How to configure an Anybus Modbus-TCP slave module with Unity Pro L

Document history

Revision	Date	Description	Author
1.00	2007-06-20	Created	Thorbjörn Palm
1.01	2007-07-18	Minor revision	Thorbjörn Palm
1.02	2007-07-20	Minor correction	Thorbjörn Palm
1.03	2008-03-05	Updated layout	Thorbjörn Palm

More information about the network and products

The latest manuals and EDS-files can be found on the HMS homepage, www.anybus.com

The Modbus user organisation has a homepage on the Internet, http://www.modbus.org/. Several technical guides are available in or via this page.

Contents

1. Applicable Anybus products

Description	Name / Type
Anybus X-gateway	Modbus-TCP
Anybus Communicator	Modbus-TCP
Anybus-Slave	Modbus-TCP
Anybus-PCI	Modbus-TCP
Anybus-CompactCom	Modbus-TCP
Anybus-IC	EtherNet/IP (supporting Modbus-TCP)

Note: The configuration of the X-gateway and the Anybus Communicator is described in this document. In the case with the remaining Anybus products this document is applicable, but the configuration is depending on the type of application.

2. Requirements

Description	Name / Type	Version
Schneider Premium PLC	TSX P571634 with ETY Port master Modbus-TCP card	n.a.
PLC software	Unity Pro L	2.1
Anybus IPconfig tool	Anybus IPconfig	1.3.1.1
X-gateway Network Interface Addendum	Anybus X-gateway Ethernet Slave Interface, Network Interface Addendum	1.02
X-gateway User Manual	X-gateway Generic User Manual	1.02
Communicator User Manual	Anybus Communicator for Ethernet, User Manual	2.01
Slave Field bus Appendix	Anybus-S Ethernet, Field bus Appendix	1.42
IC Field bus Appendix	Anybus-IC Ethernet/IP, Field bus Appendix	1.52
PCI Interface Design Guide	Anybus-S Slave & Master, Parallel Interface Design Guide	2.0
CompactCom Network Interface Appendix	CompactCom Modbus-TCP, Network Interface Appendix	2.02
Power supply 24VDC	n.a.	n.a
Configuration cables	n.a.	n.a.
Null modem cable (Supplied with the Anybus X-gateway)	n.a.	n.a.

3. Solution overview

This application note describes how to configure an Anybus Modbus-TCP Slave product with a Schneider PLC. Below you can find an overview of the system described in this document. Other nodes may be attached to the network, but are not necessary

Note: This document is valid for all Anybus Modbus TCP products, however sections written in *italics* describe the configuration of a specific product.

The contents describe step by step how a configuration is done. This document assumes the reader is familiar with industrial communication, Modbus-TCP networks and HMS Communicator and X-gateway.

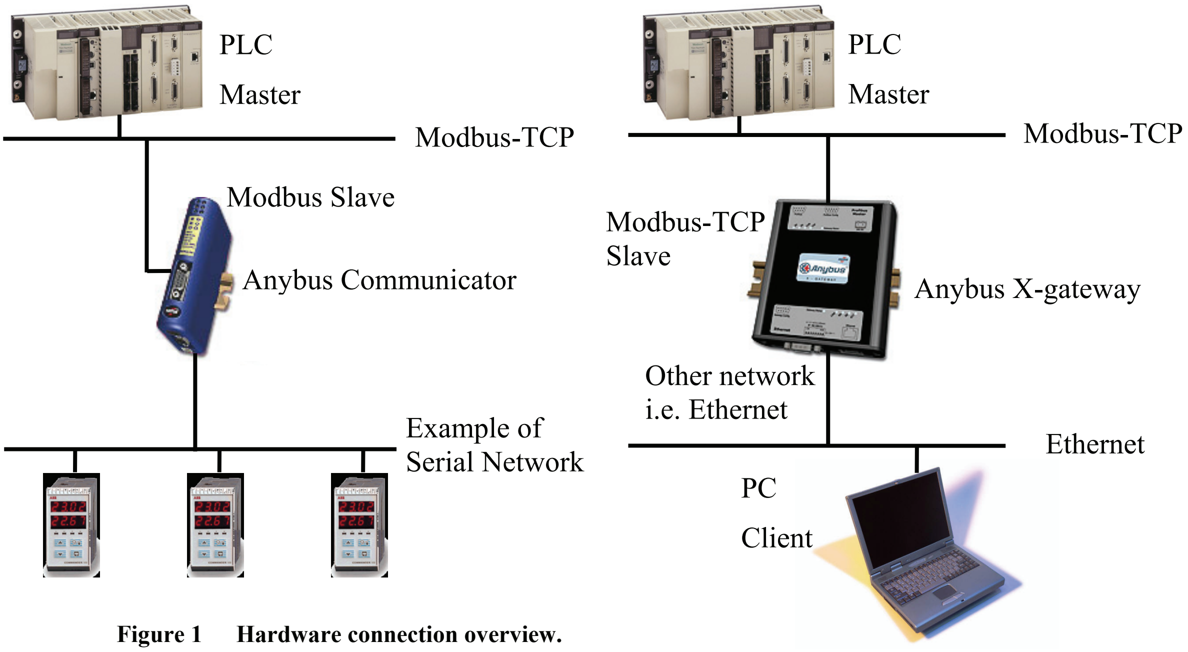


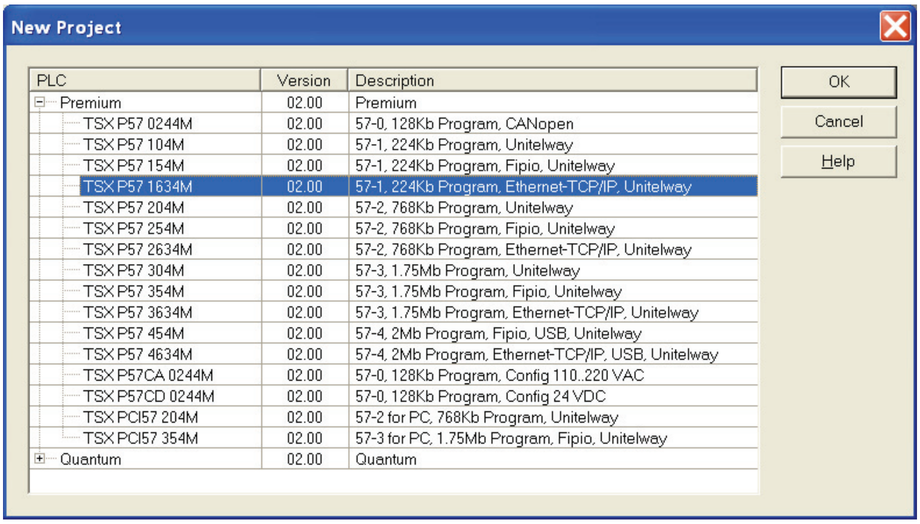
Figure 1 Hardware connection overview.

4. Modbus TCP configuration

To configure the PLC and the Modbus network the tool Unity Pro L is used. Firstly the PLC needs to be configured and secondly the Modbus network. Start the program and follow the steps below.

4.1. The PLC configuration

Start the Unity Pro L program and open the file menu and select new. In the window seen below, select the relevant PLC from the list, in this case the TSX P571634 model.

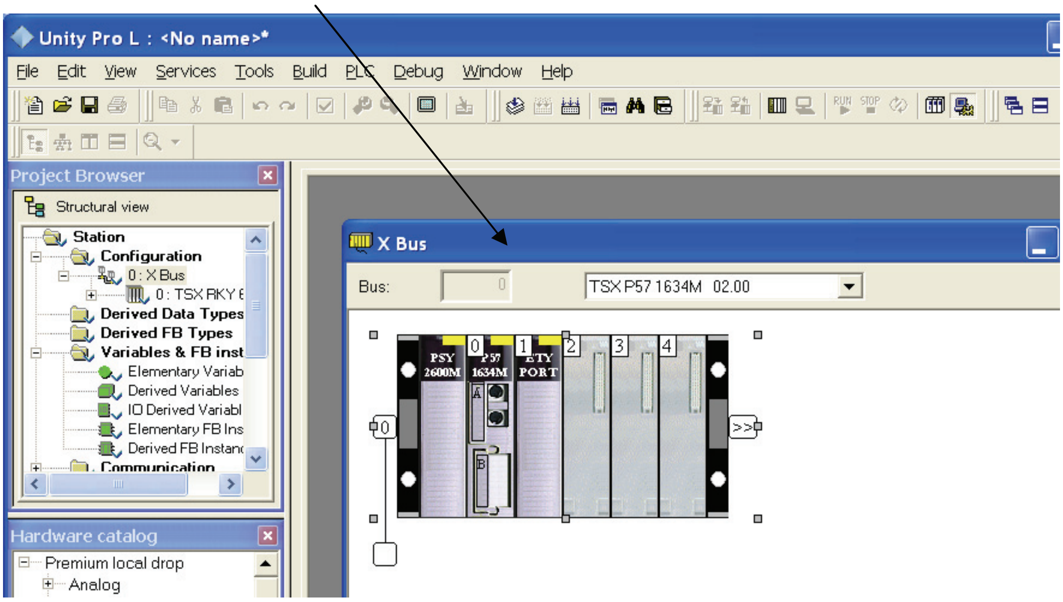


New Project

PLC	Version	Description
⊟ Premium	02.00	Premium
TSX P57 0244M	02.00	57-0, 128Kb Program, CANopen
TSX P57 104M	02.00	57-1, 224Kb Program, Unitelway
TSX P57 154M	02.00	57-1, 224Kb Program, Fipio, Unitelway
TSX P57 1634M	02.00	57-1, 224Kb Program, Ethernet-TCP/IP, Unitelway
TSX P57 204M	02.00	57-2, 768Kb Program, Unitelway
TSX P57 254M	02.00	57-2, 768Kb Program, Fipio, Unitelway
TSX P57 2634M	02.00	57-2, 768Kb Program, Ethernet-TCP/IP, Unitelway
TSX P57 304M	02.00	57-3, 1.75Mb Program, Unitelway
TSX P57 354M	02.00	57-3, 1.75Mb Program, Fipio, Unitelway
TSX P57 3634M	02.00	57-3, 1.75Mb Program, Ethernet-TCP/IP, Unitelway
TSX P57 454M	02.00	57-4, 2Mb Program, Fipio, USB, Unitelway
TSX P57 4634M	02.00	57-4, 2Mb Program, Ethernet-TCP/IP, USB, Unitelway
TSX P57CA 0244M	02.00	57-0, 128Kb Program, Config 110..220 VAC
TSX P57CD 0244M	02.00	57-0, 128Kb Program, Config 24 VDC
TSX PCI57 204M	02.00	57-2 for PC, 768Kb Program, Unitelway
TSX PCI57 354M	02.00	57-3 for PC, 1.75Mb Program, Fipio, Unitelway
⊞ Quantum	02.00	Quantum

Figure 2 Configuring the type of PLC.

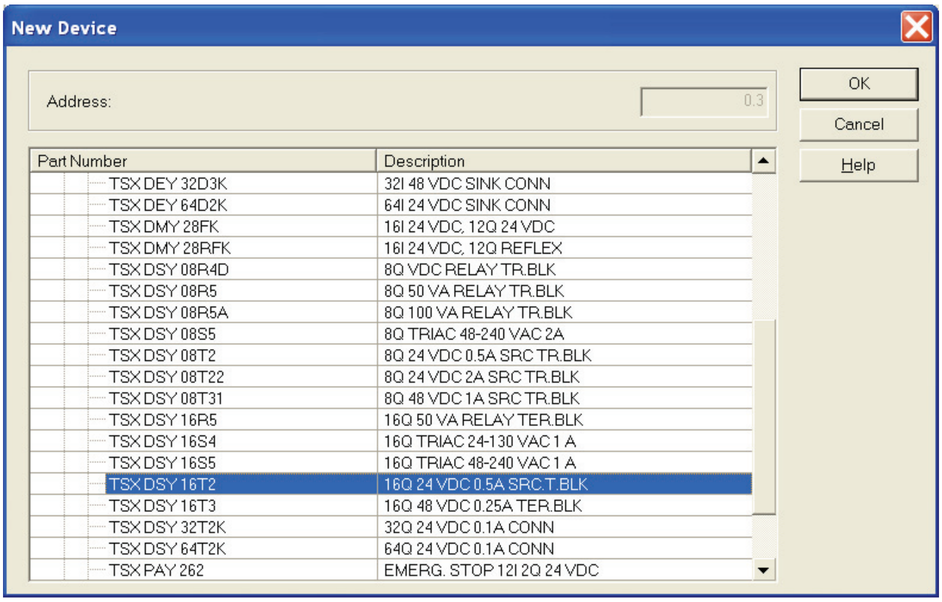
Press OK and the PLC with predefined modules is configured. Double click on the X-bus in the navigation list to the left and the window seen below will appear in a new window.



Bus: 0 TSX P57 1634M 02.00

Figure 3 Configuring the PLC modules.

Double click on the modules 2, 3 etcetera for configuring additional modules. In this case two discrete I/O modules are used, one for input and one for output data. After double clicking on one of the extra modules the list seen below will be shown.



New Device

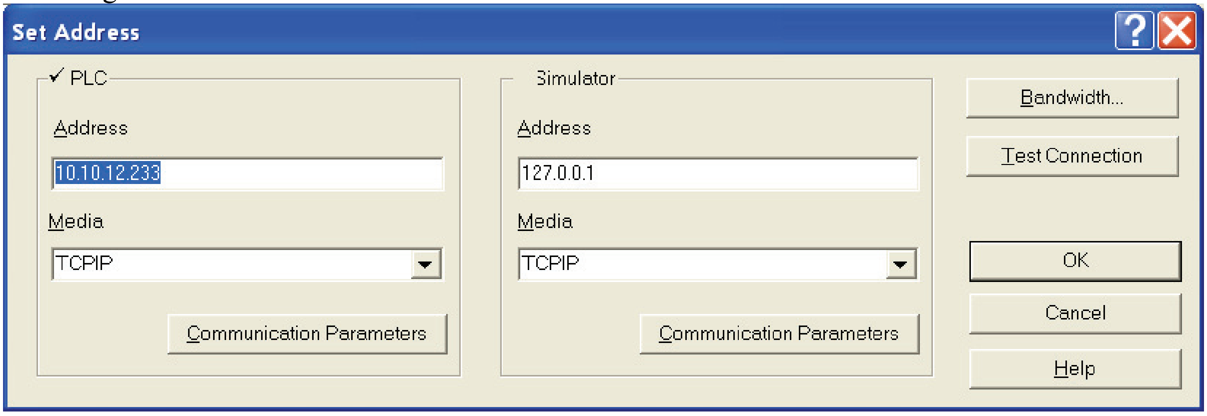
Address: 0.3

Part Number	Description
TSX DEY 32D3K	32I 48 VDC SINK CONN
TSX DEY 64D2K	64I 24 VDC SINK CONN
TSX DMY 28FK	16I 24 VDC, 12Q 24 VDC
TSX DMY 28RFK	16I 24 VDC, 12Q REFLEX
TSX DSY 08R4D	8Q VDC RELAY TR.BLK
TSX DSY 08R5	8Q 50 VA RELAY TR.BLK
TSX DSY 08R5A	8Q 100 VA RELAY TR.BLK
TSX DSY 08S5	8Q TRIAC 48-240 VAC 2A
TSX DSY 08T2	8Q 24 VDC 0.5A SRC TR.BLK
TSX DSY 08T22	8Q 24 VDC 2A SRC TR.BLK
TSX DSY 08T31	8Q 48 VDC 1A SRC TR.BLK
TSX DSY 16R5	16Q 50 VA RELAY TER.BLK
TSX DSY 16S4	16Q TRIAC 24-130 VAC 1 A
TSX DSY 16S5	16Q TRIAC 48-240 VAC 1 A
TSX DSY 16T2	16Q 24 VDC 0.5A SRC.T.BLK
TSX DSY 16T3	16Q 48 VDC 0.25A TER.BLK
TSX DSY 32T2K	32Q 24 VDC 0.1A CONN
TSX DSY 64T2K	64Q 24 VDC 0.1A CONN
TSX PAY 262	EMERG. STOP 12I 2Q 24 VDC

Figure 4 Configuring the I/O modules.

Select the desired module and click OK. In this case the I/O modules TSX DEY16D2 AND TSX DSY16T2 are added to the configuration. To configure additional modules repeat the previous steps.

To configure the IP settings of the PLC, open the PLC menu and select Set Address. The IP address is the address to which the configuration is to be downloaded to if using TCP/IP for the project transfer. The following window will be visible.



Set Address

✓ PLC

Address: 10.10.12.233

Media: TCPIP

Simulator

Address: 127.0.0.1

Media: TCPIP

Figure 5 Configuring the IP settings of the PLC.

Enter the desired IP address and select TCPIP and press OK. The IP address for the PLC is now configured.

4.2. The Modbus-TCP network configuration

The next step is to configure the Ethernet module. Right click on Networks in the navigation list to the left and select add network.

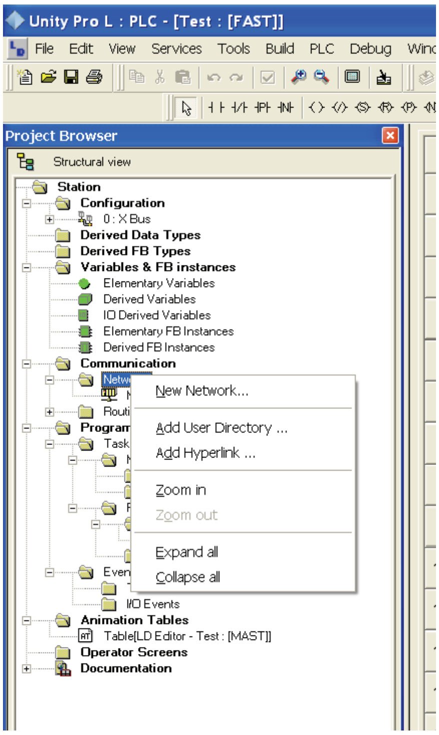
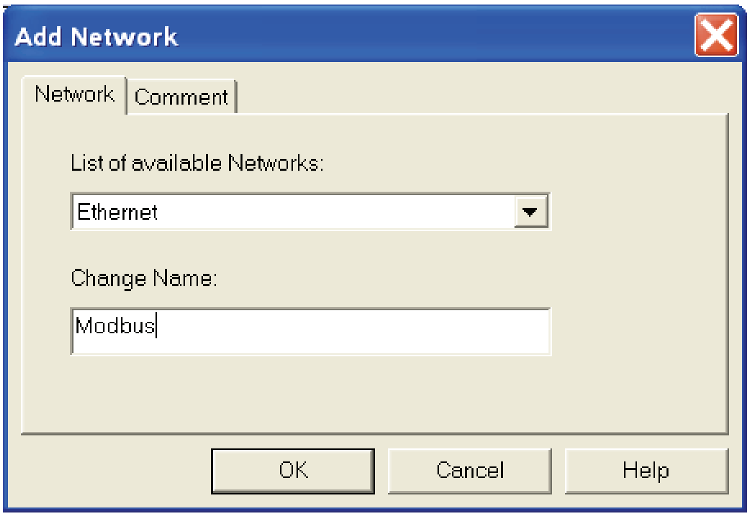


Figure 6 Adding a network.

Select Ethernet and choose a name. In this case the network is named Modbus.

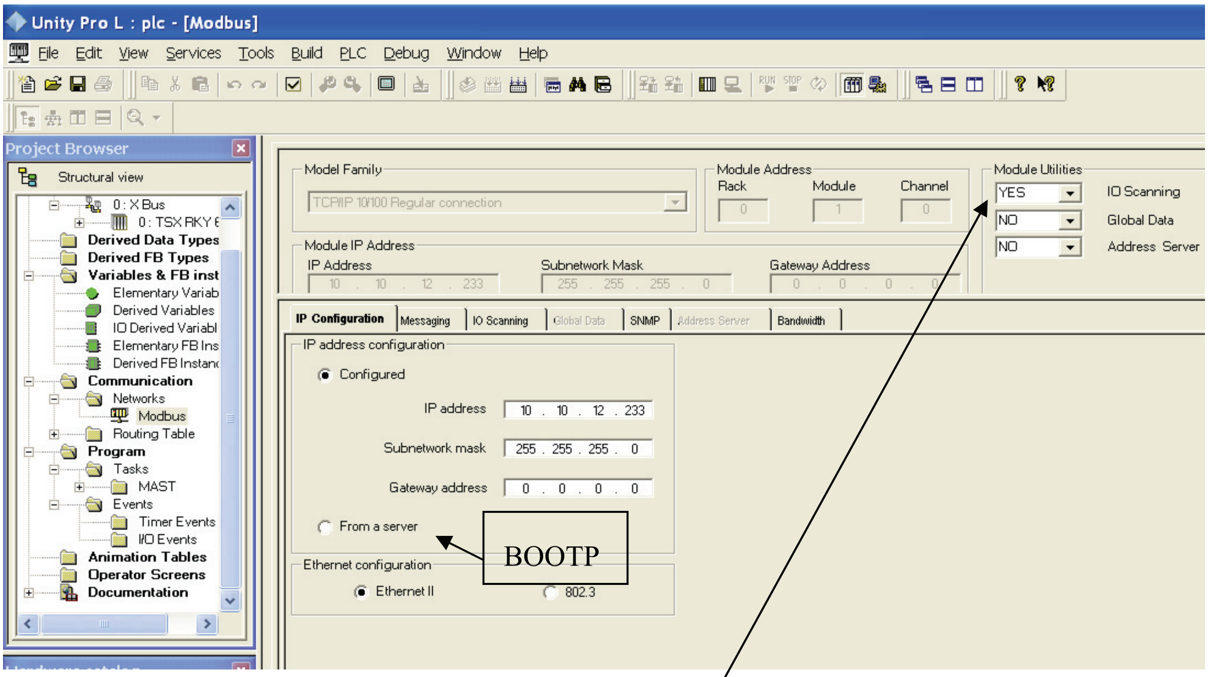


List of available Networks: Ethernet

Change Name: Modbus

Figure 7 Configuring the network.

Under Communication in the navigation list to the left double click on the network named Modbus as seen below. Enter the desired settings. By using the alternative “From a server” the IP settings are retrieved by BOOTP.



Model Family: TCP/IP 10/100 Regular connection

Module Address: Rack 0, Module 1, Channel 0

Module Utilities: IO Scanning YES; Global Data NO; Address Server NO

Module IP Address: IP Address 10 . 10 . 12 . 233; Subnetwork Mask 255 . 255 . 255 . 0; Gateway Address 0 . 0 . 0 . 0

IP address configuration:

● Configured

IP address 10 . 10 . 12 . 233

Subnetwork mask 255 . 255 . 255 . 0

Gateway address 0 . 0 . 0 . 0

○ From a server

Ethernet configuration: ● Ethernet II ○ 802.3

Figure 8 The IP configuration.

Enable the I/O scanning tab by selecting YES in the module utilities. The I/O scanning function uses the Modbus function 0x17, 23$_{dec}$, Read/Write Multiple Registers to access the data from the nodes. Using this function the user can configure data and transfer it between network nodes without specific programming. To be able to scan both Input and Output data the Anybus module needs to be configured for Anybus mode. The registers and coils can also be accessed in other ways, by commands. In each case the mapping is depending on what mode is selected. For a more detailed description see the note below.

Note:

Anybus mode: When the Anybus product is configured for Anybus mode both the Input and Output data can be accessed by the I/O scanning function. In Anybus mode it is also possible to access coils with the same memory location as of a register. In other words it is possible to access individual bits of a register. See the table below showing the Modbus addresses in Modbus and Anybus mode. For a more detailed description see the Appendix Modbus addresses.

Modbus mode: If the Modbus mode is used only the Output data can be accessed by the I/O scanning function.

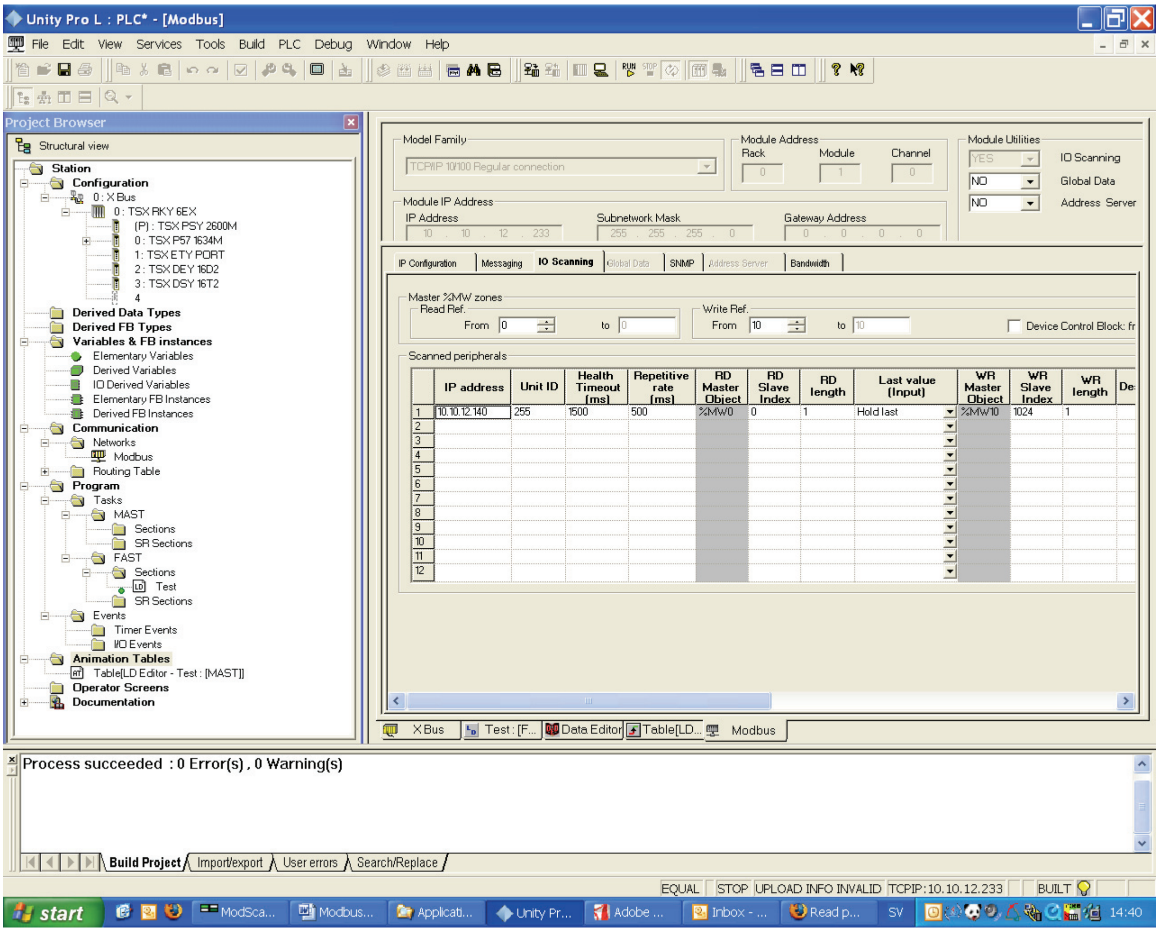
Data	**Anybus mode, Register**	**Register type**	**Modbus mode, Register**	**Register type**
Input data	1-1024	Input/Holding	1-1024	Input
Output data	1025-2048	Input/Holding	1-1024	Holding
Data	**Anybus mode, Coil**	**Coil type**	**Modbus mode, Coil**	**Coil type**
Input coil	1-16384	Input/Output	1-16384	Input
Output coil	16385-32768	Input/Output	1-16384	Output

Figure 9 Table showing the Modbus addresses used in Anybus and Modbus mode.

Note for the Anybus Communicator: *In the Communicator the registers 257 to 1024 are reserved. For a more detailed description see the Communicator User Manual.*

Note for the Anybus CompactCom: *Also note the Anybus and Modbus mode is not supported by the Anybus CompactCom. The module has a complete different I/O configuration and the maximum data exchange is 256 bytes in each direction refer to the CompactCom Software Design Guide.*

Then open the I/O scanning tile as shown below.



Model Family: TCP/IP 10/100 Regular connection

Module Address: Rack 0, Module 1, Channel 0

Module Utilities: IO Scanning YES; Global Data NO; Address Server NO

Module IP Address: IP Address 10 . 10 . 12 . 233; Subnetwork Mask 255 . 255 . 255 . 0; Gateway Address 0 . 0 . 0 . 0

Master %MW zones: Read Ref. From 0 to 0; Write Ref. From 10 to 10; ☐ Device Control Block: fr

Scanned peripherals

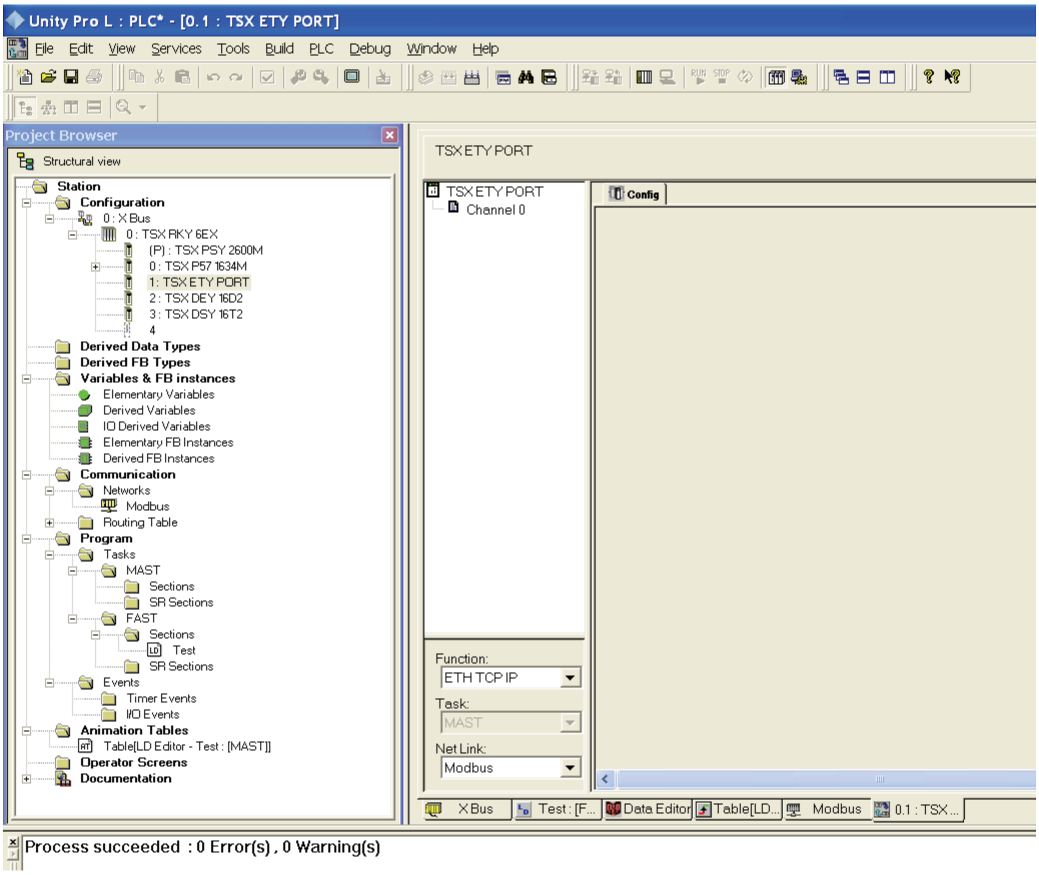
	IP address	Unit ID	Health Timeout [ms]	Repetitive rate [ms]	RD Master Object	RD Slave Index	RD length	Last value (Input)	WR Master Object	WR Slave Index	WR length	De
1	10.10.12.140	255	1500	500	%MW0	0	1	Hold last	%MW10	1024	1	
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												

Process succeeded : 0 Error(s) , 0 Warning(s)

Figure 10 Configuring the I/O scanning.

In this case the PLC is reading the Slave index 0 and writing to the index 1024 using the "zero notation". In the PLC the words with address 0 and 10 are used. The repetitive rate can be set to less than 500 ms as used in this example.

The next step is to double click on the Ethernet module and configure the network the PLC module is to be connected to. In this case the network named Modbus previously configured is selected. Open the configuration in the left navigation list and double click on the Ethernet module as seen below. Select the desired network and close the tile.



Function: ETH TCP IP

Task: MAST

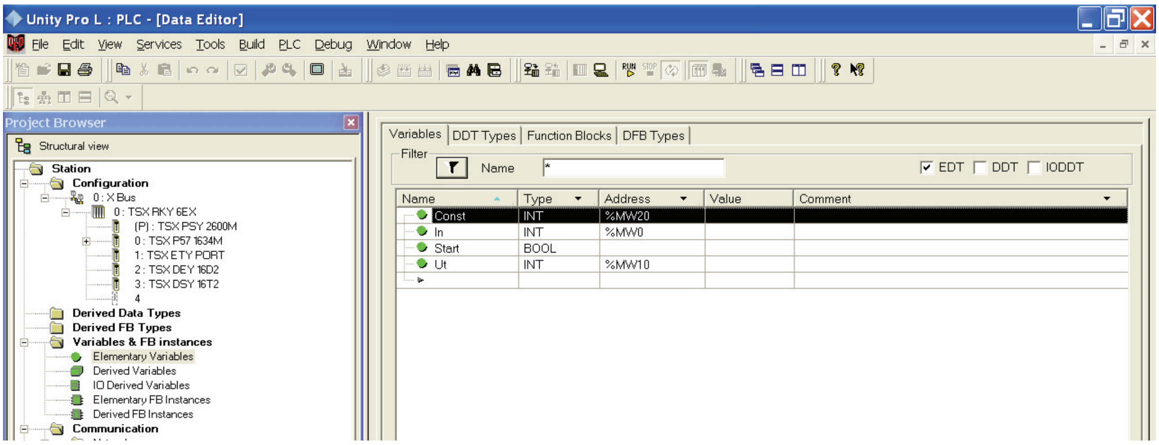
Net Link: Modbus

Process succeeded : 0 Error(s) , 0 Warning(s)

Figure 11 Configuring the PLC Ethernet module.

4.3. The PLC program

To test the communication between the slave and the master PLC a simple PLC program is used. The first step is to define the variables. Open the elementary variables and add the variables as seen below.

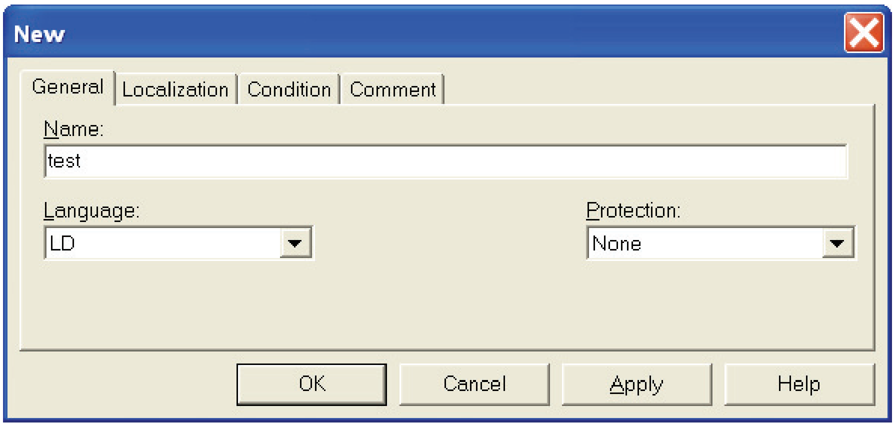


Name: * ☑ EDT ☐ DDT ☐ IODDT

Name	Type	Address	Value	Comment
Const	INT	%MW20		
In	INT	%MW0		
Start	BOOL			
Ut	INT	%MW10		

Figure 12 Defining the variables.

The second step is to put together a PLC program. The program is depending on the application; in this case a test program is used. Insert a new task of the FAST type and add a new section to the section folder. This is done by right clicking the Tasks folder under the Program folder and select add new task. Then right click on sections to add the program.



Name: test

Language: LD

Protection: None

Figure 13 Adding a new program.

The program is built in the ladder language and named test in this case.

An ADD-function, ADD_INT, is used in this case. The easiest way to insert the function is to use the Function Input Assistant.

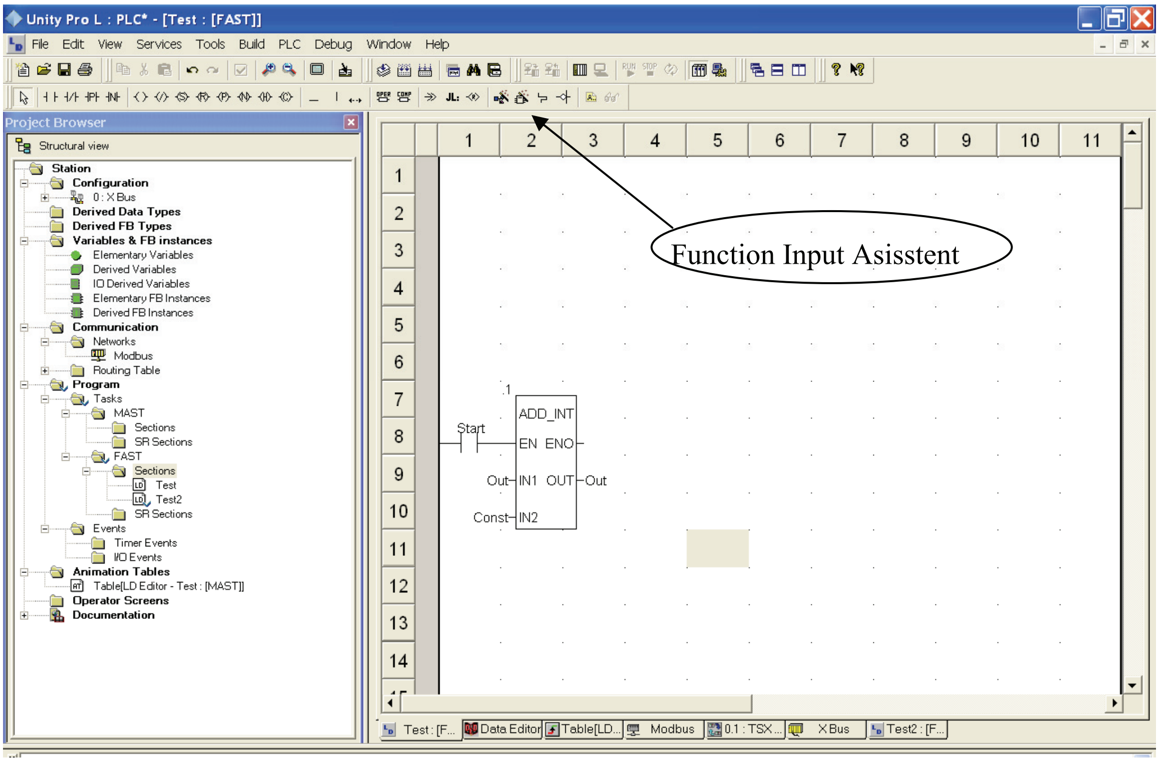


Figure 14 Adding the ADD function.

Click on the Function Input Assistant button and then on the browse button to the right in the row called FFB type.

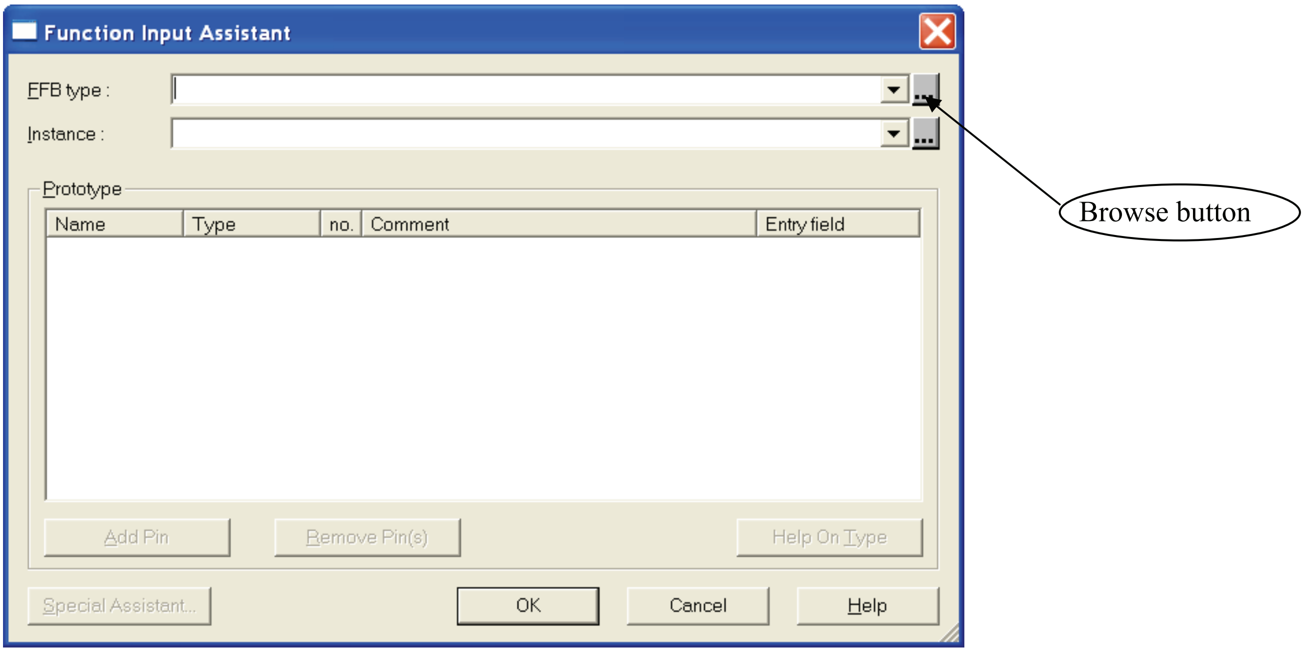
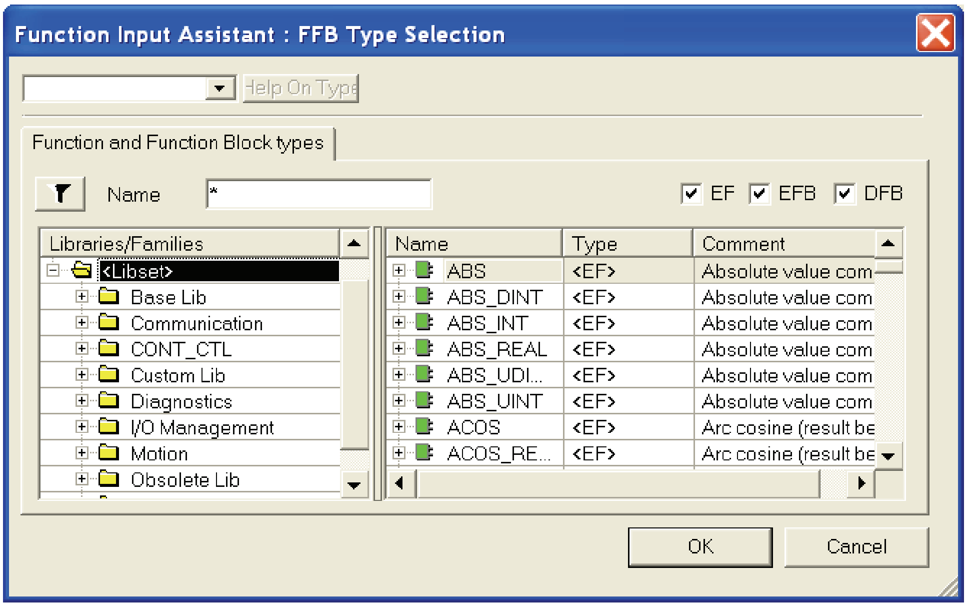


Figure 15 Browsing the Function library.

Double click on the Libset as seen below to expand the library. Select the ADD_INT function and press OK. Then click with the mouse on the working space to the right.



Function Input Assistant : FFB Type Selection

Function and Function Block types

Name: *　☑ EF　☑ EFB　☑ DFB

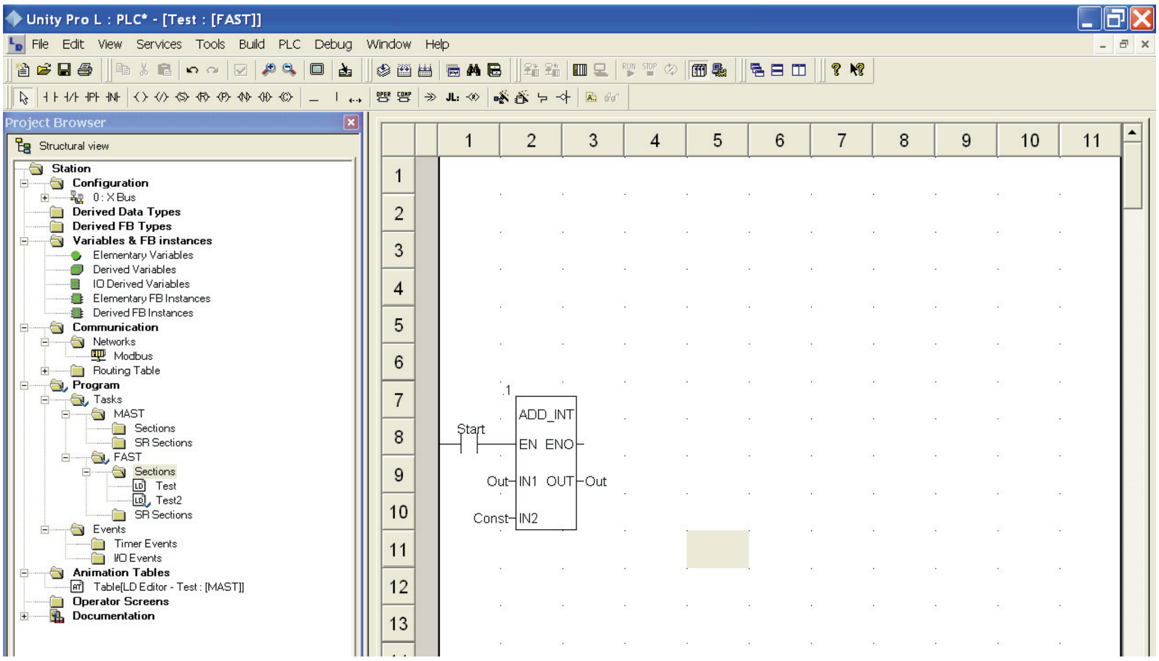
Libraries/Families:
- <Libset>
  - Base Lib
  - Communication
  - CONT_CTL
  - Custom Lib
  - Diagnostics
  - I/O Management
  - Motion
  - Obsolete Lib

Name	Type	Comment
ABS	<EF>	Absolute value com
ABS_DINT	<EF>	Absolute value com
ABS_INT	<EF>	Absolute value com
ABS_REAL	<EF>	Absolute value com
ABS_UDI...	<EF>	Absolute value com
ABS_UINT	<EF>	Absolute value com
ACOS	<EF>	Arc cosine (result be
ACOS_RE...	<EF>	Arc cosine (result be

Figure 16 Inserting the ADD function.

The start condition is controlled by the Boolean named Start. To insert the start condition mark the EN connector and click on the normally open contact button, the button is found in the top left corner. See the window below.

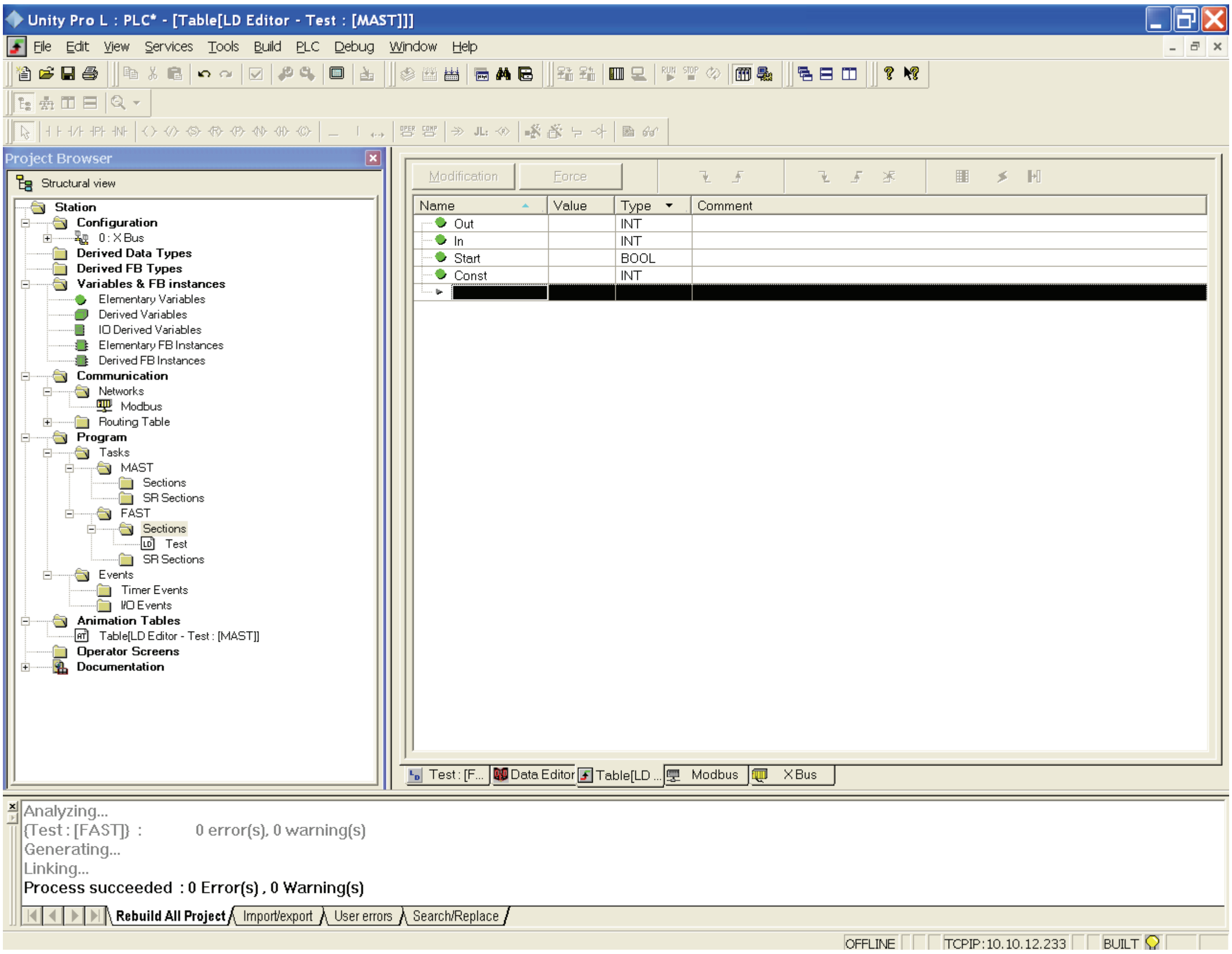
The next step is to connect the defined variables to the ADD function. Double click on the respective connector and select the variables as seen in the picture below. The Out variable is connected to the IN1 connector and the constant to the IN2 connector. The Out variable is also connected to the OUT connector.



Unity Pro L : PLC* - [Test : [FAST]]

Figure 17 Configuring the ADD function.

The next step is to insert a new animation table to monitor the variables. Right click on the Animation Tables in the navigation list to the left and select add a new Animation Table. Double click in the name column and add the desired variables.

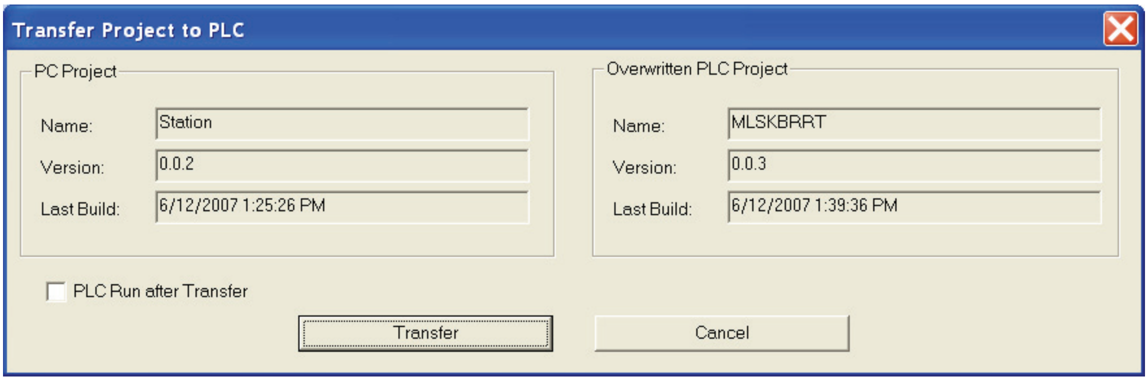


Name	Value	Type	Comment
Out		INT	
In		INT	
Start		BOOL	
Const		INT	

```
Analyzing...
{Test : [FAST]} :          0 error(s), 0 warning(s)
Generating...
Linking...
Process succeeded : 0 Error(s) , 0 Warning(s)
```

Figure 18 The animation table.

To be able to monitor the variables the program needs to be transferred to the PLC. Firstly the project needs to be compiled and saved. Press the rebuild all button and then save the project. After that open the PLC menu and select Transfer project to the PLC. Press the transfer button.



Transfer Project to PLC

PC Project

Name: Station

Version: 0.0.2

Last Build: 6/12/2007 1:25:26 PM

Overwritten PLC Project

Name: MLSKBRRT

Version: 0.0.3

Last Build: 6/12/2007 1:39:36 PM

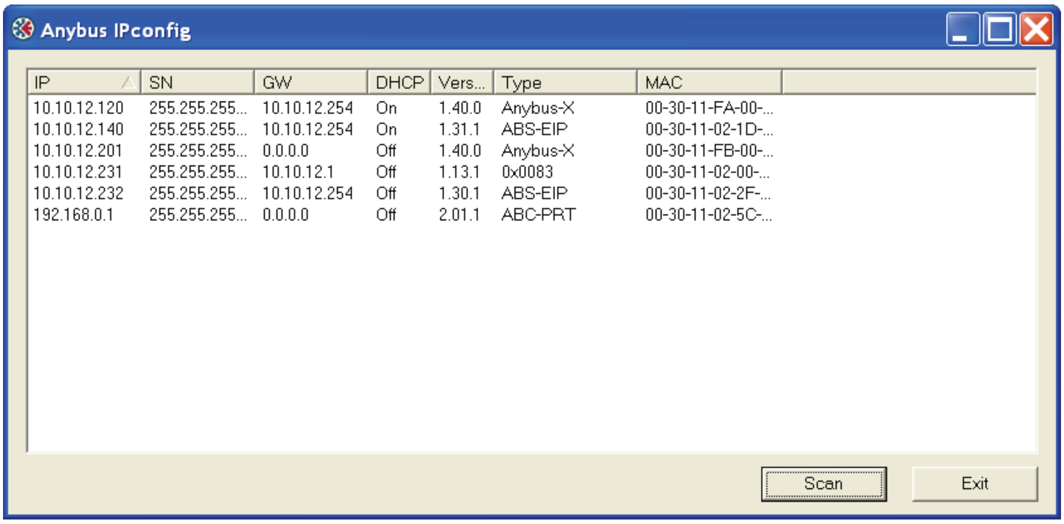
□ PLC Run after Transfer

Figure 19 Downloading the program.

5. Anybus configuration

5.1. IP settings

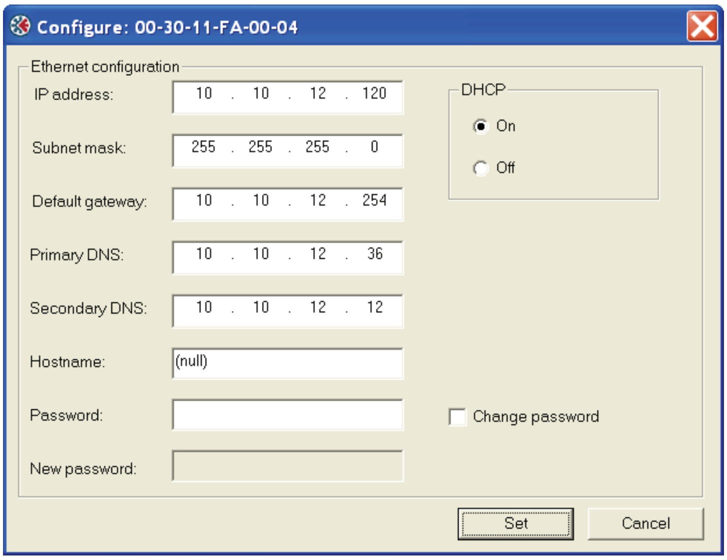
Make sure each node on the network has a unique IP address. The IP settings of the Anybus Modbus-TCP modules can be configured in various ways. It is recommended to use the Anybus IPconfig tool for configuring the IP-settings. The Anybus IPconfig tool can be used to configure the IP settings of all Anybus Modbus-TCP modules. The program can be downloaded at HMS[1] website. Start the program and the main window will be opened.



IP	SN	GW	DHCP	Vers...	Type	MAC
10.10.12.120	255.255.255...	10.10.12.254	On	1.40.0	Anybus-X	00-30-11-FA-00-...
10.10.12.140	255.255.255...	10.10.12.254	On	1.31.1	ABS-EIP	00-30-11-02-1D-...
10.10.12.201	255.255.255...	0.0.0.0	Off	1.40.0	Anybus-X	00-30-11-FB-00-...
10.10.12.231	255.255.255...	10.10.12.1	Off	1.13.1	0x0083	00-30-11-02-00-...
10.10.12.232	255.255.255...	10.10.12.254	Off	1.30.1	ABS-EIP	00-30-11-02-2F-...
192.168.0.1	255.255.255...	0.0.0.0	Off	2.01.1	ABC-PRT	00-30-11-02-5C-...

Figure 20 The Anybus IPconfig tool.

The program scans the network for Anybus Ethernet modules. By double clicking on the desired module the IP-settings can be entered. The settings can be configured manually or the DHCP function can be used. DHCP is activated by default.



IP address:	10 . 10 . 12 . 120
Subnet mask:	255 . 255 . 255 . 0
Default gateway:	10 . 10 . 12 . 254
Primary DNS:	10 . 10 . 12 . 36
Secondary DNS:	10 . 10 . 12 . 12
Hostname:	(null)
Password:	
New password:	

DHCP
● On
○ Off

□ Change password

Figure 21 Configuring the IP settings.

[1] www.anybus.com

As mentioned before there are other ways to configure the IP settings. See the notes below for more information.

Note for the Anybus-S Slave, and Communicator:
DIP switches 1-8 set the last part of the IP address, 192.168.0.xxx. The switches will be found next to the indication LEDs on the Communicator and the Slave Interface. The modules can also be configured by mailbox commands or by the ethcfg.cfg configuration file.

Note for the X-gateway:
The X-gateway can be configured by the DIP switches as described above or by the ethcfg.cfg configuration file.

Note for the Anybus PCI card:
The module can be configured by mailbox commands or by the ethcfg.cfg configuration file.

Note for the Anybus-IC:
For the Anybus–IC for Modbus-TCP the configuration is depending on the application. The IP address is set by the IP configuration parameters if the NA-bit is 1 or, if mounted, by switches if the NA-bit is 0. The IP address can also be configured by the network via EtherNet/IP via the TCP/IP Interface Object.

Note for the Anybus CompactCom:

The Anybus CompactCom can be configured by the network configuration object

For further instructions see the respective manual or appendix.

5.2. I/O configuration

The Anybus product has to be configured for the same I/O sizes as set up in the Modbus scanner configuration. The configuration procedure is depending on the type of module. See the notes below.

Note: The I/O sizes are depending on the application, the configured I/O sizes in this chapter are just examples.

Note for the Anybus-S Slave Interface and the Anybus PCI card:

The Anybus Slave Interface and the Anybus PCI card are configured by mailbox commands. Refer to the respective Fieldbus Appendix for details.

Note for the Anybus CompactCom and the Anybus-IC:

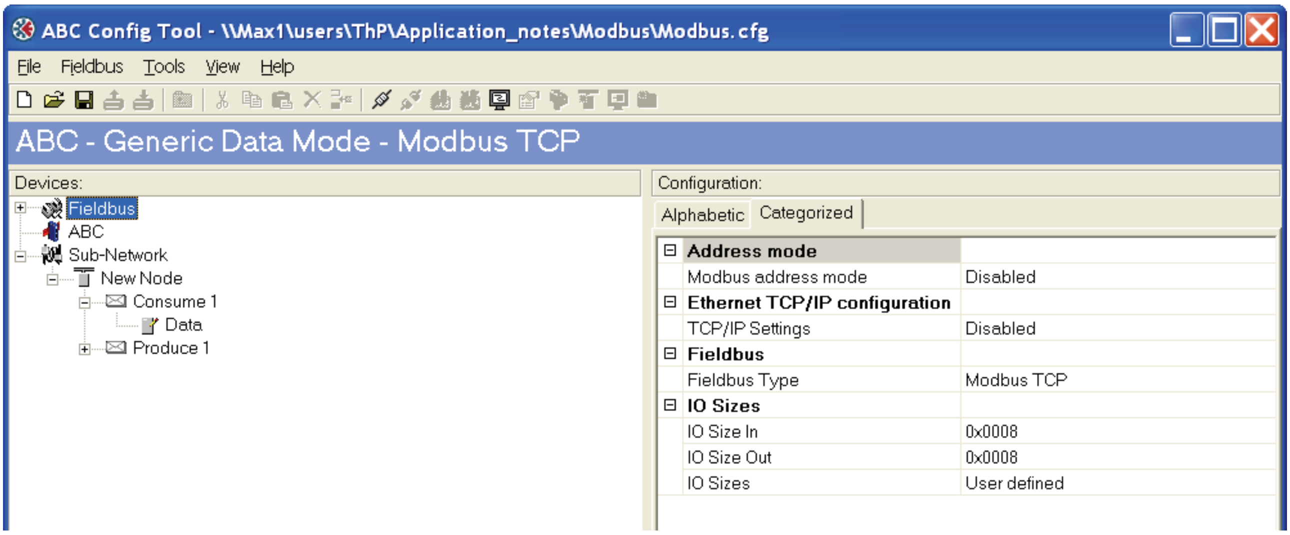
The Anybus CompactCom and the Anybus-IC are configured by messages send by the application. Refer to the respective Design Appendix for details.

Note for the Anybus Communicator and the Anybus X-gateway:

The configuration of the Anybus Communicator and the Anybus X-gateway is described in separate sections below.

5.3. Communicator configuration

To configure the Communicator start the ABC Config Tool and start a new project. Select the field bus Modbus-TCP and the desired I/O data size.

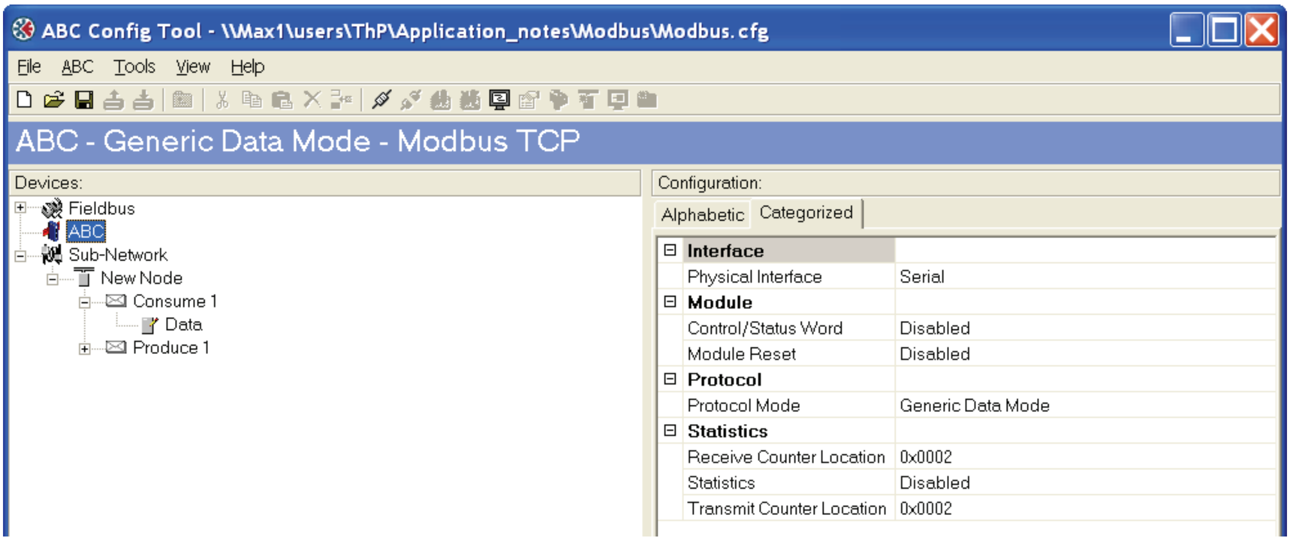


Address mode	
Modbus address mode	Disabled
Ethernet TCP/IP configuration	
TCP/IP Settings	Disabled
Fieldbus	
Fieldbus Type	Modbus TCP
IO Sizes	
IO Size In	0x0008
IO Size Out	0x0008
IO Sizes	User defined

Figure 22 Configuring the Field bus.

In this case 8 bytes are configured and the Modbus addressing mode is disabled. This means that the Anybus mode is enabled, see the chapter 4.2 and the Communicator User Manual.

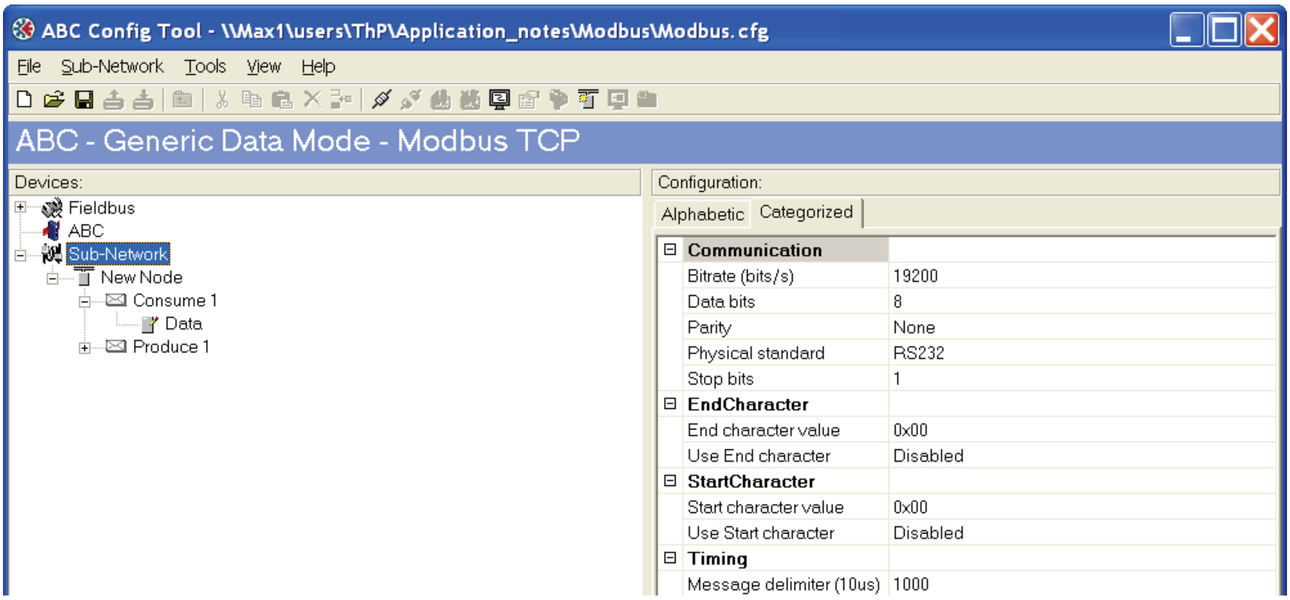
In this case a loop back dongle at the serial connection of the Communicator is connected. To loop data the generic data mode is selected; all other values are left at their defaults.



Interface	
Physical Interface	Serial
Module	
Control/Status Word	Disabled
Module Reset	Disabled
Protocol	
Protocol Mode	Generic Data Mode
Statistics	
Receive Counter Location	0x0002
Statistics	Disabled
Transmit Counter Location	0x0002

Figure 23 Configuring the Communicator.

The next step is to configure the sub-network.


ABC Config Tool - \\Max1\users\ThP\Application_notes\Modbus\Modbus.cfg
File
Sub-Network
Tools
View
Help
ABC - Generic Data Mode - Modbus TCP
Devices:
Configuration:
Alphabetic
Categorized

Devices:

- Fieldbus
- ABC
- Sub-Network
  - New Node
    - Consume 1
      - Data
    - Produce 1

Configuration:	
Communication	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS232
Stop bits	1
EndCharacter	
End character value	0x00
Use End character	Disabled
StartCharacter	
Start character value	0x00
Use Start character	Disabled
Timing	
Message delimiter (10us)	1000

Figure 24 Configuring the sub network.

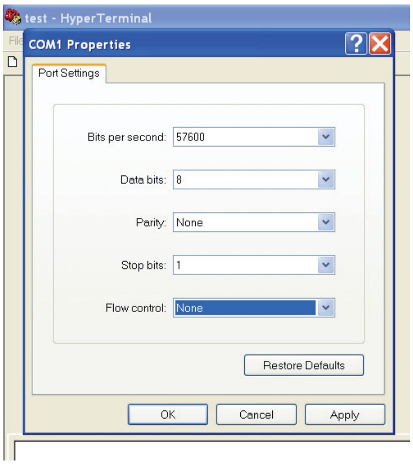
Right click on new node and add a consume and a produce transaction as shown above. Change the Offline timeout time to 200 ms and leave the settings for the produce transaction at the defaults. Right click on the produce and consume transaction respectively and select add data. In this case 8 bytes of data is used.

Note: *The update time for the produce transaction is to be set to less than the offline timeout time for the consume transaction; in this case the update time is set to 100 ms.*

For a more detailed description see the Communicator user manual.

5.4. X-gateway configuration

Use the HyperTerminal on a PC and configure the X-gateway. Connect a serial cable between the PC and the config port on the X-gateway. Open the "File" menu and click on new, choose the desired COM port and then click on OK. The following window will appear.



Bits per second: 57600

Data bits: 8

Parity: None

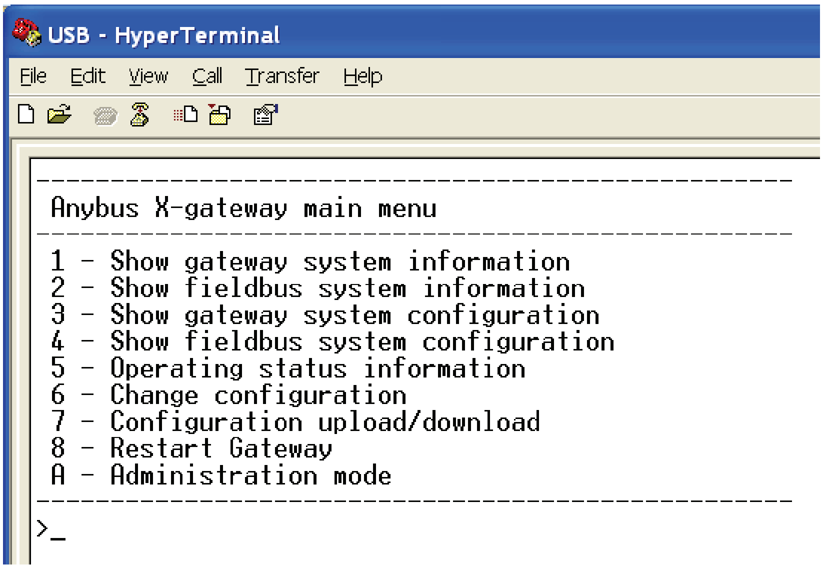
Stop bits: 1

Flow control: None

Figure 25 Configuring the connection in the HyperTerminal.

Make sure the settings are identical to those shown in the window above. Alternatively download a HyperTerminal session file from the HMS website[2], double click on it and select COM port.

Connect and press ESC and the following menu will appear.

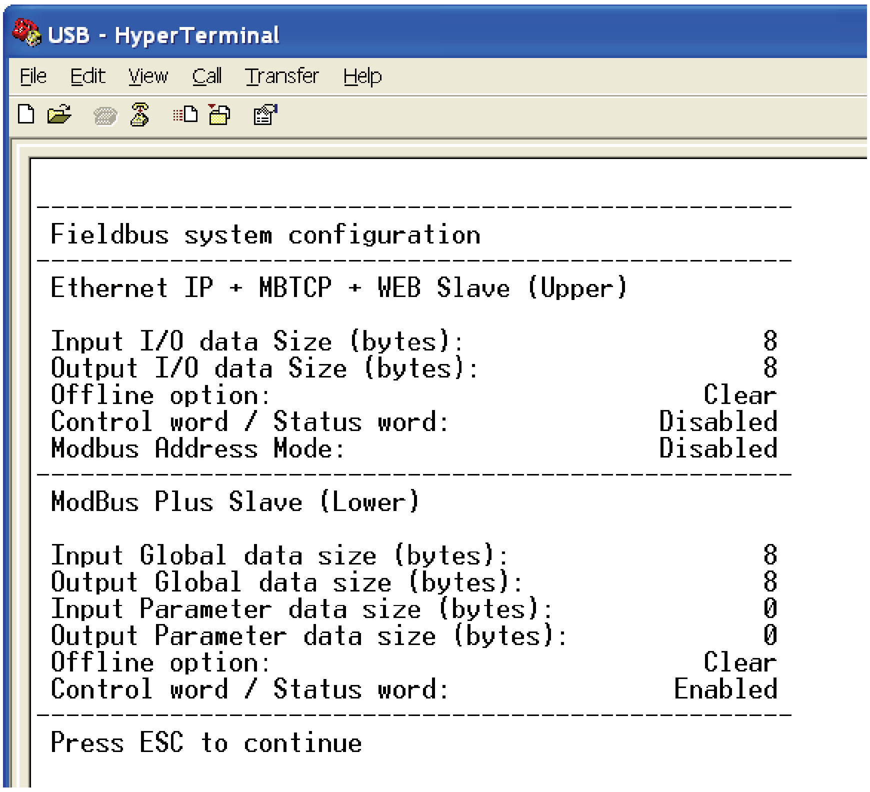


```
----------------------------------------------------
 Anybus X-gateway main menu
----------------------------------------------------
 1 - Show gateway system information
 2 - Show fieldbus system information
 3 - Show gateway system configuration
 4 - Show fieldbus system configuration
 5 - Operating status information
 6 - Change configuration
 7 - Configuration upload/download
 8 - Restart Gateway
 A - Administration mode
----------------------------------------------------
>_
```

Figure 26 Anybus X-gateway Main menu.

Press 6 and enter the desired configuration. The figure below shows an example; in this case an Ethernet Modbus-TCP/Modbus Plus X-gateway is used and 8 bytes of I/O data is configured. The Modbus addressing mode is disabled. This means that the Anybus mode is enabled. See the chapter 4.2 and also the X-gateway Ethernet Slave Interface Adapter Interface, Network Interface Addendum for a more detailed description.

[2] www.anybus.com



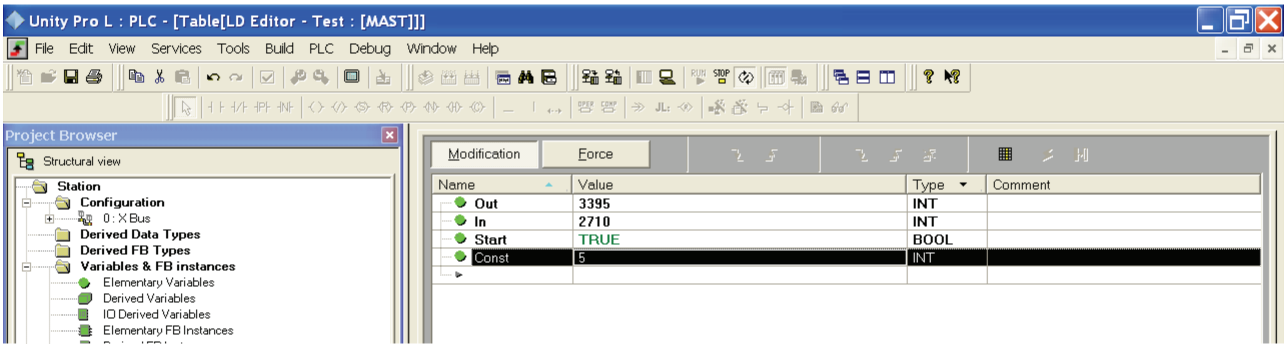
```
 -----------------------------------------------------
  Fieldbus system configuration
 -----------------------------------------------------
  Ethernet IP + MBTCP + WEB Slave (Upper)

  Input I/O data Size (bytes):                      8
  Output I/O data Size (bytes):                     8
  Offline option:                               Clear
  Control word / Status word:                Disabled
  Modbus Address Mode:                       Disabled
 -----------------------------------------------------
  ModBus Plus Slave (Lower)

  Input Global data size (bytes):                   8
  Output Global data size (bytes):                  8
  Input Parameter data size (bytes):                0
  Output Parameter data size (bytes):               0
  Offline option:                               Clear
  Control word / Status word:                 Enabled
 -----------------------------------------------------
  Press ESC to continue
```

Figure 27 The X-gateway configuration.

6. Testing

After the configuration the testing of the network can be done.

Open the PLC menu again and select the command connect. Then press the run button in the top navigation list. In the animation window the variables can now be monitored. Open the animation table and press the modification button and modify the variables.

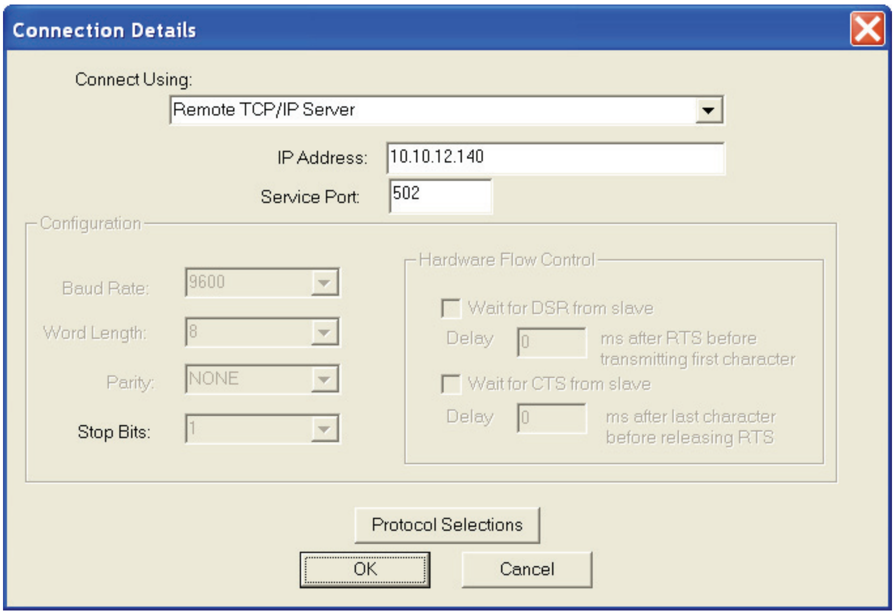


Name	Value	Type	Comment
Out	3395	INT	
In	2710	INT	
Start	TRUE	BOOL	
Const	5	INT	

Figure 28 Modifying the variables.

The constant used is set to five, in other words the variable OUT is increased by five for each program scan cycle. To start the program set the value of the Boolean Start to one. The In variable will then obtain a new value as seen above.

To see the data being transmitted on the Modbus network the tool Modscan32 is used. Open the connection menu and select connect as seen below.



Connect Using: Remote TCP/IP Server

IP Address: 10.10.12.140

Service Port: 502

Configuration

Baud Rate: 9600

Word Length: 8

Parity: NONE

Stop Bits: 1

Hardware Flow Control

☐ Wait for DSR from slave

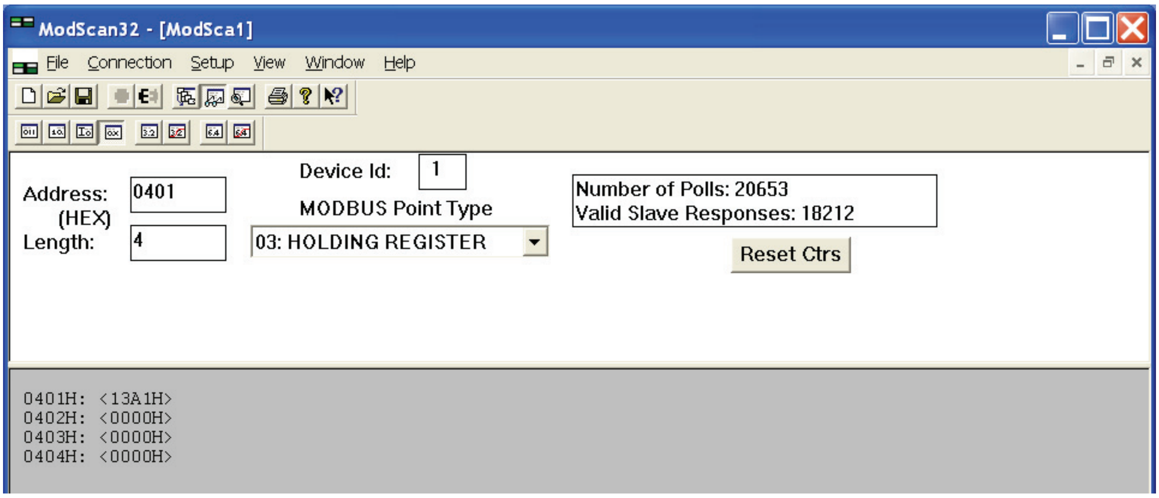
Delay 0 ms after RTS before transmitting first character

☐ Wait for CTS from slave

Delay 0 ms after last character before releasing RTS

Figure 29 Configuring the connection.

Select Remote TCP/IP Server and enter the IP address for the slave, in this case 10.10.12.140. After pressing OK, select 03 Holding Register and enter 401 in the address field as seen below.



ModScan32 - [ModSca1]

Address: 0401 (HEX)

Length: 4

Device Id: 1

MODBUS Point Type: 03: HOLDING REGISTER

Number of Polls: 20653
Valid Slave Responses: 18212

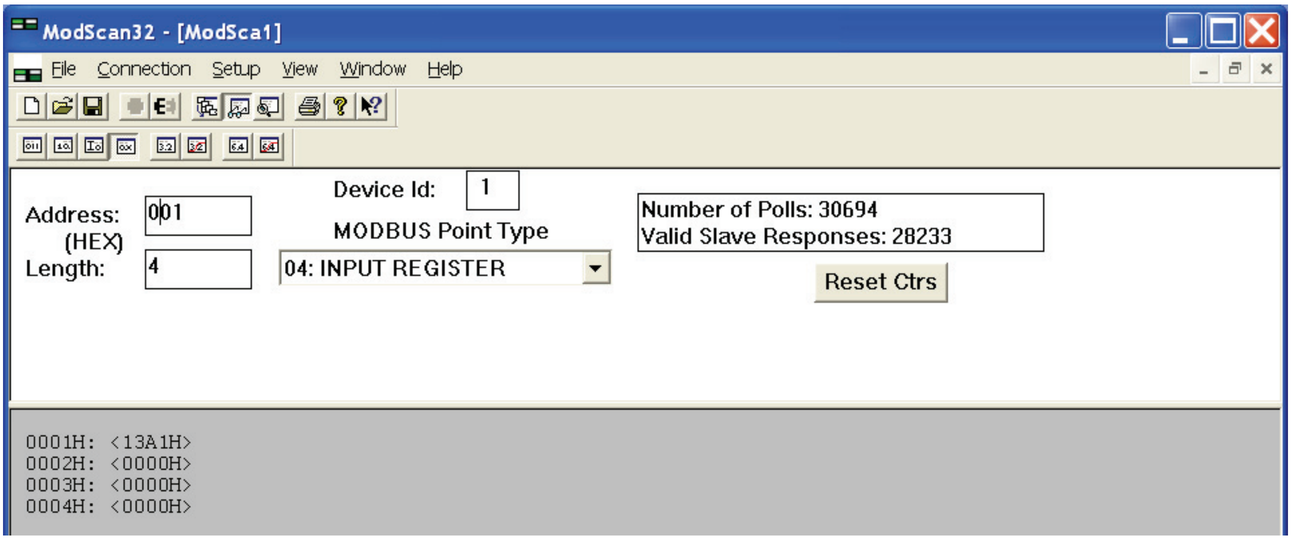
```
0401H: <13A1H>
0402H: <0000H>
0403H: <0000H>
0404H: <0000H>
```

Figure 30 Scanning the Holding register.

The Register 401 Hex is 1025 in decimal format, in other words the register corresponds to the one written by the PLC. The value seen is the same as of variable Out.

Note: In ModScan32 the starting address is 1 and not 0 as in the PLC.

In the same way the variable In can be monitored by scanning the register numbered 1. Change the Modbus Point Type to 04 Input Register and enter 1 in the address field as seen below.



ModScan32 - [ModSca1]

Address: 0001 (HEX)

Length: 4

Device Id: 1

MODBUS Point Type: 04: INPUT REGISTER

Number of Polls: 30694
Valid Slave Responses: 28233

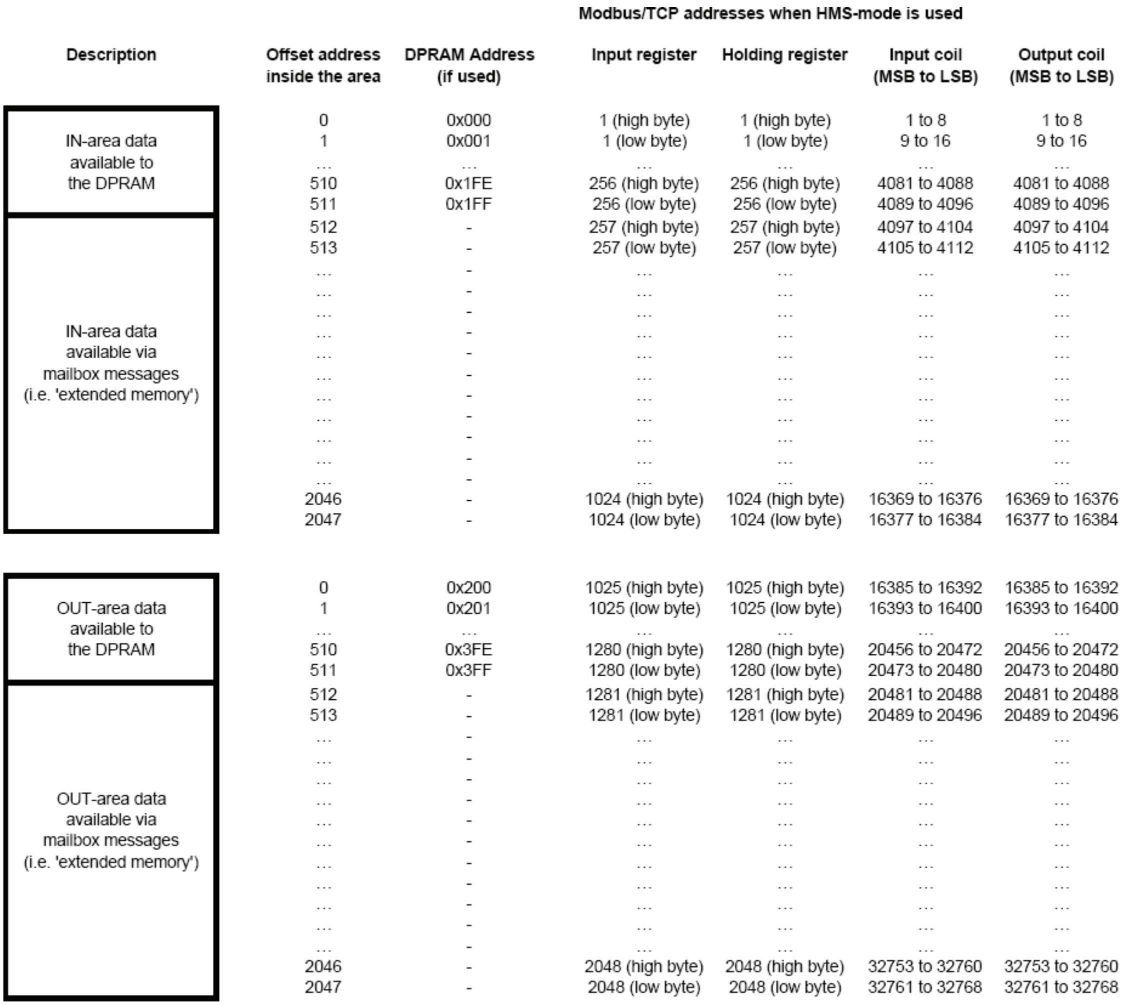
```
0001H: <13A1H>
0002H: <0000H>
0003H: <0000H>
0004H: <0000H>
```

Figure 31 Scanning the Input register.

Appendix

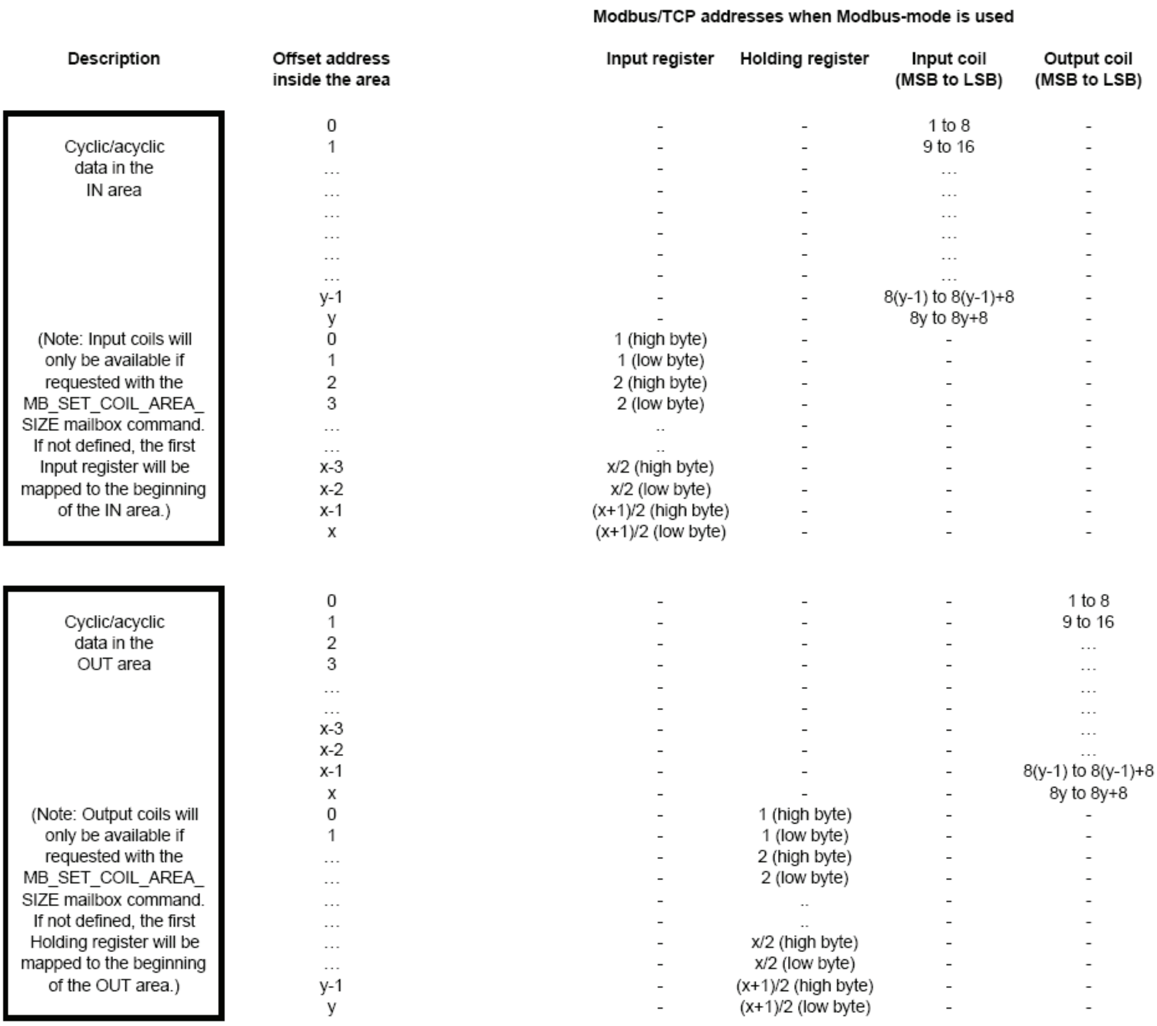
Modbus-TCP addresses

The two tables below describe the Modbus-TCP addresses when using respectively Modbus mode and Anybus mode. HMS mode is analogue to Anybus mode as mentioned in the table below. The tables shows the Register and Coil addresses, register type but also the offset address in the area and the DPRAM address in the Anybus module.



Modbus/TCP addresses when HMS-mode is used

Description	Offset address inside the area	DPRAM Address (if used)	Input register	Holding register	Input coil (MSB to LSB)	Output coil (MSB to LSB)
IN-area data available to the DPRAM	0	0x000	1 (high byte)	1 (high byte)	1 to 8	1 to 8
	1	0x001	1 (low byte)	1 (low byte)	9 to 16	9 to 16
	...	...	...	...	...	...
	510	0x1FE	256 (high byte)	256 (high byte)	4081 to 4088	4081 to 4088
	511	0x1FF	256 (low byte)	256 (low byte)	4089 to 4096	4089 to 4096
IN-area data available via mailbox messages (i.e. 'extended memory')	512	-	257 (high byte)	257 (high byte)	4097 to 4104	4097 to 4104
	513	-	257 (low byte)	257 (low byte)	4105 to 4112	4105 to 4112
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	2046	-	1024 (high byte)	1024 (high byte)	16369 to 16376	16369 to 16376
	2047	-	1024 (low byte)	1024 (low byte)	16377 to 16384	16377 to 16384
OUT-area data available to the DPRAM	0	0x200	1025 (high byte)	1025 (high byte)	16385 to 16392	16385 to 16392
	1	0x201	1025 (low byte)	1025 (low byte)	16393 to 16400	16393 to 16400
	...	...	...	...	...	...
	510	0x3FE	1280 (high byte)	1280 (high byte)	20456 to 20472	20456 to 20472
	511	0x3FF	1280 (low byte)	1280 (low byte)	20473 to 20480	20473 to 20480
OUT-area data available via mailbox messages (i.e. 'extended memory')	512	-	1281 (high byte)	1281 (high byte)	20481 to 20488	20481 to 20488
	513	-	1281 (low byte)	1281 (low byte)	20489 to 20496	20489 to 20496
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	...	-	...	...	...	...
	2046	-	2048 (high byte)	2048 (high byte)	32753 to 32760	32753 to 32760
	2047	-	2048 (low byte)	2048 (low byte)	32761 to 32768	32761 to 32768



Description	Offset address inside the area	Modbus/TCP addresses when Modbus-mode is used			
		Input register	Holding register	Input coil (MSB to LSB)	Output coil (MSB to LSB)
Cyclic/acyclic data in the IN area (Note: Input coils will only be available if requested with the MB_SET_COIL_AREA_ SIZE mailbox command. If not defined, the first Input register will be mapped to the beginning of the IN area.)	0	-	-	1 to 8	-
	1	-	-	9 to 16	-
	…	-	-	…	-
	…	-	-	…	-
	…	-	-	…	-
	…	-	-	…	-
	…	-	-	…	-
	…	-	-	…	-
	y-1	-	-	8(y-1) to 8(y-1)+8	-
	y	-	-	8y to 8y+8	-
	0	1 (high byte)	-	-	-
	1	1 (low byte)	-	-	-
	2	2 (high byte)	-	-	-
	3	2 (low byte)	-	-	-
	…	..	-	-	-
	…	..	-	-	-
	x-3	x/2 (high byte)	-	-	-
	x-2	x/2 (low byte)	-	-	-
	x-1	(x+1)/2 (high byte)	-	-	-
	x	(x+1)/2 (low byte)	-	-	-
Cyclic/acyclic data in the OUT area (Note: Output coils will only be available if requested with the MB_SET_COIL_AREA_ SIZE mailbox command. If not defined, the first Holding register will be mapped to the beginning of the OUT area.)	0	-	-	-	1 to 8
	1	-	-	-	9 to 16
	2	-	-	-	…
	3	-	-	-	…
	…	-	-	-	…
	…	-	-	-	…
	x-3	-	-	-	…
	x-2	-	-	-	…
	x-1	-	-	-	8(y-1) to 8(y-1)+8
	x	-	-	-	8y to 8y+8
	0	-	1 (high byte)	-	-
	1	-	1 (low byte)	-	-
	…	-	2 (high byte)	-	-
	…	-	2 (low byte)	-	-
	…	-	..	-	-
	…	-	..	-	-
	…	-	x/2 (high byte)	-	-
	…	-	x/2 (low byte)	-	-
	y-1	-	(x+1)/2 (high byte)	-	-
	y	-	(x+1)/2 (low byte)	-	-