables spanning multiple registers use ‘big-endian’ data representation. This means that the lowest register holds the least significant word of the variable.
Although most Modbus Masters will support variables that span 2 registers, variables spanning more registers sometimes require you to manually calculate the resulting value.

WARNING **Variables spanning multiple registers have always to be read / written in 1 single action!**

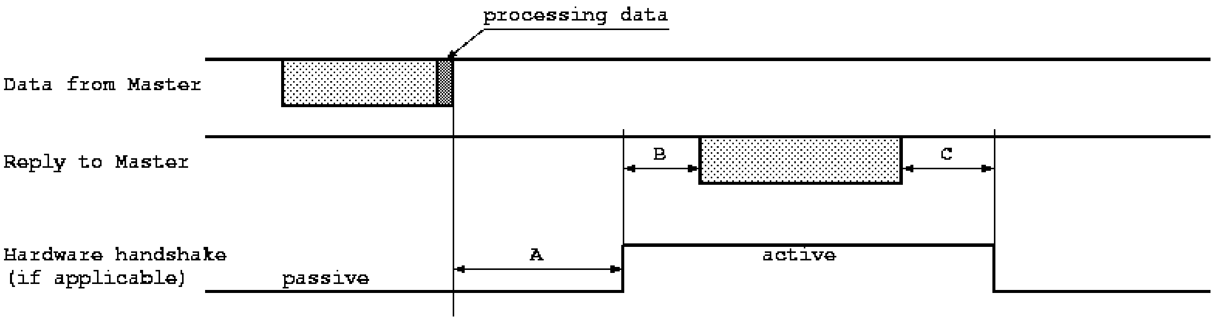
Following example shows how data is represented and how this calculation can be accomplished:
For a total-value of 158928, the following register data has been received by the Modbus master:

PDU Address 566	= 0x0000	= 0
PDU Address 567	= 0x0002	= 2
PDU Address 568	= 0x6CD0	= 27856

If we interpret this as a 48 bit unsigned integer value, it's value would be:
0x0000.0002.6CD0 = 158928.
If this value needs to be calculated: 0*65536*65536 + 2*65536 + 27856 = 158928.

Communication timing

The current Modbus implementation provides a way to optimize communication timing. This is especially important when hardware handshake is applicable, i.e. RS485.



A = comdelay (register 14, 147)
B = pre send delay (register 1273)
C = post send delay (register 1274)

The processing and validation of the request frame from the master takes place almost immediate after the message is completed. It takes ample time to process the data and prepare the response frame back to the master. The device could answer very fast which may give problems at the master communication channel.

The parameter comdelay (register 14, 147 in ms) provides a response delay between the end of the request frame and the start of the response. The comdelay settings don't have a menu entry but can be configured over Modbus.

For RS485 the response action would first put the RS485 transceiver in transmit mode. The device now could send the response frame immediately and after completion of the frame the RS485 transceiver would switch back to receive mode.

This may cause problems in some environments. Therefore we added two extra delay settings. The pre-send delay setting (in ms) adds an extra delay after the transceiver is switched to transmit mode and before the first data character is output. The post-send delay setting (in ms) adds an extra delay after the last data character is output and before the transceiver is switched back to receive mode. This may overcome glitches on the data line due to switching of the transceiver.

The pre-send and post-send delay settings don't have a menu entry but can be configured over Modbus.

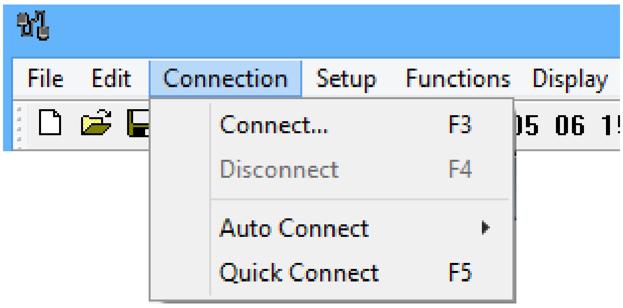
Example using MODBUS Poll

Below are a few screen shots of configuring with MODBUS Poll, a computer-utility that can easily be obtained from the internet.

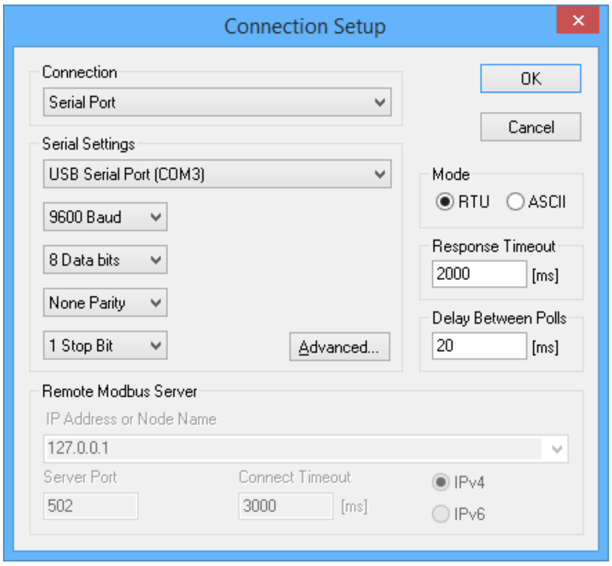
The connection settings

These examples assume that the E-series is connected to the computer and the comport number in use is known.
When using an RS232 to RS485 converter, as in the example, switch on the Flow control, RTS Toggle box.

1. Double click the Modbus poll icon to open the application.
2. From the menu-bar, select the Connection tab.
3. In the Connection pull down menu, select Connect…
4. Wait for the Dialog box to open.



5. Complete the settings in the Connection Setup dialog box as shown.
   The Serial Settings depend on the comport in use an might show another comport (COMx).
6. Click the OK button to confirm the selections.



Connection: Serial Port

Serial Settings: USB Serial Port (COM3)
9600 Baud
8 Data bits
None Parity
1 Stop Bit

Mode: ● RTU ○ ASCII

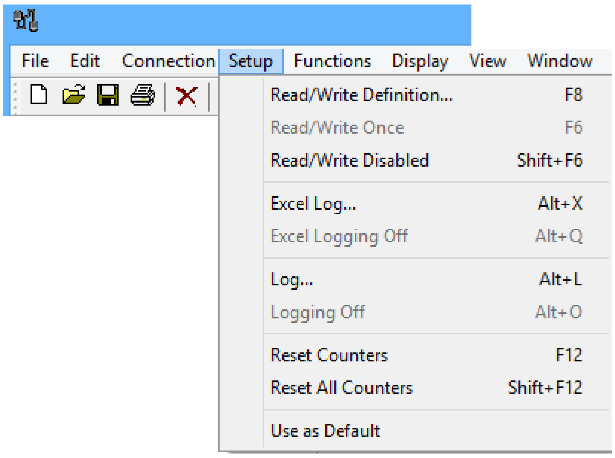
Response Timeout: 2000 [ms]

Delay Between Polls: 20 [ms]

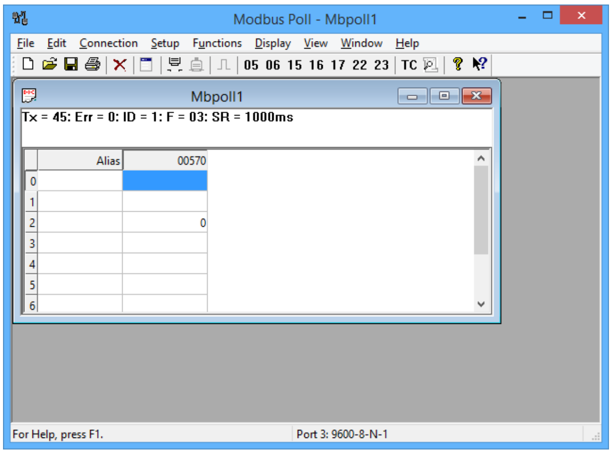
Remote Modbus Server
IP Address or Node Name: 127.0.0.1
Server Port: 502
Connect Timeout: 3000 [ms]
● IPv4
○ IPv6

The poll-definition for reading the Serial Number, which is 2 registers long corresponds with the PLC address 40003 and 40004.

7. From the menu-bar, select the Setup tab.
8. In the Setup pull down menu, select Read/Write Definition…
9. Wait for the Dialog box to open.



10. Complete the settings in the Read/Write Definition dialog box as shown.

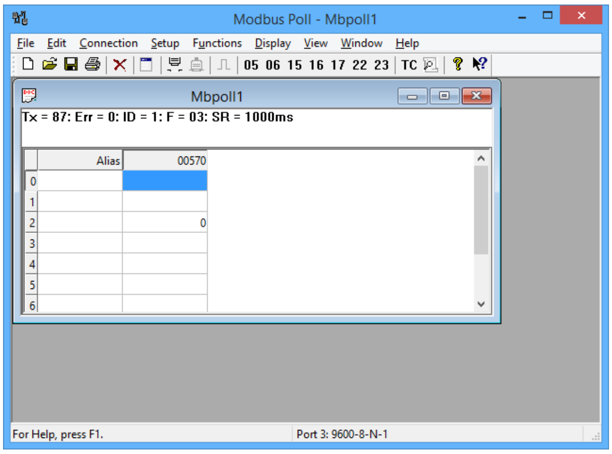

Modbus Poll - Mbpoll1
File Edit Connection Setup Functions Display View Window Help
05 06 15 16 17 22 23 TC
Mbpoll1

Tx = 45: Err = 0: ID = 1: F = 03: SR = 1000ms

	Alias	00570
0		
1		
2		0
3		
4		
5		
6		

For Help, press F1.
Port 3: 9600-8-N-1

11. Wait for the Modbus Poll main screen to show.
12. Note that there is communication traffic which is indicated by the running numbers.


Modbus Poll - Mbpoll1
File Edit Connection Setup Functions Display View Window Help
05 06 15 16 17 22 23 TC
Mbpoll1

Tx = 87: Err = 0: ID = 1: F = 03: SR = 1000ms

	Alias	00570
0		
1		
2		0
3		
4		
5		
6		

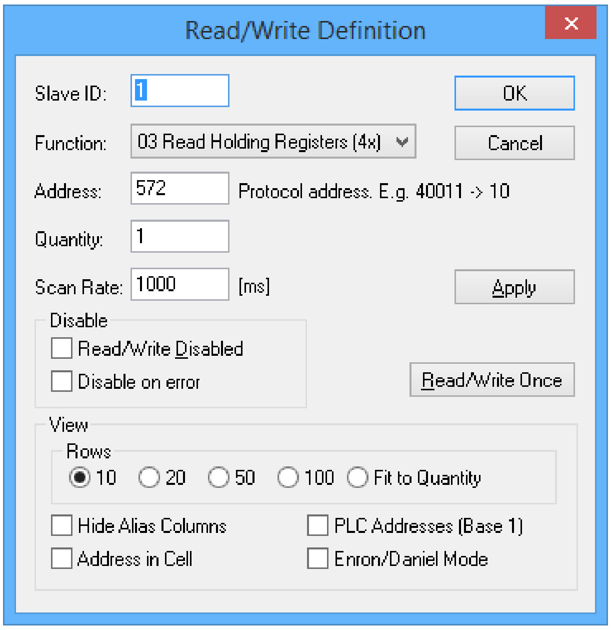
For Help, press F1.
Port 3: 9600-8-N-1

Troubleshooting with MODBUS Poll

Using the communication traffic window

Anytime you cannot get the communication to work after checking all settings, it is advised to examine the Communication Traffic window. This window displays all data that is sent and received across the communication line.
For example:
The Read/Write Definition for reading the Serial number, which is 1 register long corresponding to PLC address 40573.

1. From the menu-bar, select the Setup tab.
2. In the Setup pull down menu, select Read/Write Definition…
3. Wait for the Dialog box to open.
4. Complete the settings in the Read/Write Definition dialog box as shown.
5. Click the OK button, to execute the command.



Read/Write Definition

Slave ID: 1

Function: 03 Read Holding Registers (4x)

Address: 572 Protocol address. E.g. 40011 -> 10

Quantity: 1

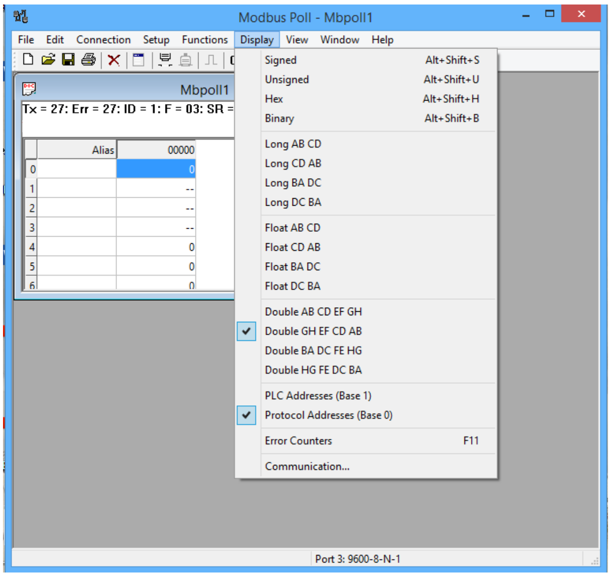
Scan Rate: 1000 [ms]

Disable
□ Read/Write Disabled
□ Disable on error

View
Rows
● 10 ○ 20 ○ 50 ○ 100 ○ Fit to Quantity

□ Hide Alias Columns
□ PLC Addresses (Base 1)
□ Address in Cell
□ Enron/Daniel Mode

6. From the menu-bar, select the Display tab.
7. In the Display pull down menu, select Communication...
8. Wait for the Dialog box to open.



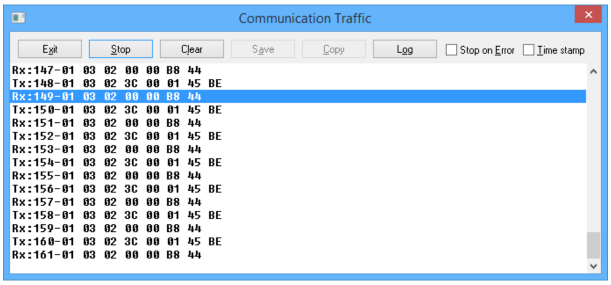
Modbus Poll - Mbpoll1

Tx = 27: Err = 27: ID = 1: F = 03: SR =

Port 3: 9600-8-N-1

9. The Communication Traffic window opens.

   Note, the screen shows a valid communication.



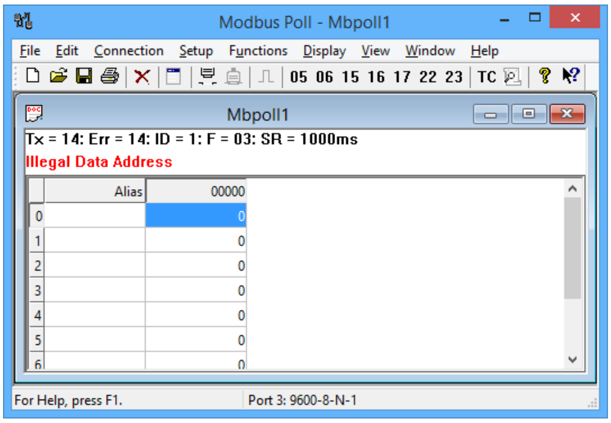
Communication Traffic

□ Stop on Error □ Time stamp

```
Rx:147-01 03 02 00 00 B8 44
Tx:148-01 03 02 3C 00 01 45 BE
Rx:149-01 03 02 00 00 B8 44
Tx:150-01 03 02 3C 00 01 45 BE
Rx:151-01 03 02 00 00 B8 44
Tx:152-01 03 02 3C 00 01 45 BE
Rx:153-01 03 02 00 00 B8 44
Tx:154-01 03 02 3C 00 01 45 BE
Rx:155-01 03 02 00 00 B8 44
Tx:156-01 03 02 3C 00 01 45 BE
Rx:157-01 03 02 00 00 B8 44
Tx:158-01 03 02 3C 00 01 45 BE
Rx:159-01 03 02 00 00 B8 44
Tx:160-01 03 02 3C 00 01 45 BE
Rx:161-01 03 02 00 00 B8 44
```

Error messages

Whenever something is wrong with the incoming message, the receiver answers with an error-message.
Note that there is no communication.



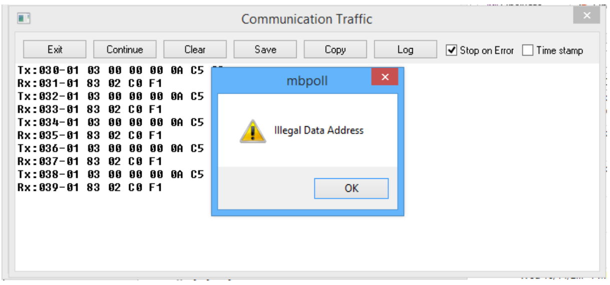
Modbus Poll - Mbpoll1
File Edit Connection Setup Functions Display View Window Help
05 06 15 16 17 22 23 TC
Mbpoll1

```
Tx = 14: Err = 14: ID = 1: F = 03: SR = 1000ms
Illegal Data Address
```

	Alias	00000
0		0
1		0
2		0
3		0
4		0
5		0
6		0

For Help, press F1. Port 3: 9600-8-N-1

Error messages are also visible in the communication traffic window. For error messages, select the Stop on Error box. When and error occurs, a popup shows with the error message.



☑ Stop on Error ☐ Time stamp

```
Tx:030-01 03 00 00 00 0A C5
Rx:031-01 83 02 C0 F1
Tx:032-01 03 00 00 00 0A C5
Rx:033-01 83 02 C0 F1
Tx:034-01 03 00 00 00 0A C5
Rx:035-01 83 02 C0 F1
Tx:036-01 03 00 00 00 0A C5
Rx:037-01 83 02 C0 F1
Tx:038-01 03 00 00 00 0A C5
Rx:039-01 83 02 C0 F1
```

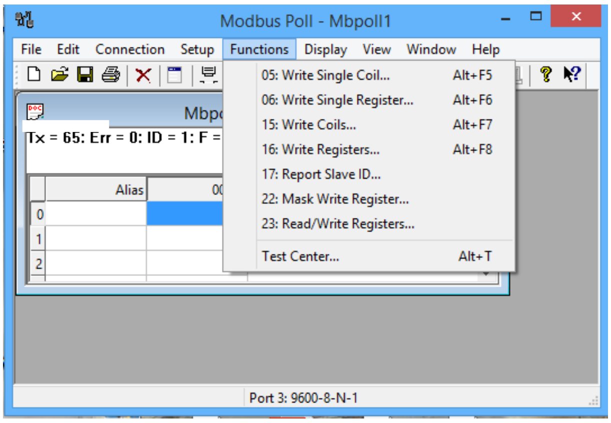
Illegal Data Address

Problems with USB and/or signal-converters

When using signal-converters to interface between e.g. RS-232, RS-485 or RS-422, sometimes problems occur due to delay between switching from send to receive of such converters. Especially the combination of a converter and a USB-com port can give problems because of the total delay that is introduced in the system.

There is one option you could try if you think this could be causing the problems. There is a possibility to set a communication delay in the E-Series This delay is then inserted between receiving a message and responding to it.

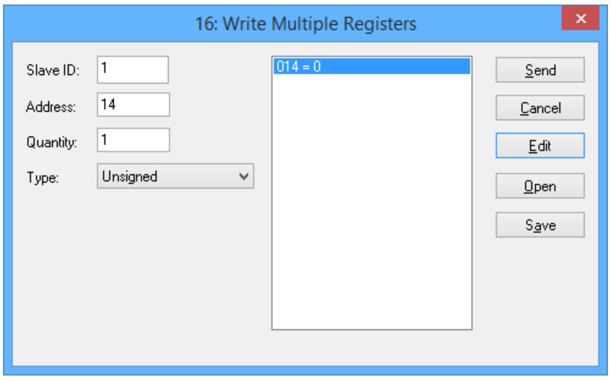
1. From the menu-bar, select the Functions tab.
2. In the Functions menu, select Write Registers…
3. Wait for the Dialog box to open.


Modbus Poll - Mbpoll1
File Edit Connection Setup Functions Display View Window Help
05: Write Single Coil... Alt+F5
06: Write Single Register... Alt+F6
15: Write Coils... Alt+F7
16: Write Registers... Alt+F8
17: Report Slave ID...
22: Mask Write Register...
23: Read/Write Registers...
Test Center... Alt+T
Mbpo
Alias
0
1
2

Tx = 65: Err = 0: ID = 1: F =

Port 3: 9600-8-N-1

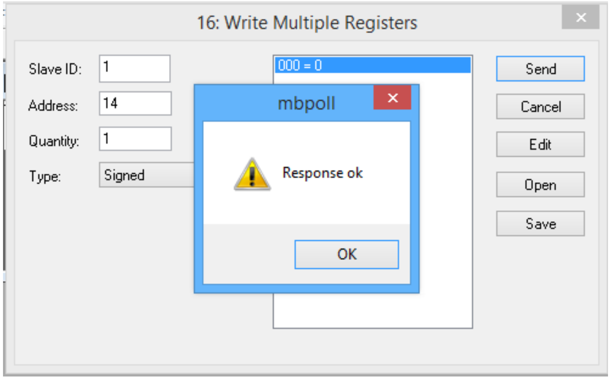
4. The 16: Write Multiple Registers Dialog box opens.
5. Complete the settings in the 16: Write Multiple Registers dialog box as shown



Slave ID: 1
Address: 14
Quantity: 1
Type: Unsigned

014 = 0

6. Click the Send button.
   If the communication is OK, the Response OK message popup shows.
7. Click the OK button to close the message popup.

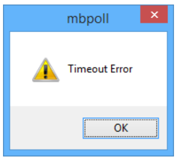


Slave ID: 1
Address: 14
Quantity: 1
Type: Signed

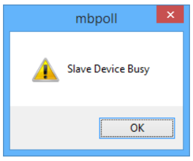
000 = 0

Response ok

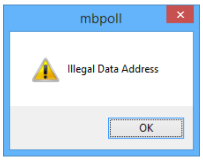
- A wrong slave ID will result in a Timeout Error message.
- A wrong or unknown address can result in a Slave Device Busy message.
- A wrong Quantity can result in an Illegal Data Address message.



Timeout Error



Slave Device Busy



Illegal Data Address

NOTES

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

NOTES

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....