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

Document history

Revision	Date	Description	Author
1.00	2007-10-17	Created	Thorbjörn Palm
1.01	2007-12-03	Updated layout	Thorbjörn Palm
1.02	2007-12-13	Minor corrections	Thorbjörn Palm

More information about the network and products

For further information about the Anybus products, please consult the HMS webpage, www.anybus.com. The latest manuals, etcetera can be downloaded from that location.

The Modbus user organisation has a webpage on the Internet, 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-RTU
Anybus Communicator	Modbus-RTU
Anybus-S Slave	Modbus-RTU
Anybus-CompactCom	Modbus-RTU

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	-
PCMCIA card for Modbus connection	TSX SCP 114	-
PLC software	Unity Pro L	2.1
Communicator configuration software	ABC Config Tool	2.32
X-gateway terminal software	HyperTerminal	5.1
X-gateway Network Interface Addendum	Anybus X-gateway Modbus RTU Slave Interface, Network Interface Addendum	1.03
X-gateway User Manual	X-gateway Generic User Manual	1.10
Communicator User Manual	Anybus Communicator for Modbus RTU, User Manual	2.52
Slave Fieldbus Appendix	Anybus-S Modbus RTU, Fieldbus Appendix	1.05
CompactCom Network Interface Appendix	CompactCom Modbus RTU, Network Interface Appendix	1.10
Power supply 24VDC	n.a.	n.a
Configuration cables	n.a.	n.a.

3. Solution overview

This application note describes how to configure an Anybus Modbus-RTU 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.

The configuration is described in two steps.

1. At first the PLC configuration is explained.
2. Secondly the configuration of the communication parameters and the I/O data of the Anybus module are described.

The pin configuration on the Modbus-RTU connector is the same on the Schneider PLC as on the Anybus module. Also make sure the network is terminated in a correct way.

Note: This document is valid for all Anybus Modbus RTU 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-RTU networks and HMS Communicator and X-gateway.

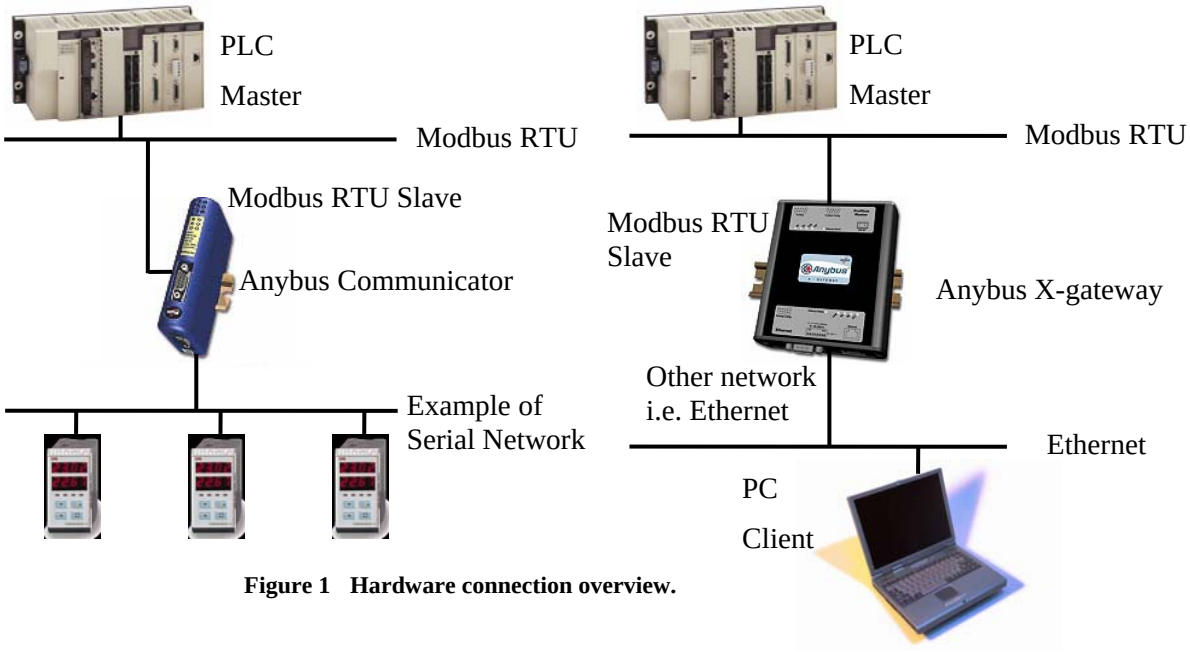


Figure 1 Hardware connection overview.

4. Modbus RTU 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.

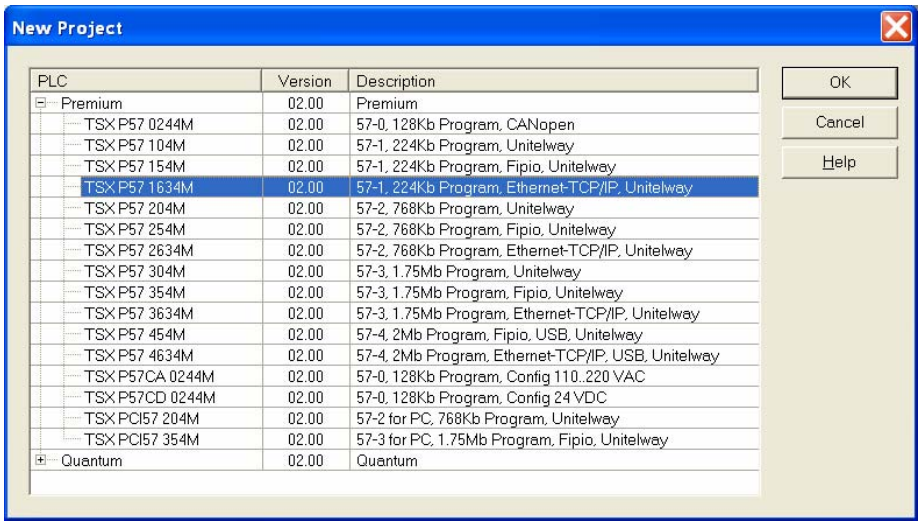


Figure 2 Configuring the type of PLC.

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.

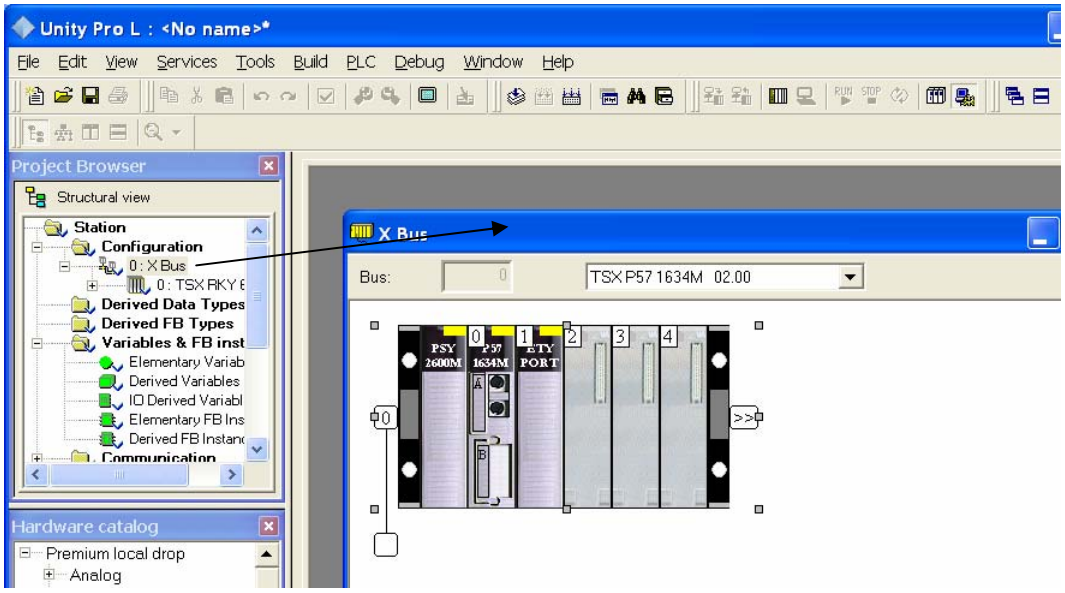


Figure 3 Configuring the PLC modules.

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 additional modules the list seen below will be shown.

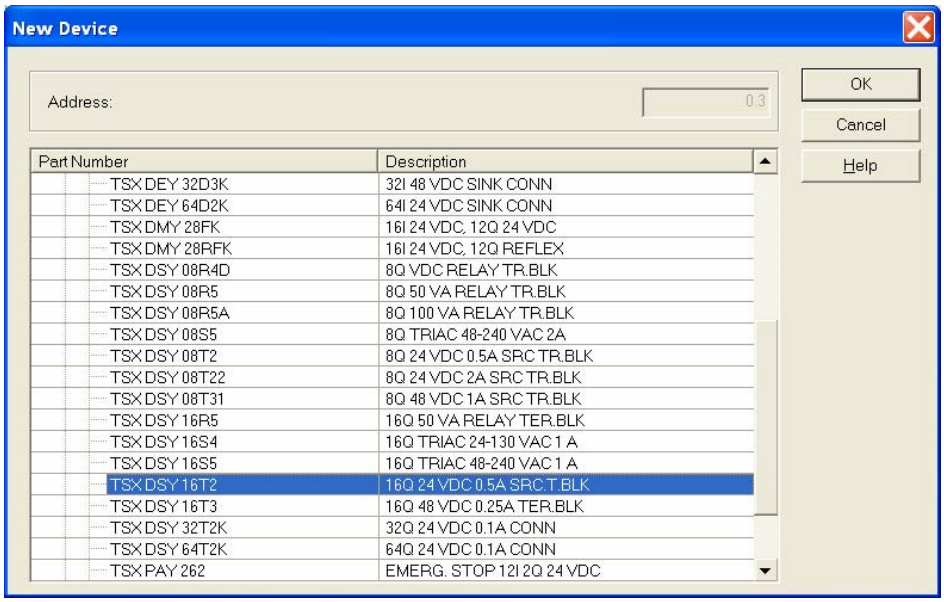


Figure 4 Configuring the I/O modules.

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.

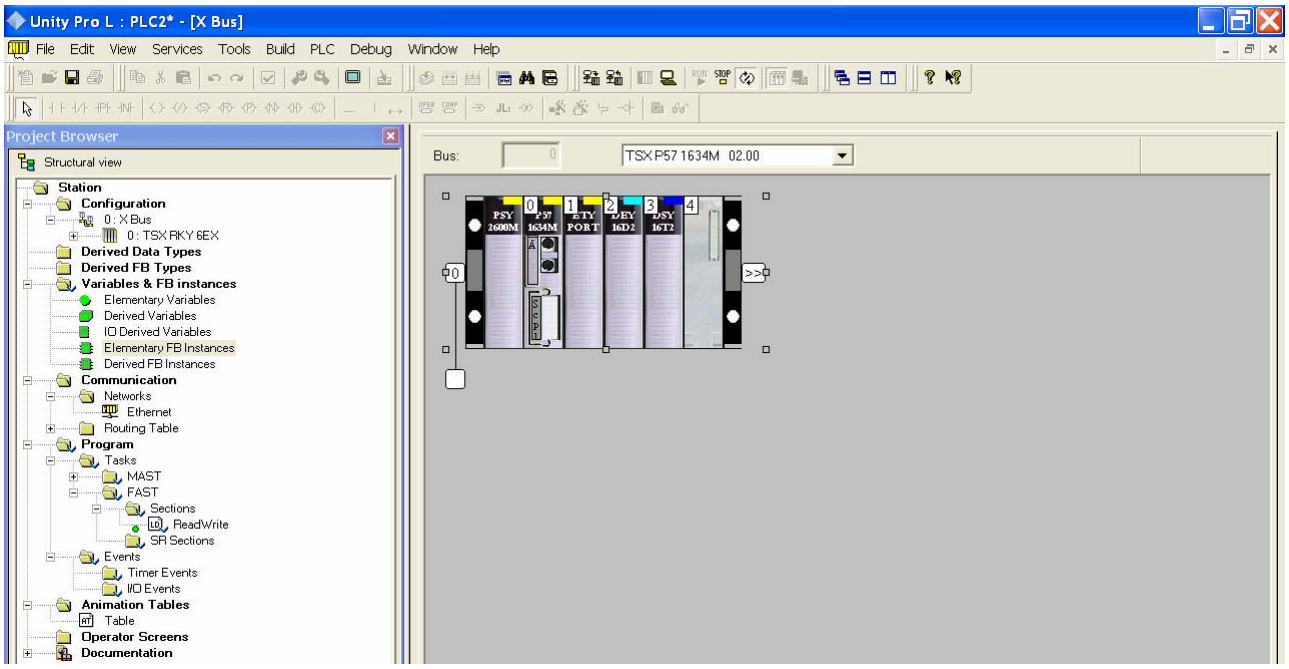


Figure 5 The module configuration of the PLC.

In this case the PLC is configured by the Ethernet connection. If you are using the serial connection it is not necessary to configure the Ethernet port. In that case please step forward to chapter 4.2, The Modbus-RTU master configuration.

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.

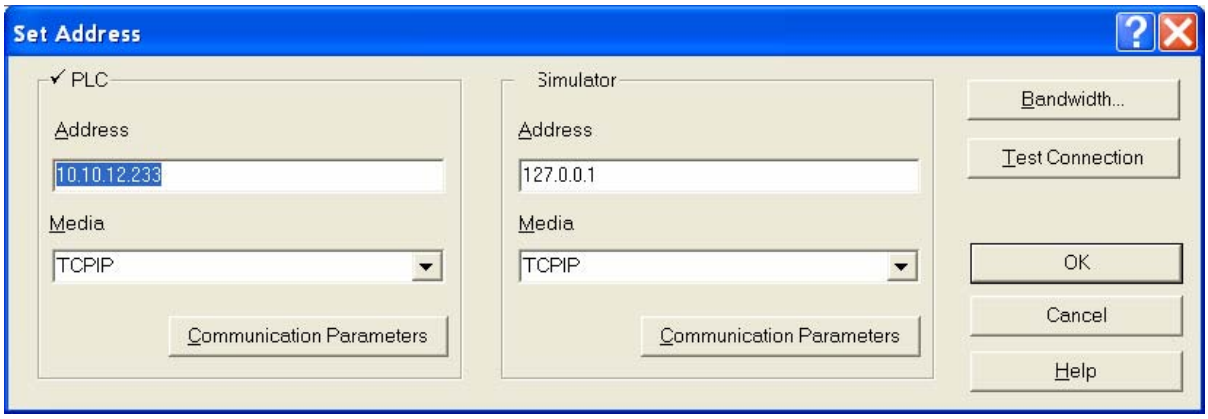


Figure 6 Configuring the IP settings of the PLC.

Enter the desired IP address and select TCPIP and press OK. The IP address for the PLC is now configured. Now the Ethernet module has to be configured. Right click on Networks in the navigation list to the left and select New Network.

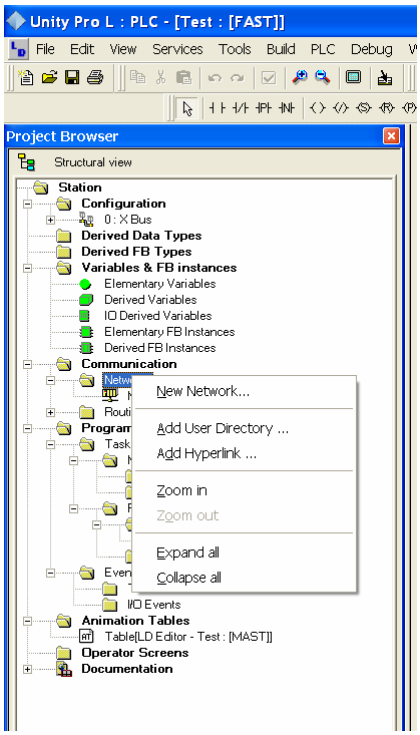


Figure 7 Adding a network.

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

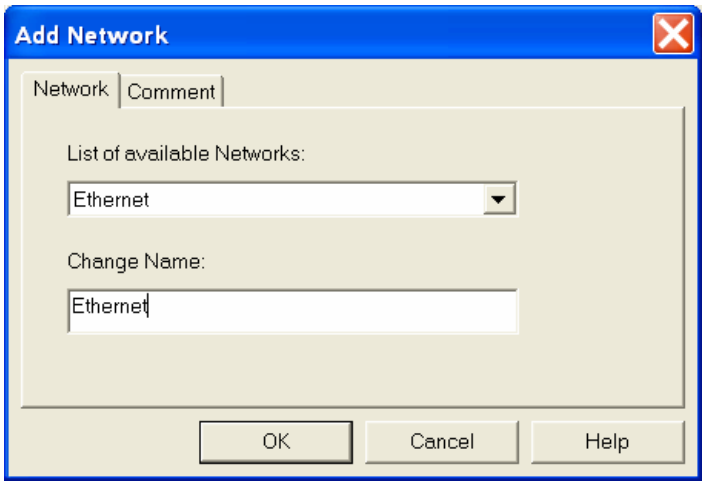


Figure 8 Configuring the network.

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

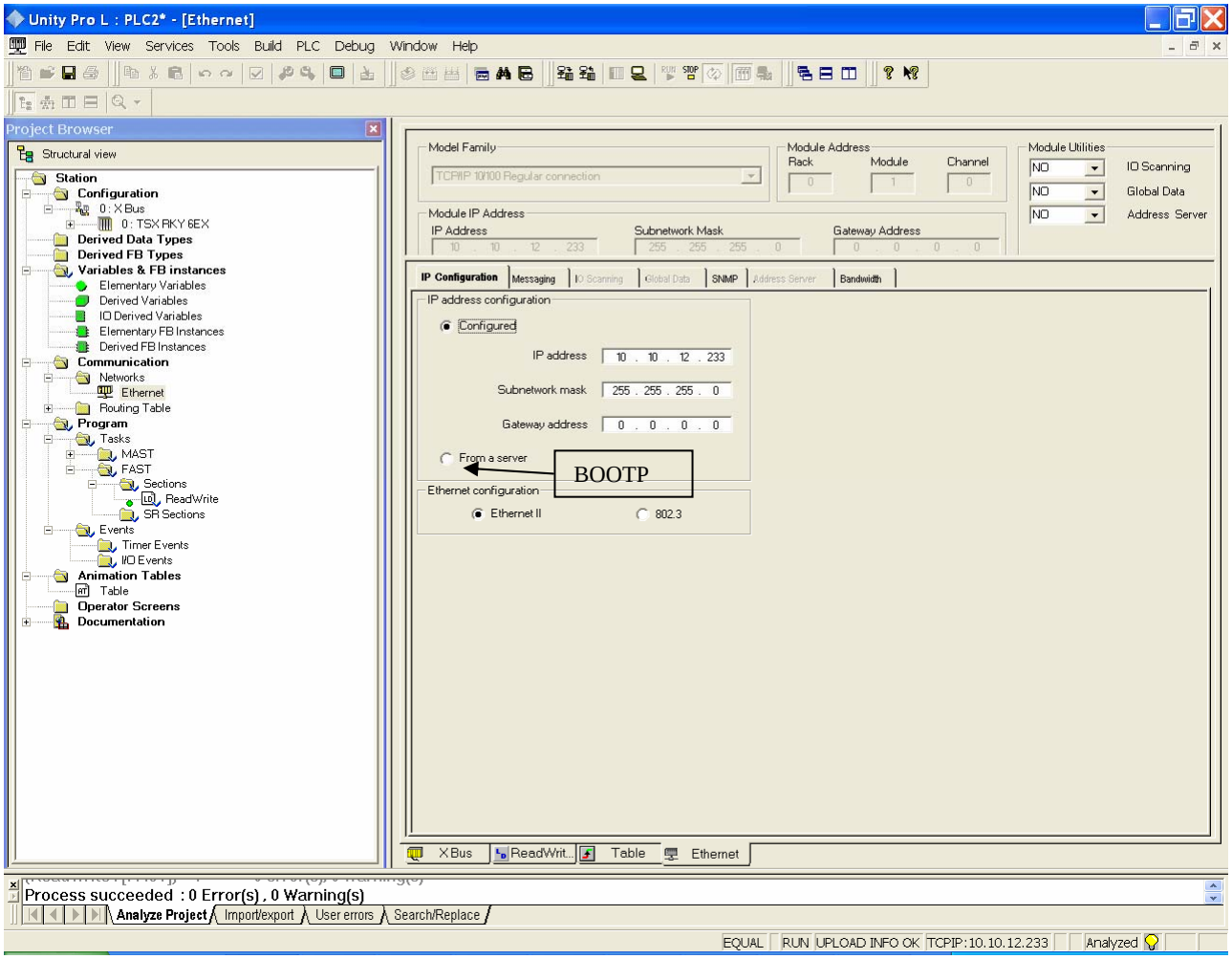


Figure 9 The IP configuration.

The next step is to configure the network the PLC module is to be connected to. Open the configuration in the left navigation list and double click on the Ethernet module as seen below.

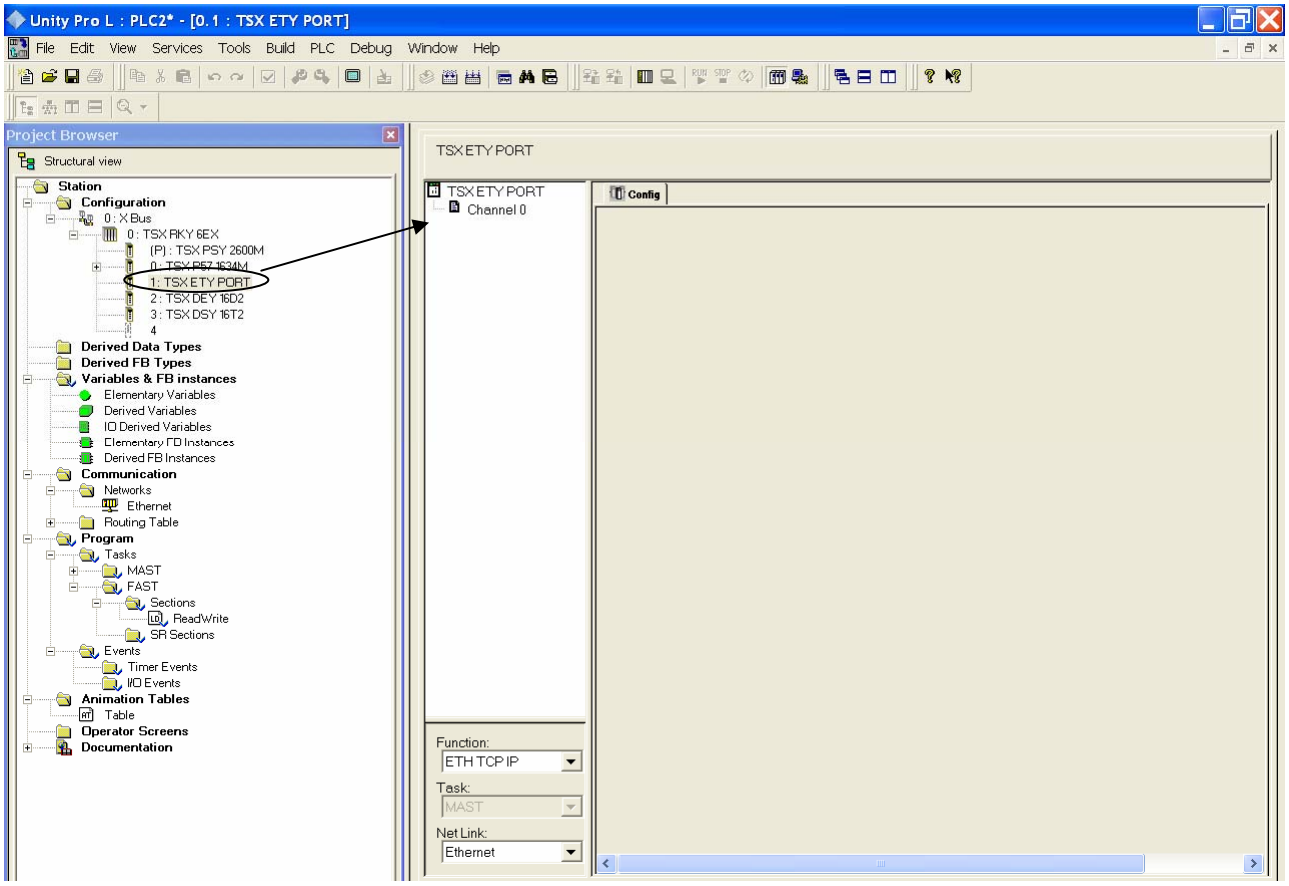


Figure 10 Configuring the PLC Ethernet module.

Select the desired network and close the tile. In this case the network named Ethernet previously configured is selected.

4.2. The Modbus-RTU master configuration

The next step is to configure the Modbus-RTU master in the PLC. Open the X-bus as seen below.

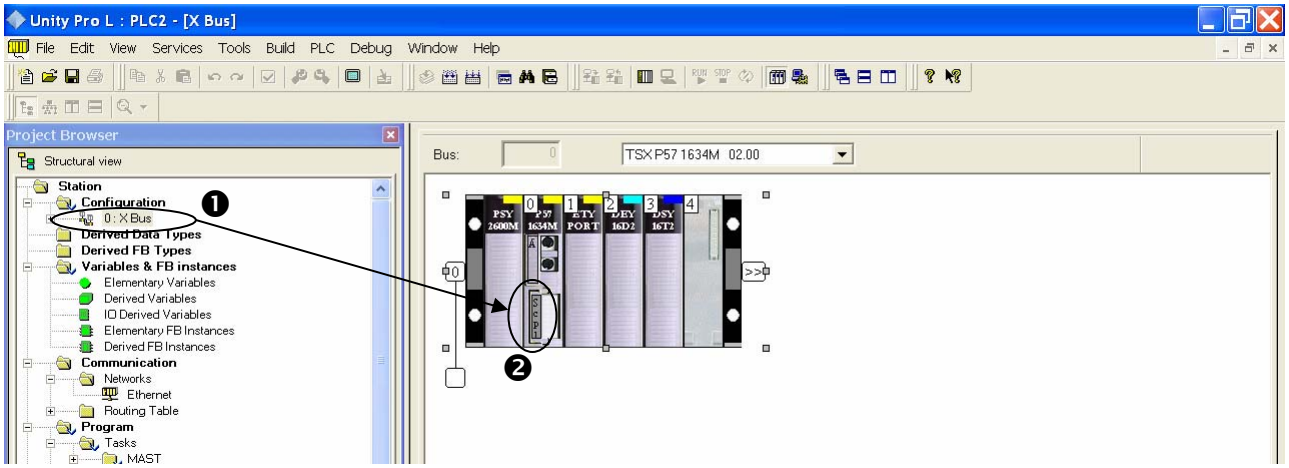


Figure 11 Opening the Modbus-RTU bus.

Double click on the X Bus ❶ to open the PLC configuration. Then double click on the card slot ❷ for the TSX SCP 114 module.

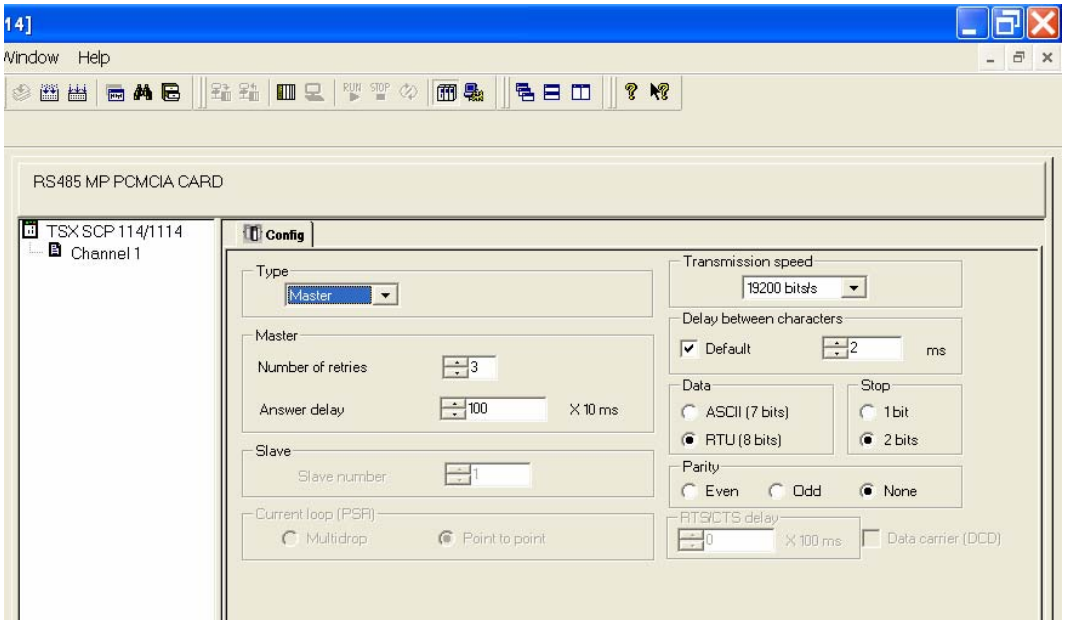


Figure 12 The Modbus-RTU master configuration.

Select the type of module; in this case Master is selected. The parameters are configured as follows; transmission speed 19200 bit/s, RTU 8 bits, 2 stop bits, none parity. The Anybus module has to be configured for the same settings.

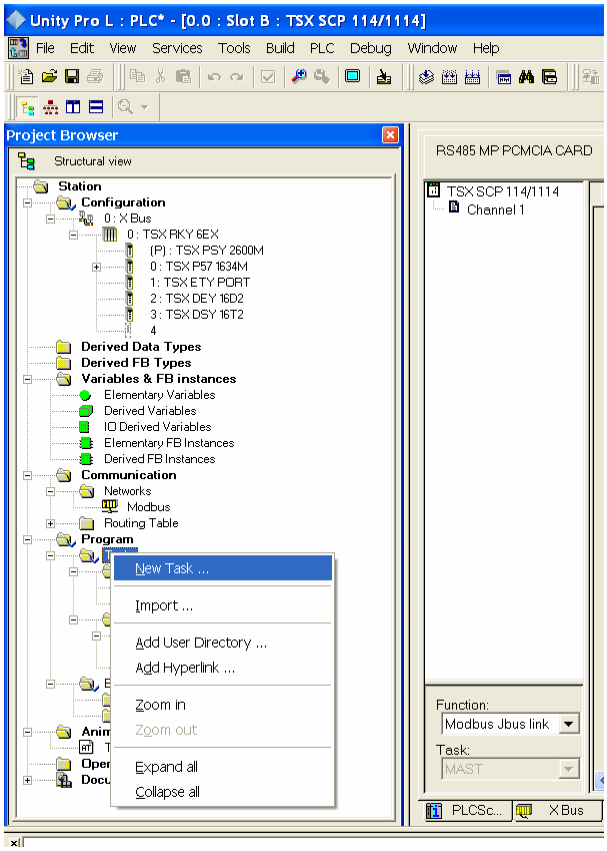
You are advised to use the default value for the parameter “Delay between characters”, the default value is depending on the transmission speed.

4.3. The PLC program

To be able to communicate with the Anybus module the PLC must be set up to exchange data with the Modbus-RTU bus. This can be done by using two function blocks read and write data to and from the Modbus-RTU bus. The following sections describe step by step how the PLC program is configured.

4.3.1. Inserting a new task

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 New Task.



Then right click on sections to add a new section. The section is the same as the actual PLC program.

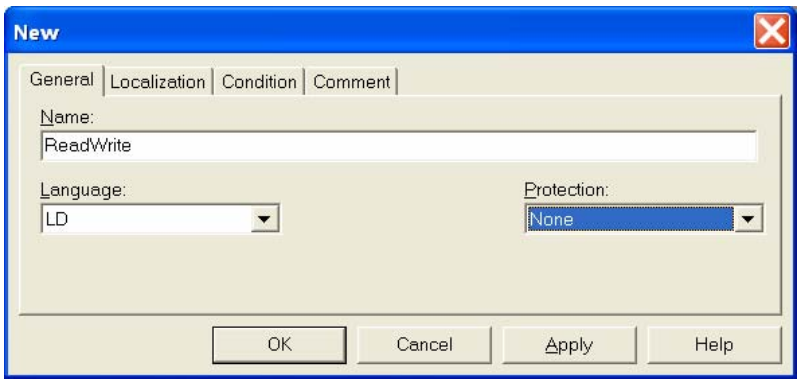


Figure 13 Adding a new program.

The program is built in the ladder language and named ReadWrite in this case. Before the program is set up some settings has to be changed. Open the Tools menu and select the Project Settings. Open the tile Language extension as seen below.

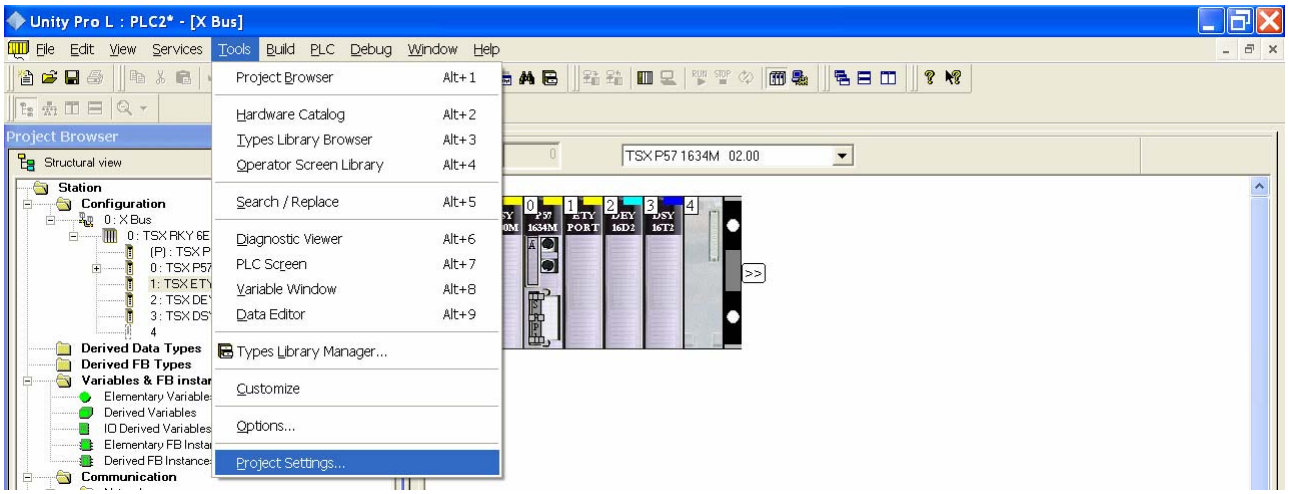


Figure 14 Opening the project settings.

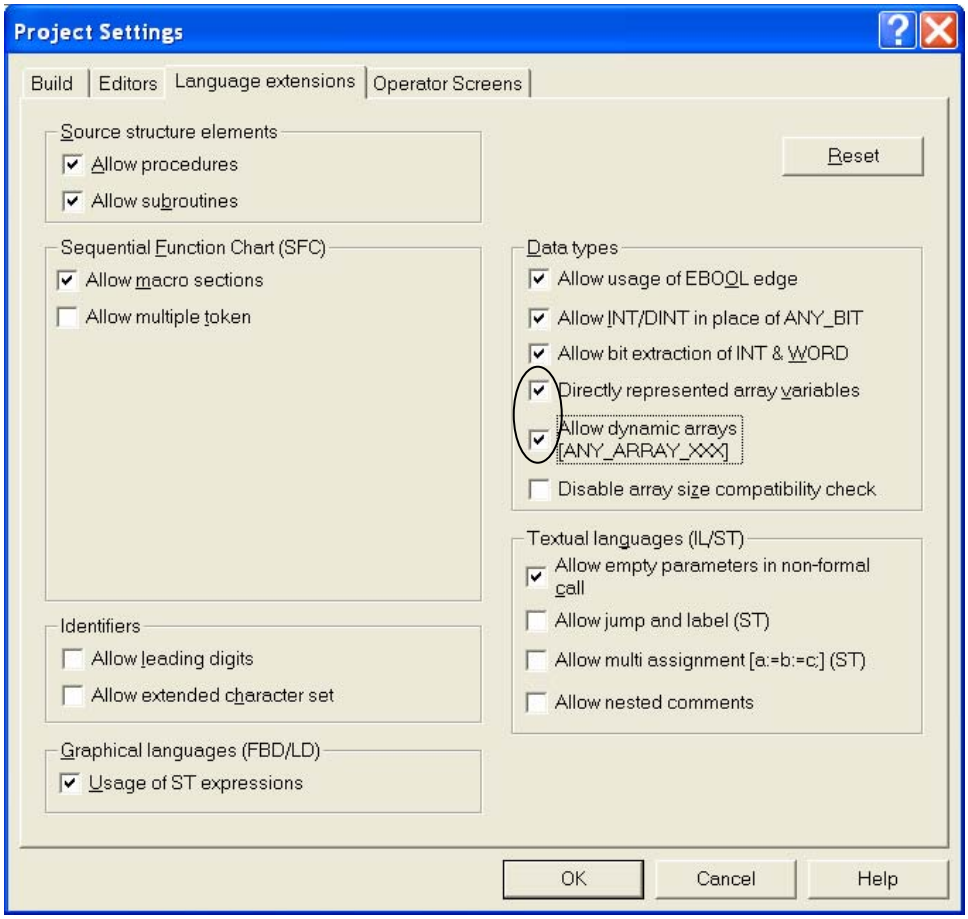


Figure 15 Changing the Project Settings.

Mark the check boxes “Allow dynamic arrays” and “Directly represented array variables”.

4.3.2. The function blocks

The function blocks used in this case is the address, read variable and write variable function. They are named ADDR, READ_VAR and WRITE_VAR respectively. The address function block is used to address the module on the network. The read and write variable function is used to read and write data. The function block uses the two Modbus functions Write multiple registers 16 and Read Holding registers 03. The easiest way to insert the function blocks is to use the Function Input Assistant.

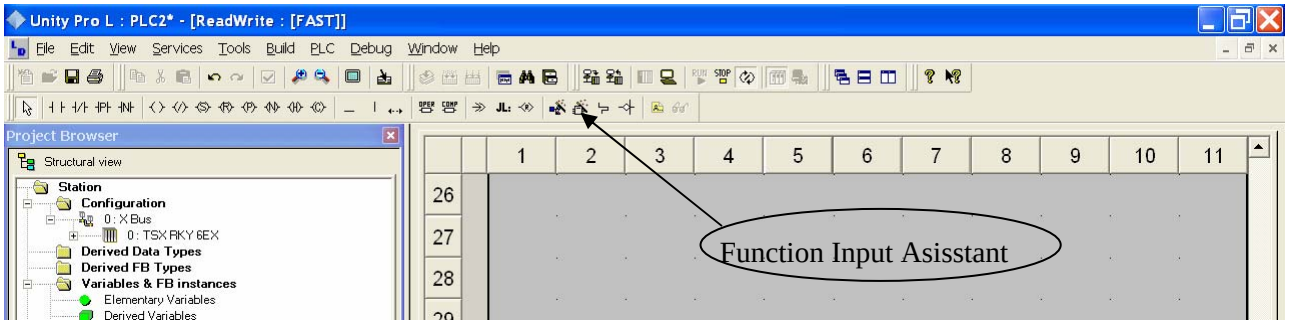


Figure 16 Using the Function Input Assistant.

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

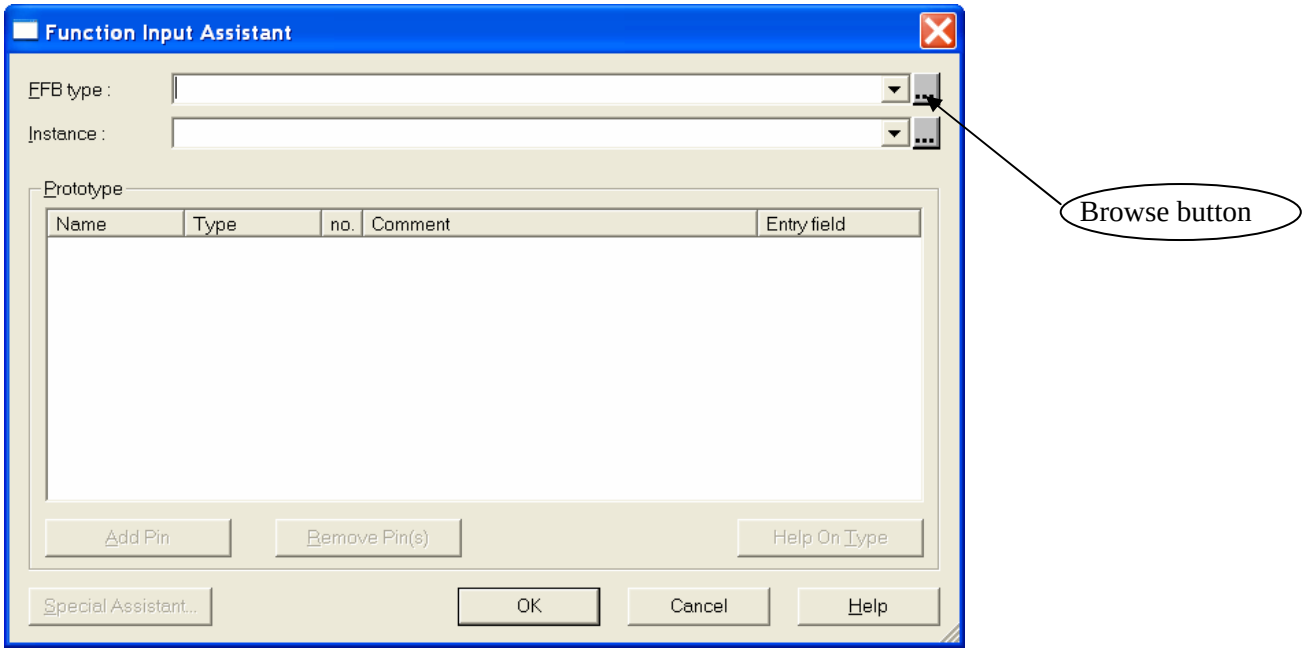


Figure 17 Browsing the Function library.

Double click on the Libset as seen below to expand the library. Select the ADDR function and press OK.

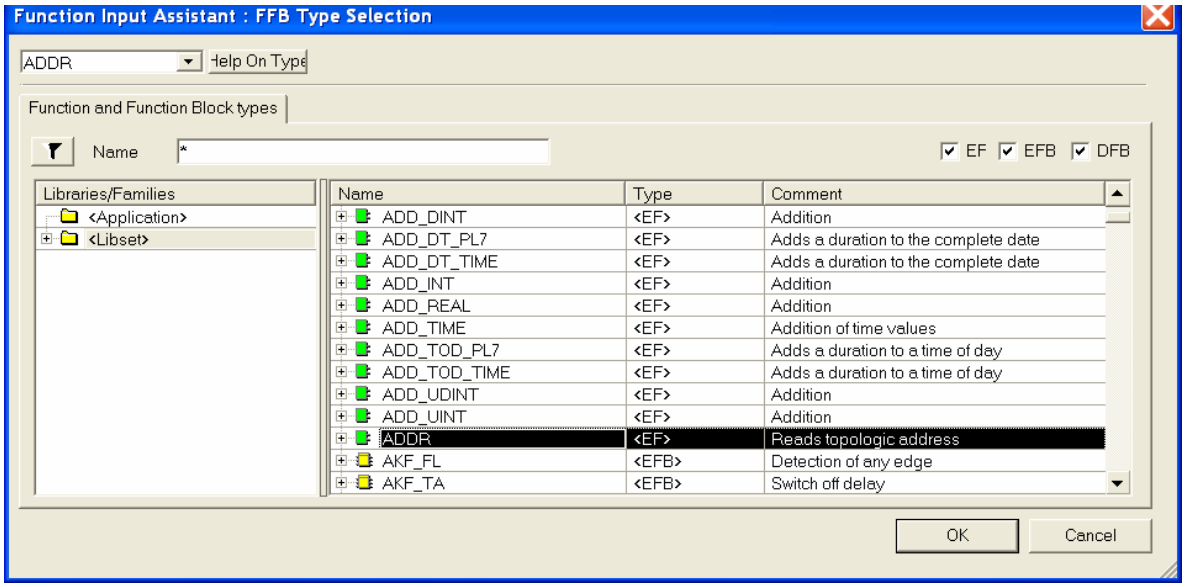


Figure 18 Inserting the ADDR function.

Then click on the working space to add the function block. The figure below shows the section when all the necessary function blocks have been imported. The additional function blocks can be inserted by repeating the previous step. When putting the function blocks next to each other the connection lines are linked. It is important the function blocks are inserted in the right order, otherwise the whole program will not be executed.

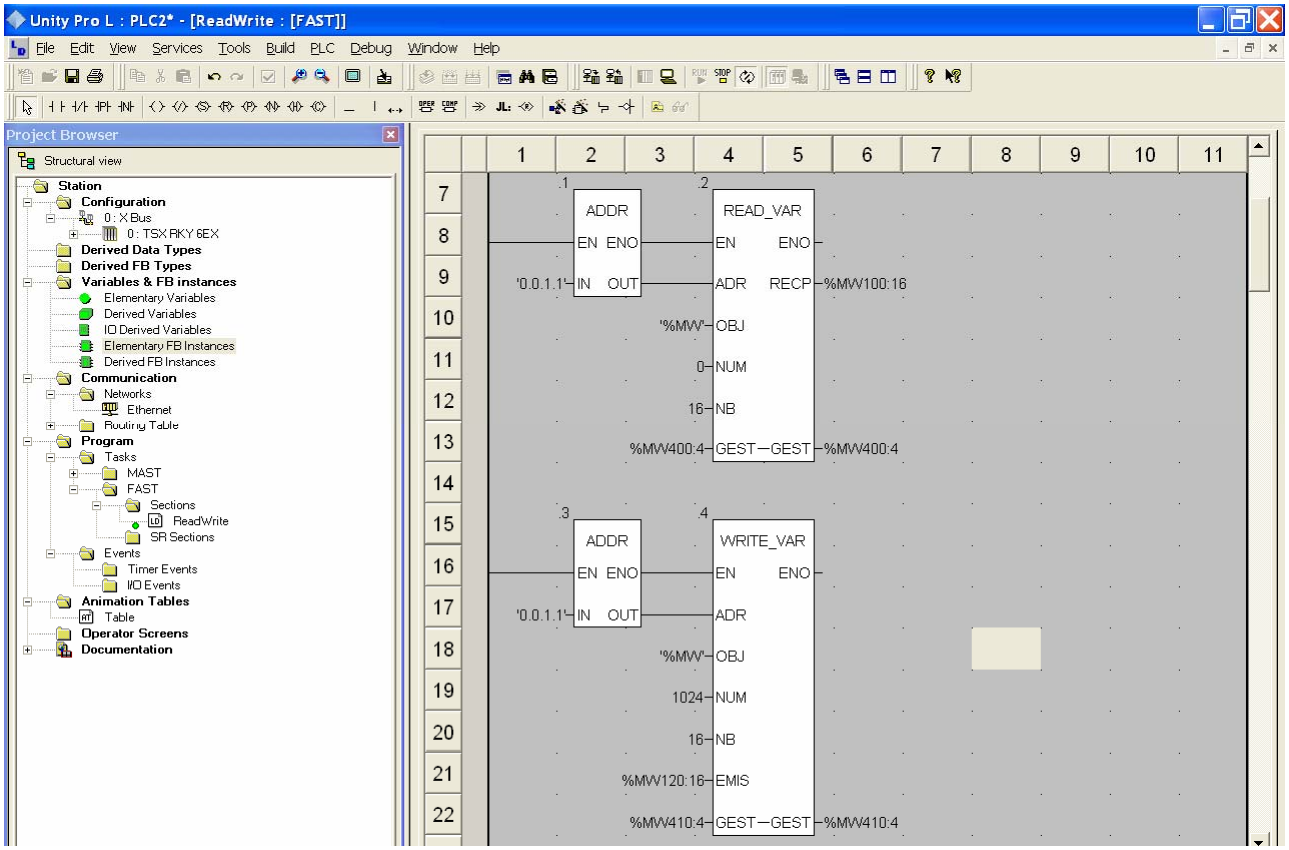


Figure 19 Configuring the function blocks.

After inserting the function blocks the parameters has to be configured. By pressing F1 the help function will be available. The help function contains a detailed useful description of the function blocks. Below follows information collected from the help function in the Unity software concerning the configuration of the parameters.

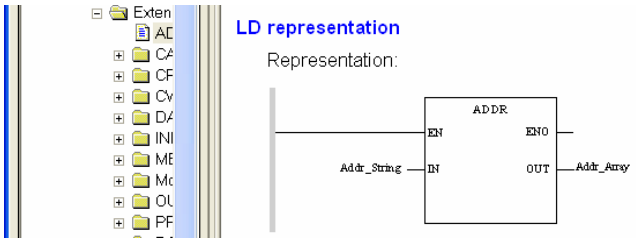


Figure 20 The representation of the address function.

Parameter	Description	Configuration used
Address string	Address to the module	0.0.1.1
Address array	Address sent to the next function	n.a

The address is specified according to rack.module.channel.node address. In this case the rack and the module are numbered to 0. The channel is numbered 1 and the node ID used is 1, so consequently the address is specified to 0.0.1.1. This is the address the PLC is reading and writing data to and from.

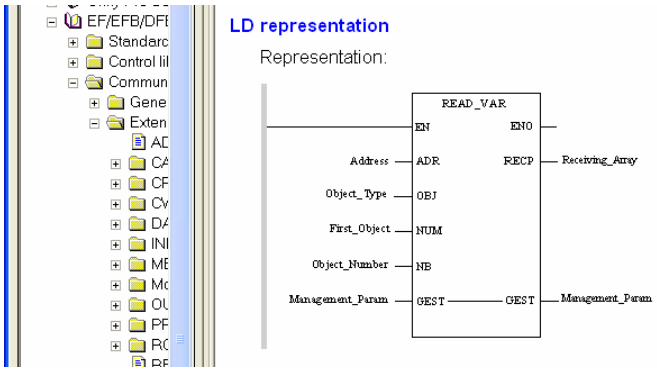


Figure 21 The representation of the read variable function.

For the read function the following has to be configured.

Parameter	Description	Configuration used
Address	Address of the module	Output from the Address function
Object type	Type of object	%MW, word, 2 bytes
First object	Number of first object	0
Object number	Number of object(s)	16
Management parameters	Reports managed by the system	%MW400:4, word 400-404
Receiving array	Location for the received data	%MW100:16, word 100-116

In this case 16 words with start at address 0 is read by the read variable function. The received data is stored in the memory location 100-116 in the memory of the PLC.

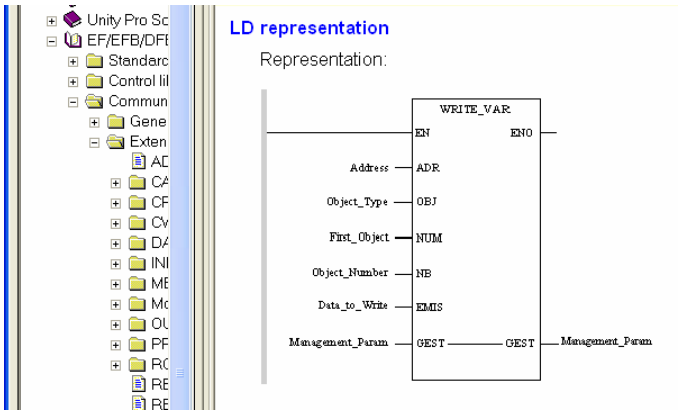


Figure 22 The representation of the write variable function.

For the write function the following has to be configured.

Parameter	Description	Configuration used
Address	Address of the module	Output from the Address function
Object type	Type of object	%MW, word, 2 bytes
First object	Number of the first object	1024
Note: Using the CompactCom the first object (register) to write is 255.		
Object number	Number of object(s)	16
Data to write	Location of the data to write	%MW120:16, word 120-135
Management parameters	Reports managed by the system	%MW400:4, word 400-404

In this case 16 words with start address at 1024 are written by the write variable function. The data to be written is stored in the memory location 120-135 in the memory of the PLC.

4.4. Transferring the project

To be able to run the program the project 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. Make sure the “PLC Run after Transfer” box is checked. Press the transfer button.

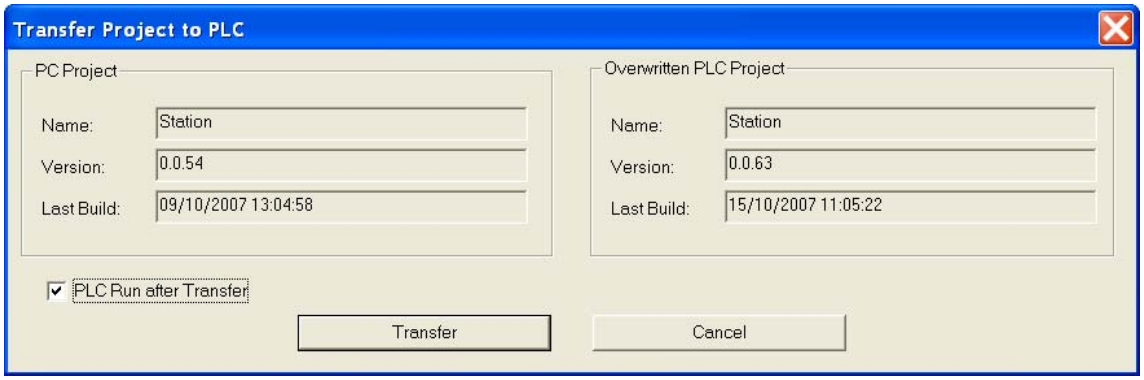


Figure 23 Downloading the program.

5. Anybus configuration

The Anybus product has to be configured for the same I/O sizes and parameter settings as set up in the Modbus master configuration. The parameters configured are the node ID, baudrate, parity and physical interface type. The configuration procedure is depending on the type of module. See the sections below.

5.1. Configuration by switches

The Anybus Communicator, Anybus-S Slave and Anybus X-gateway can be configured by switches. The switches are numbered according to the figures below. The first switch enables the termination resistor.

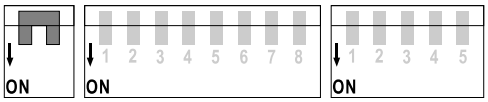


Figure 24 Setting the termination switch.

If the Anybus module is the last node in the network the termination resistor is to be set in the ON position. In this case the switch is set to the ON position.

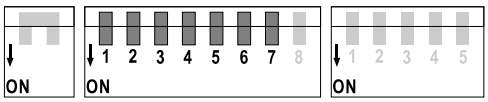


Figure 25 Configuring the Node ID switches.

The Node ID is configured to 1 as in the Modbus master configuration. The switches are set in the following positions, 0000001.

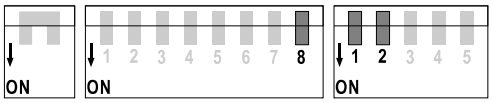


Figure 26 Configuring the baudrate switches.

Then the baudrate is configured to 19200kbit/s as in the Modbus master configuration. The switches are set in the following positions, 101.

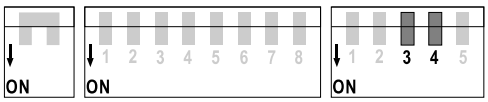


Figure 27 Configuring the parity switches.

Next the parity switches are configured for no parity and 2 stop bits as configured in the master configuration. The switches are set in the following positions, 01.

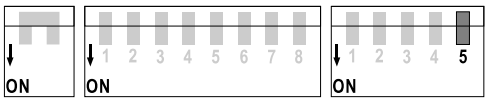


Figure 28 Configuring the physical interface switch.

Finally the switch for the physical interface is configured for RS485. The RS485 is the interface supported by the Modbus Master module. The switch is set in the OFF position. For further information consult the Anybus-S Fieldbus Appendix. Now the parameters are configured, the next step is to configure the I/O sizes.

5.2. CompactCom configuration

The module is configured by messages send by the application. The maximum data exchange is 512 bytes in each direction. The input data is mapped to Holding register 0-255 and the output data to Holding register 255-512. Refer to the Network Interface Appendix for details.

5.3. Slave Interface configuration

The Modbus parameters of the Anybus Slave Interface are configured by switches or by mailbox commands. The maximum Node ID to be used by the switches is 127, using mailbox command the maximum address is 247. Refer to the Anybus-S Fieldbus Appendix for details of the mailbox commands. For configuration by switches see the description in chapter 5.1, Configuration by switches.

The size of the data exchange used by Anybus Slave Interface is configured by mailbox commands. The maximum data exchange is 2048 bytes in each direction. The input data is mapped to Holding register 0-1023 and the output data to Holding register 1024-2047. Refer to the Anybus-S Fieldbus Appendix for details.

5.4. Communicator configuration

Modbus parameters

The Modbus parameters of the Anybus Communicator are configured by switches or by mailbox commands for the advanced user. The maximum Node ID to be used by the switches is 127, using mailbox the maximum address is 247. Refer to the Anybus-S Fieldbus Appendix and the Anybus Communicator User manual for details of the mailbox commands. For configuration by switches please see the description in chapter 5.1, Configuration by switches.

Communicator configuration

To configure the Communicator start the ABC Config Tool and start a new project. Select the fieldbus Modbus-RTU.

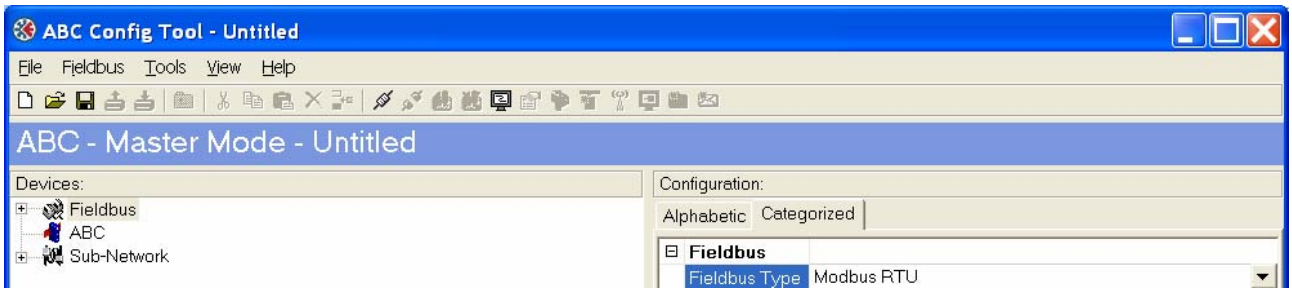


Figure 29 Configuring the Field bus.

Note: *The amount of data configured for the fieldbus is depending on the sub-network configuration. The fieldbus I/O data will be of the same size as configured for the sub-network. The maximum data exchange is 512 bytes in each direction. The input data is mapped to Holding register 0-255 and the output data to Holding register 1024-1279.*

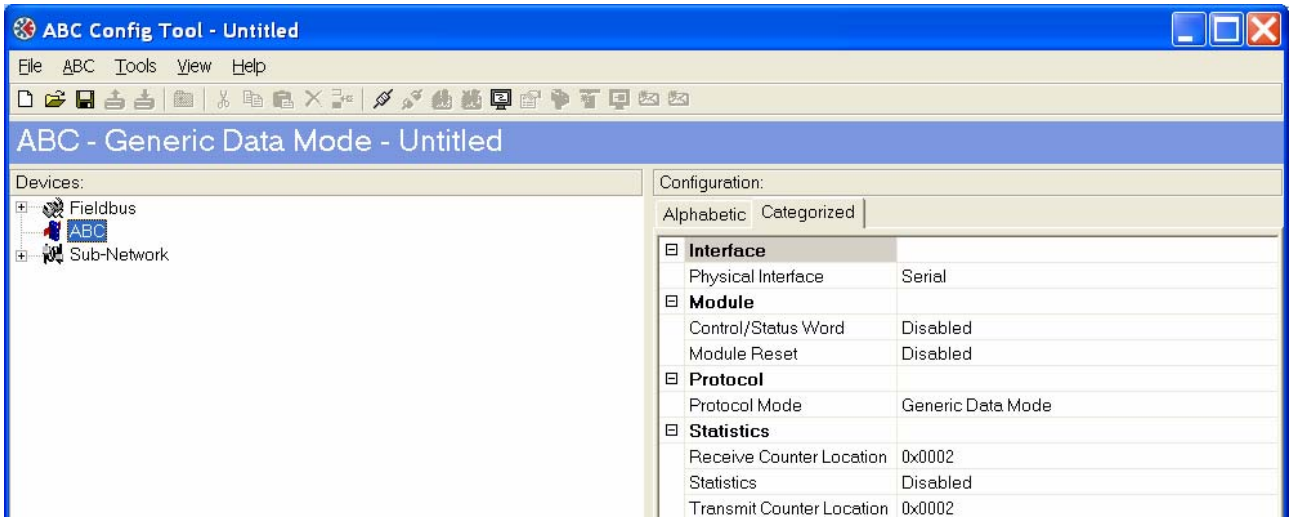


Figure 30 Configuring the Communicator.

The next step is to configure the sub-network.

The configuration is depending on the application, in other words what nodes are connected. In this case a loop back dongle at the serial connection of the Communicator is connected. Therefore a test configuration is used to loop data. For this purpose the generic data mode is selected; all other values are left at their defaults.

***Note:** The configuration seen below is the sub-network configuration, the Modbus-RTU configuration is done by switches.*

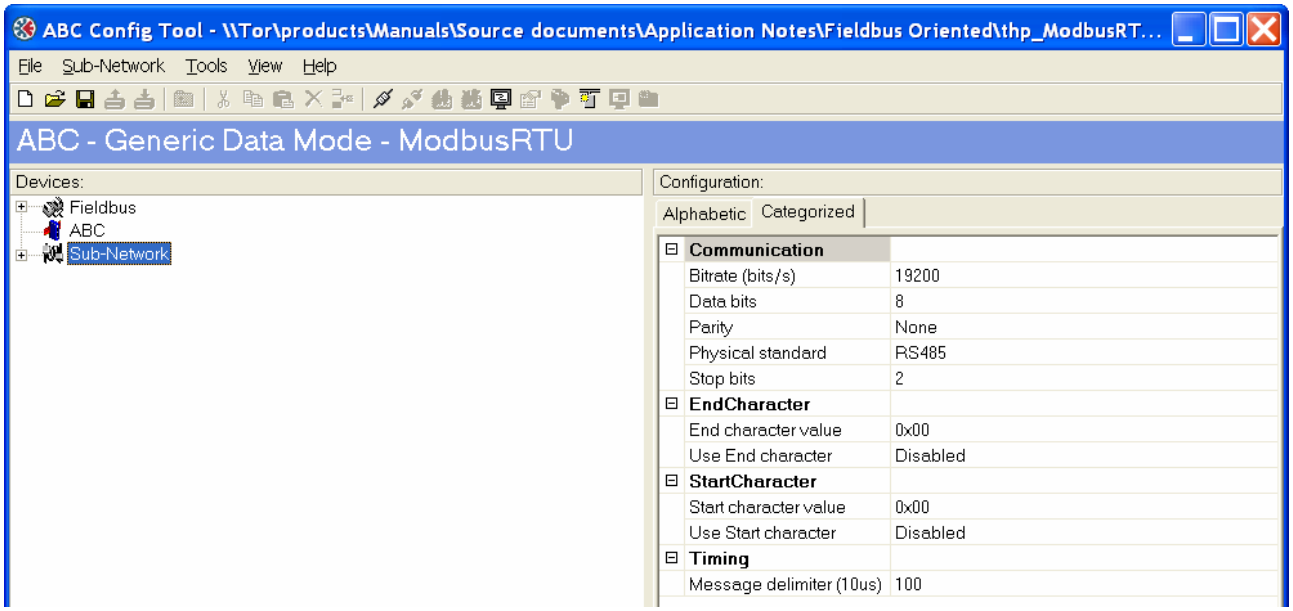
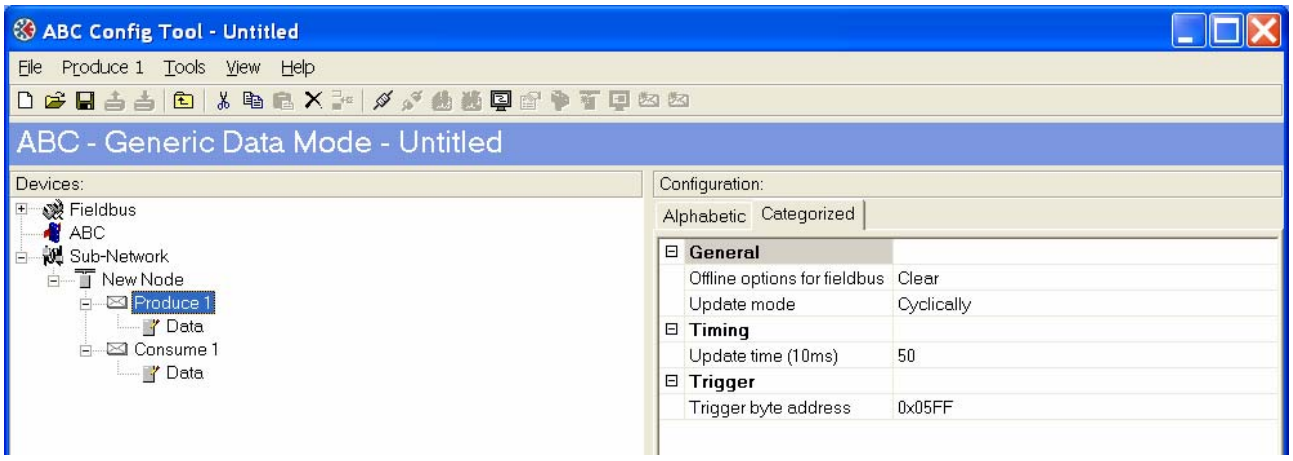


Figure 31 Configuring the sub network.

Right click on new node and add a consume and a produce transaction as shown below



Change the Update time to 500ms and leave the settings for the Consume transaction at the defaults. Right click on the produce and consume transaction respectively and select add data. In this case 32 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 500ms and the offline timeout time to 1000ms.*

For a more detailed description see the Communicator User Manual.

5.5. X-gateway configuration

Modbus parameters

The Modbus parameters of the Anybus X-gateway are configured by switches. The maximum Node ID to be used is 127. For configuration by switches please see the description in chapter 5.1, Configuration by switches.

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.

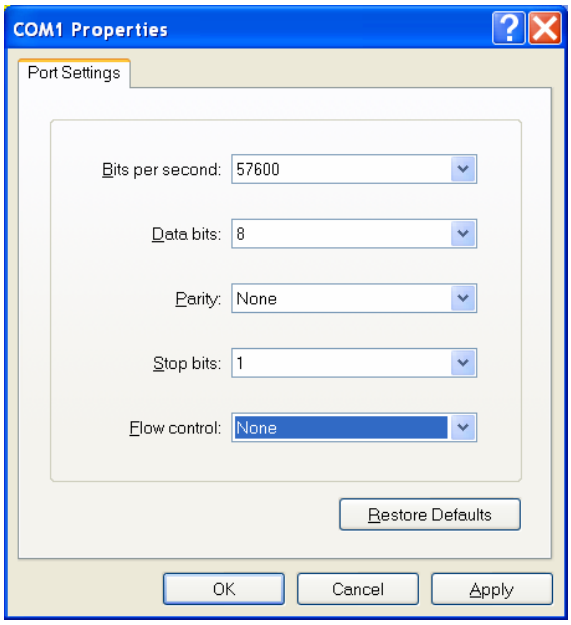


Figure 32 Configuring the connection in the HyperTerminal.

Make sure the settings are identical to those shown in the window above. Alternatively download a HyperTerminal session file from the HMS website[1], double click on it and select COM port.

Connect and press ESC and the following menu will appear.

[1] www.anybus.com

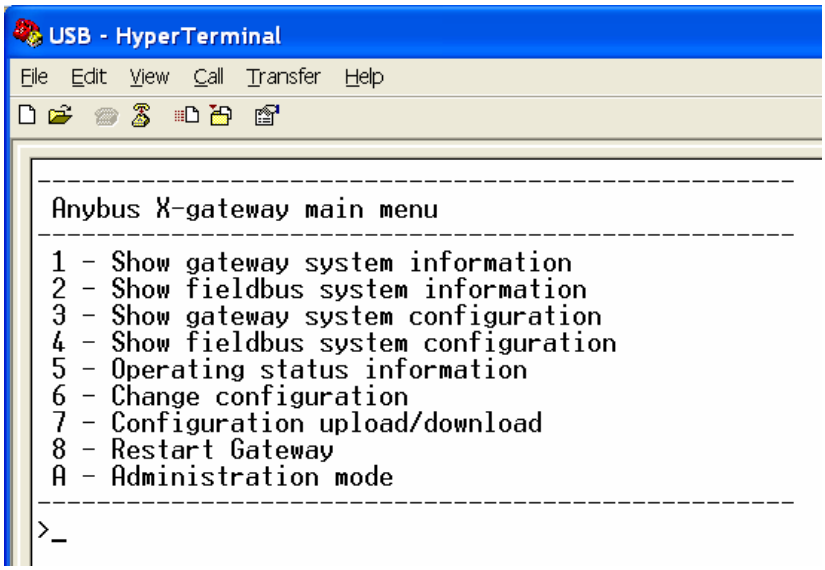


Figure 33 Anybus X-gateway Main menu.

Press 6 and enter the desired configuration.

The figure below shows an example; in this case a PROFIBUS-DPV1/Modbus-RTU X-gateway is used and 32 bytes of I/O data is configured.

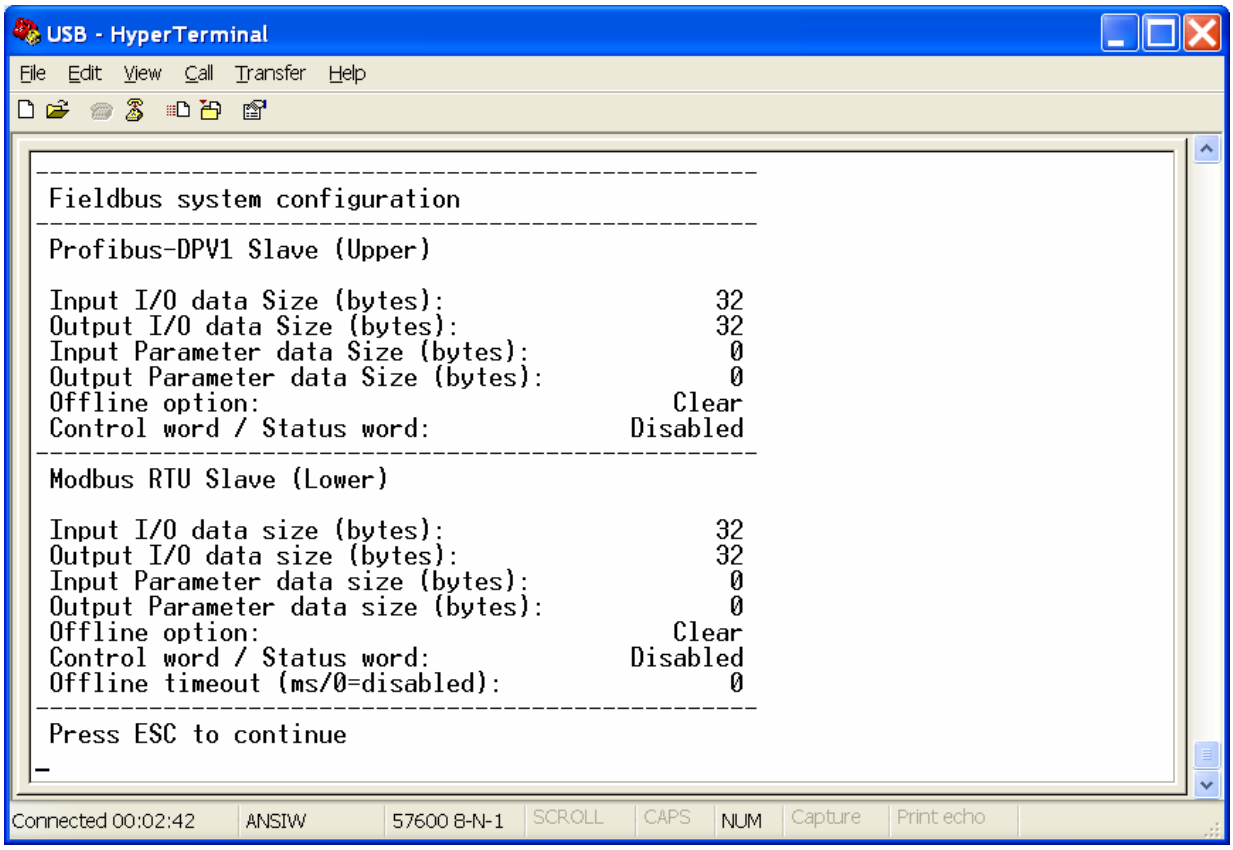


Figure 34 The X-gateway configuration.

The maximum data exchange is 512 bytes in each direction. The input data is mapped to Holding register 0-255 and the output data to Holding register 1024-1279. For a more detailed description see the X-gateway Manual Addendum.

6. Testing

After finishing the configuration the testing of the network can be done. In this case the Anybus Communicator with a loop dongle at the sub-network connector is used for test purpose.

Start the Unity Pro L software again. To be able to monitor the Input and Output an animation table has to be inserted. 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 addresses to monitor. The memory destination of the I/O data can be seen in Figure 19. In this case the I/O data to be written is stored in words 120 to 135 and the data to be read in position 100 to 115.

Note: The addresses entered in the animation table depend on your PLC configuration. The animation table below is just an example.

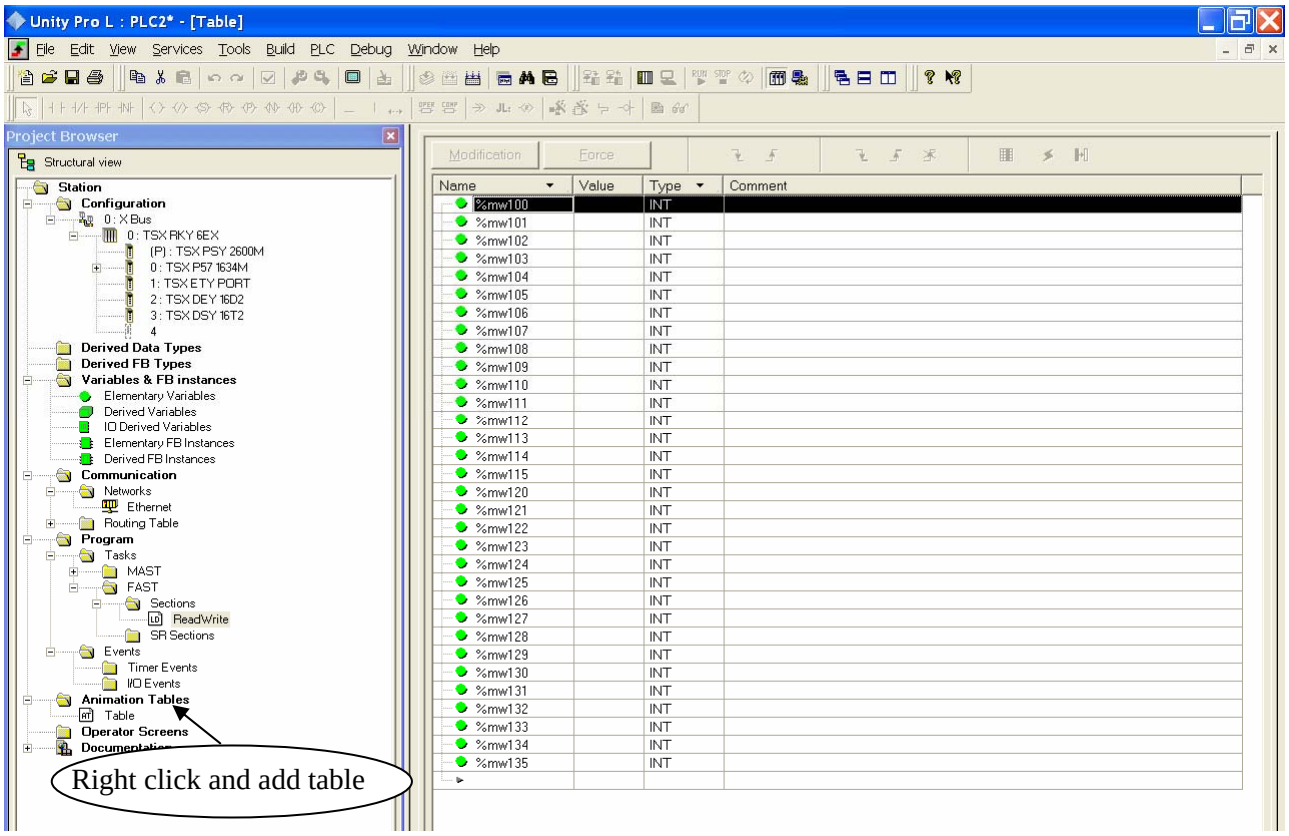


Figure 35 The animation table.

The next step is to go online and modify the memory positions in the PLC to test the communication. Press the connect button❶ to go online. Then press the run button❷ in the top navigation list and press the modification button❸ to modify the variables.

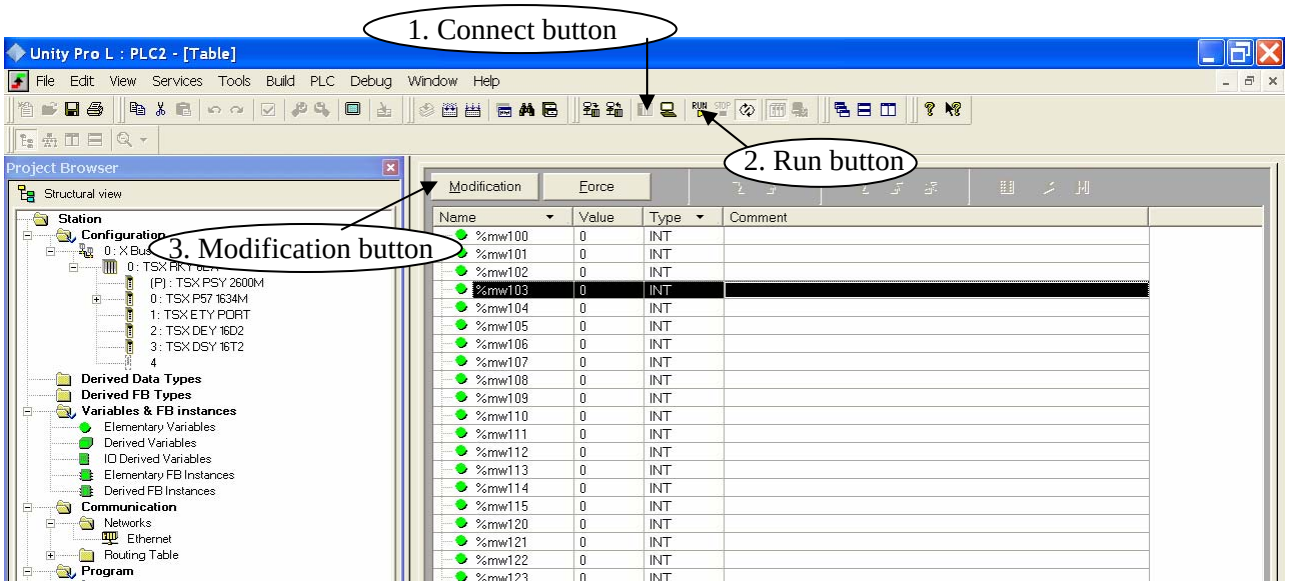


Figure 36 Modifying the animation tables.

In this case the values 1, 3, 4, 5 are entered as seen below❶. If the network is set up correctly the corresponding input will be seen❷.

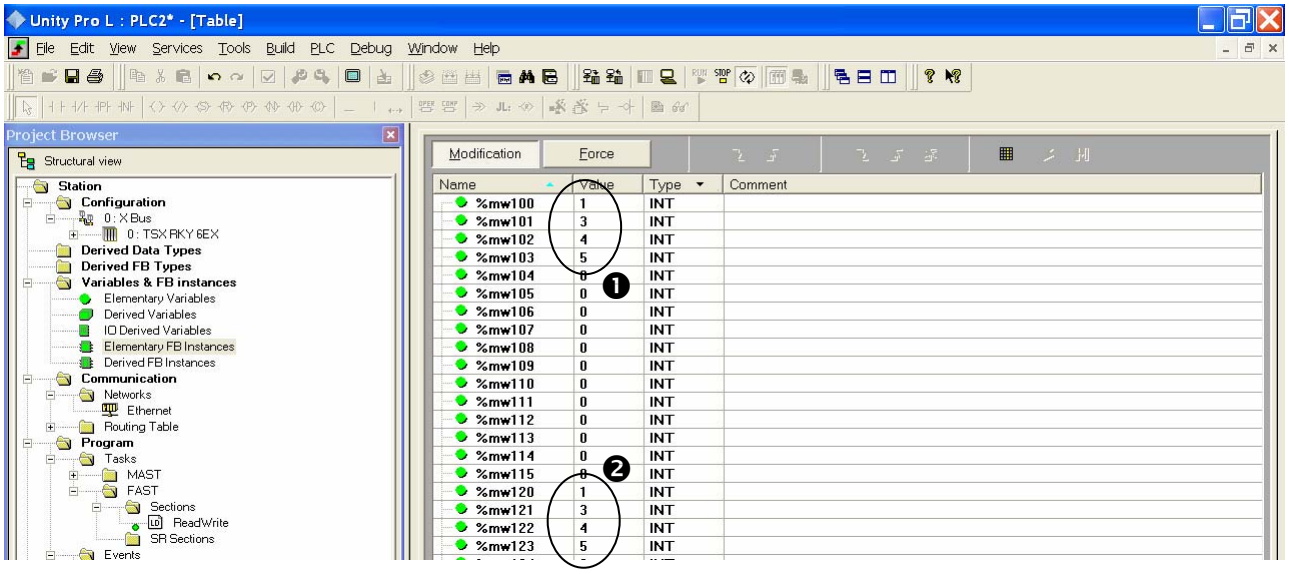


Figure 37 Scanning the Input register.