Setup of Modbus TCP in CODESYS

Abstract:
This application note helps you creating your first Modbus TCP connection on a u-OS device with CODESYS. It describes the configuration of both the Modbus TCP Server and the Modbus TCP client in CODESYS.

Hardware reference

No.	Component name	Article No.	Hardware / Firmware version
1	UC20-WL2000-AC	1334950000	u-OS 2.0.0
2	IoT-GW30	2682620000 2682630000	u-OS 2.0.0

Software reference

No.	Software name	Article No.	Software version
1	CODESYS Development System		SP18 Patch 4
2	CODESYS Runtime App		4.7.0.0-2
3	CODESYS Control SL for Weidmueller u-OS		4.7.0.0

Contact
Weidmüller Interface GmbH & Co. KG
Klingenbergstraße 26
32758 Detmold, Germany
www.weidmueller.com

For any further support please contact your local sales representative:
https://www.weidmueller.com/countries

Content

1 Warning and Disclaimer.......... 4

2 Introduction.......... 5

3 Modbus TCP Master (Client) Configuration 6

3.1 Function codes 10

4 Modbus TCP Slave (Server) Configuration 12

5 Function test 16

1 Warning and Disclaimer

Warning
Controls may fail in unsafe operating conditions, causing uncontrolled operation of the controlled devices. Such hazardous events can result in death and / or serious injury and / or property damage. Therefore, there must be safety equipment provided / electrical safety design or other redundant safety features that are independent from the automation system.

Disclaimer
This Application Note / Quick Start Guide / Example Program does not relieve you of the obligation to handle it safely during use, installation, operation and maintenance. Each user is responsible for the correct operation of his control system. By using this Application Note / Quick Start Guide / Example Program prepared by Weidmüller, you accept that Weidmüller cannot be held liable for any damage to property and / or personal injury that may occur because of the use.

Note
The given descriptions and examples do not represent any customer-specific solutions, they are simply intended to help for typical tasks. The user is responsible for the proper operation of the described products. Application notes / Quick Start Guides / Example Programs are not binding and do not claim to be complete in terms of configuration as well as any contingencies. By using this Application Note / Quick Start Guide / Example Program, you acknowledge that we cannot be held liable for any damages beyond the described liability regime. We reserve the right to make changes to this application note / quick start guide / example at any time without notice. In case of discrepancies between the proposals Application Notes / Quick Start Guides / Program Examples and other Weidmüller publications, like manuals, such contents have always more priority to the examples. We assume no liability for the information contained in this document. Our liability, for whatever legal reason, for damages caused using the examples, instructions, programs, project planning and performance data, etc. described in this Application Note / Quick Start Guide / Example is excluded.

Security notes
In order to protect equipment, systems, machines and networks against cyber threats, it is necessary to implement (and maintain) a complete state-of-the-art industrial security concept. The customer is responsible for preventing unauthorized access to his equipment, systems, machines and networks. Systems, machines and components should only be connected to the corporate network or the Internet if necessary and appropriate safeguards (such as firewalls and network segmentation) have been taken.

2 Introduction

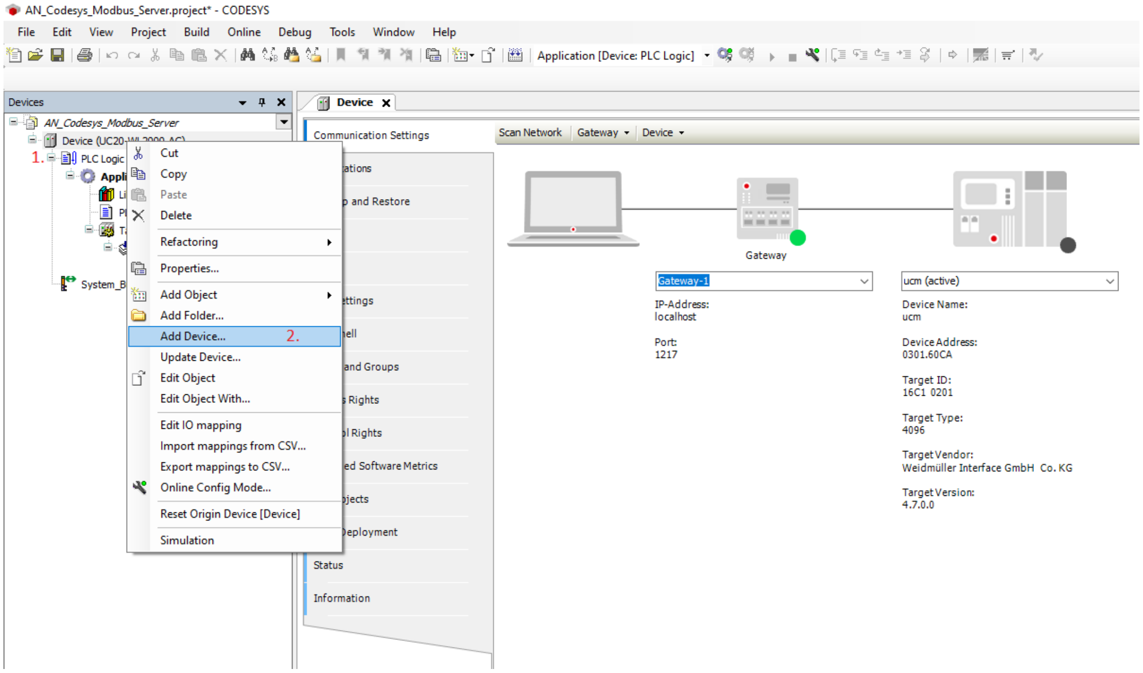
In most automation systems, sooner or later you are faced with the task of implementing a fieldbus connection. Among other things, this may be necessary if external sensors are required, or various I/O modules are to be integrated. In addition, a fieldbus system can be used to exchange data between different control devices.

This application note explains how to setup a Modbus TCP communication with CODESYS. It depends on your application if your controller needs to act as Modbus client or Modbus server. We will describe both variants by providing an example for a Modbus TCP communication between two CODESYS controllers (Client and Server).

3 Modbus TCP Master (Client) Configuration

Open a new CODESYS standard project for the Master application.

1. First add an "Ethernet node" to the CODESYS device in the device tree ⇒ Device (right click) ⇒ Add Device...



Gateway-1

IP-Address:
localhost

Port:
1217

ucm (active)

Device Name:
ucm

Device Address:
0301.60CA

Target ID:
16C1 0201

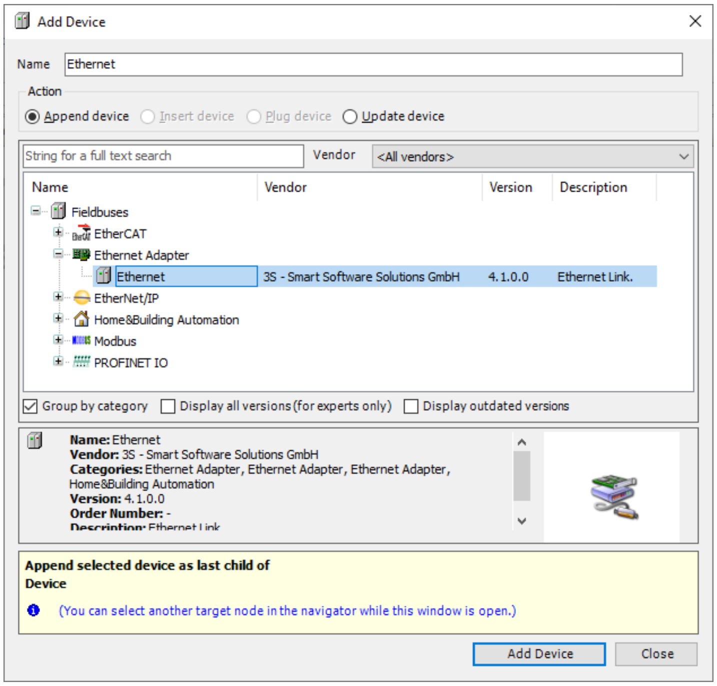
Target Type:
4096

Target Vendor:
Weidmüller Interface GmbH Co. KG

Target Version:
4.7.0.0

Figure 1: Add Device

2. Ethernet Adapter ⇒ Ethernet ⇒ Add Device



Name: Ethernet

● Append device ○ Insert device ○ Plug device ○ Update device

String for a full text search Vendor <All vendors>

Name	Vendor	Version	Description
Fieldbuses			
EtherCAT			
Ethernet Adapter			
Ethernet	3S - Smart Software Solutions GmbH	4.1.0.0	Ethernet Link.
EtherNet/IP			
Home&Building Automation			
Modbus			
PROFINET IO			

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

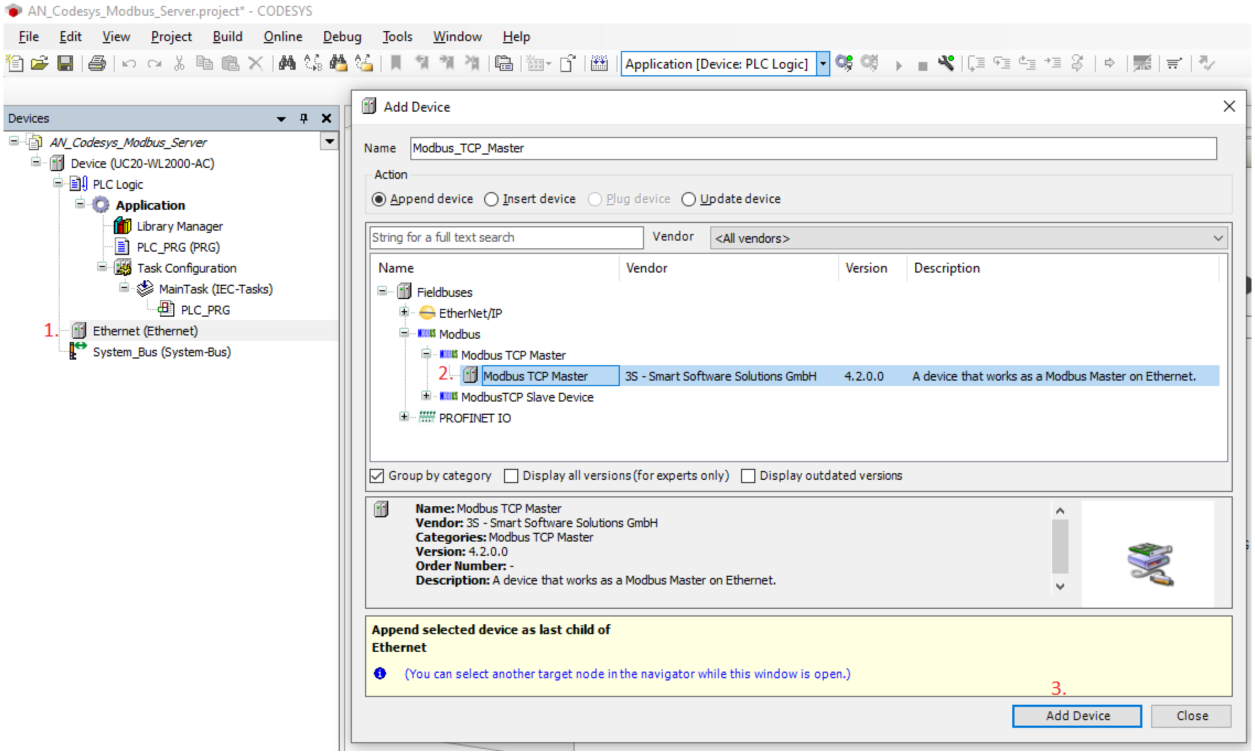
Name: Ethernet
Vendor: 3S - Smart Software Solutions GmbH
Categories: Ethernet Adapter, Ethernet Adapter, Ethernet Adapter, Home&Building Automation
Version: 4.1.0.0
Order Number: -
Description: Ethernet Link

Append selected device as last child of Device

(You can select another target node in the navigator while this window is open.)

Figure 2: Ethernet device

3. Then attach the Modbus TCP Master to the previously inserted Ethernet element. In the device tree ⇒ Ethernet (right click) ⇒ Add device... ⇒ Modbus TCP Master



Name: Modbus_TCP_Master

Action: ● Append device ○ Insert device ○ Plug device ○ Update device

String for a full text search Vendor: <All vendors>

Name	Vendor	Version	Description
Modbus TCP Master	3S - Smart Software Solutions GmbH	4.2.0.0	A device that works as a Modbus Master on Ethernet.

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

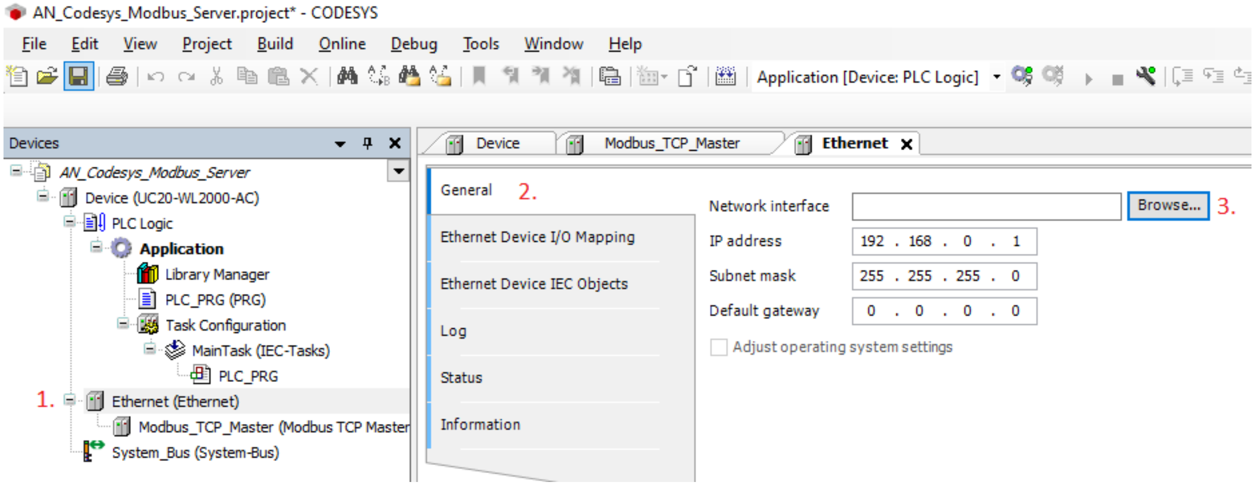
Name: Modbus TCP Master
Vendor: 3S - Smart Software Solutions GmbH
Categories: Modbus TCP Master
Version: 4.2.0.0
Order Number: -
Description: A device that works as a Modbus Master on Ethernet.

Append selected device as last child of Ethernet

(You can select another target node in the navigator while this window is open.)

Figure 3: Add Modbus Master

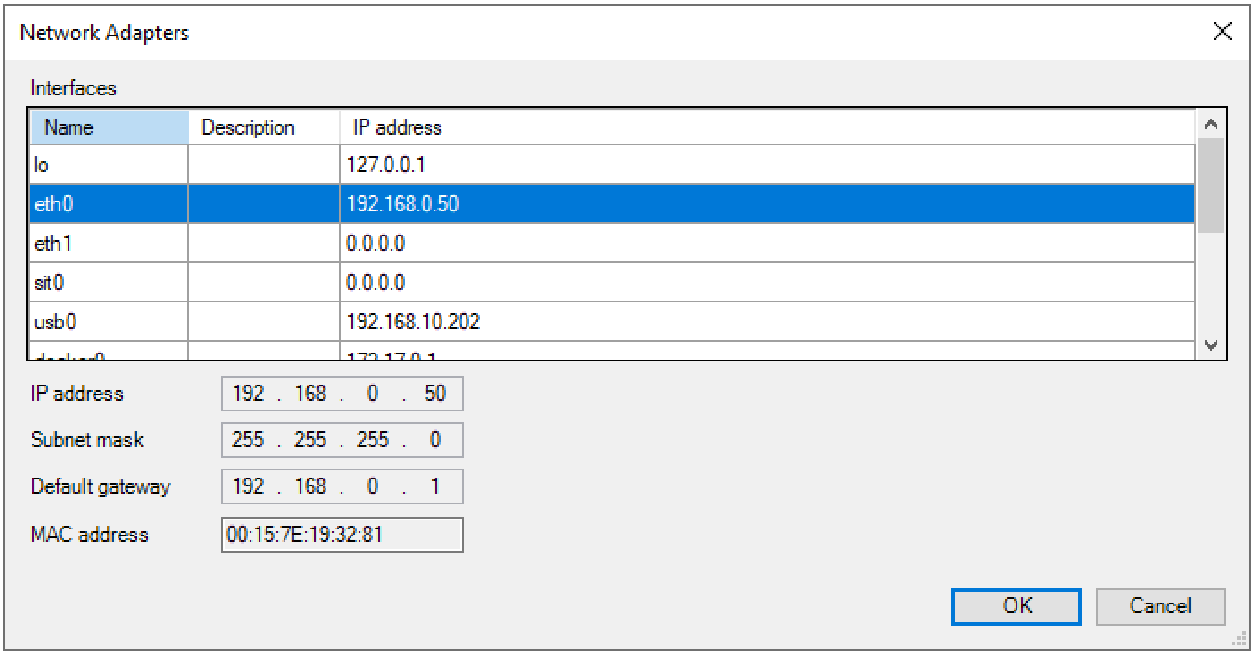
4. The Modbus TCP Master is now available below the Ethernet node in the device tree. First the Ethernet must be parameterized. To do this, open the configurator (by double-clicking on Ethernet) and click on the button Browse… behind the Network interface.



Network interface:
IP address: 192 . 168 . 0 . 1
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 0 . 0 . 0 . 0
☐ Adjust operating system settings

Figure 4: Network interface

5. In the following dialogue, select "eth0" or "eth1“ or another suitable adapter for your controller, depending on which connection is to be used.



Network Adapters

Interfaces

Name	Description	IP address
lo		127.0.0.1
eth0		192.168.0.50
eth1		0.0.0.0
sit0		0.0.0.0
usb0		192.168.10.202
docker0		172.17.0.1

IP address: 192 . 168 . 0 . 50

Subnet mask: 255 . 255 . 255 . 0

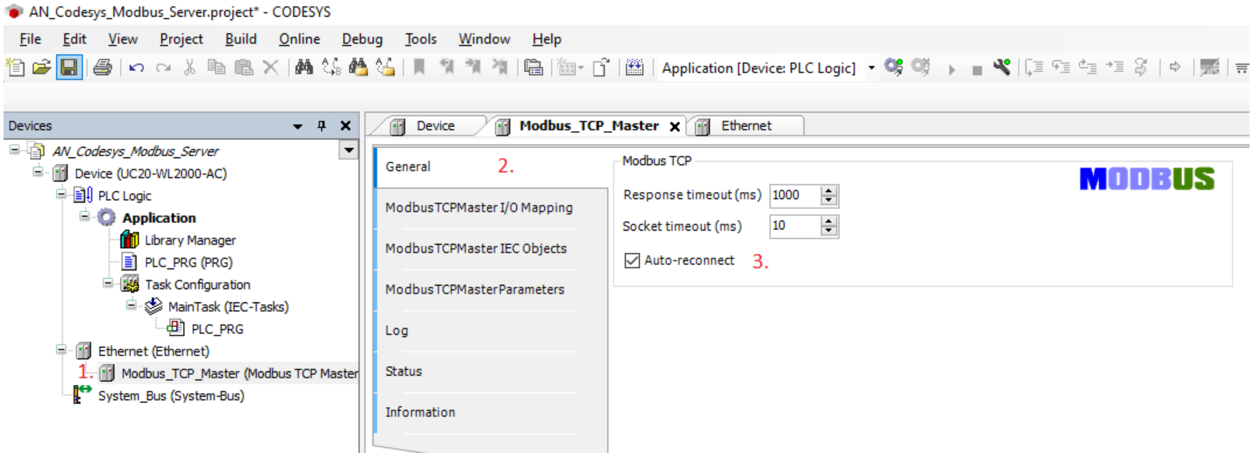
Default gateway: 192 . 168 . 0 . 1

MAC address: 00:15:7E:19:32:81

Figure 5: eth0

Note:
To configure the Network interface, you must be connected to the controller.

6. You also need to set up the Modbus TCP master: Open the Modbus Master editor and adjust the settings. It is recommended to activate the option for auto-reconnect in the "General" tab. All the settings for the master are completed.



AN_Codesys_Modbus_Server.project* - CODESYS

- AN_Codesys_Modbus_Server
  - Device (UC20-WL2000-AC)
    - PLC Logic
      - Application
        - Library Manager
        - PLC_PRG (PRG)
        - Task Configuration
          - MainTask (IEC-Tasks)
            - PLC_PRG
    - Ethernet (Ethernet)
      - Modbus_TCP_Master (Modbus TCP Master
    - System_Bus (System-Bus)

General
ModbusTCPMaster I/O Mapping
ModbusTCPMaster IEC Objects
ModbusTCPMasterParameters
Log
Status
Information
1.
2.
3.
MODBUS

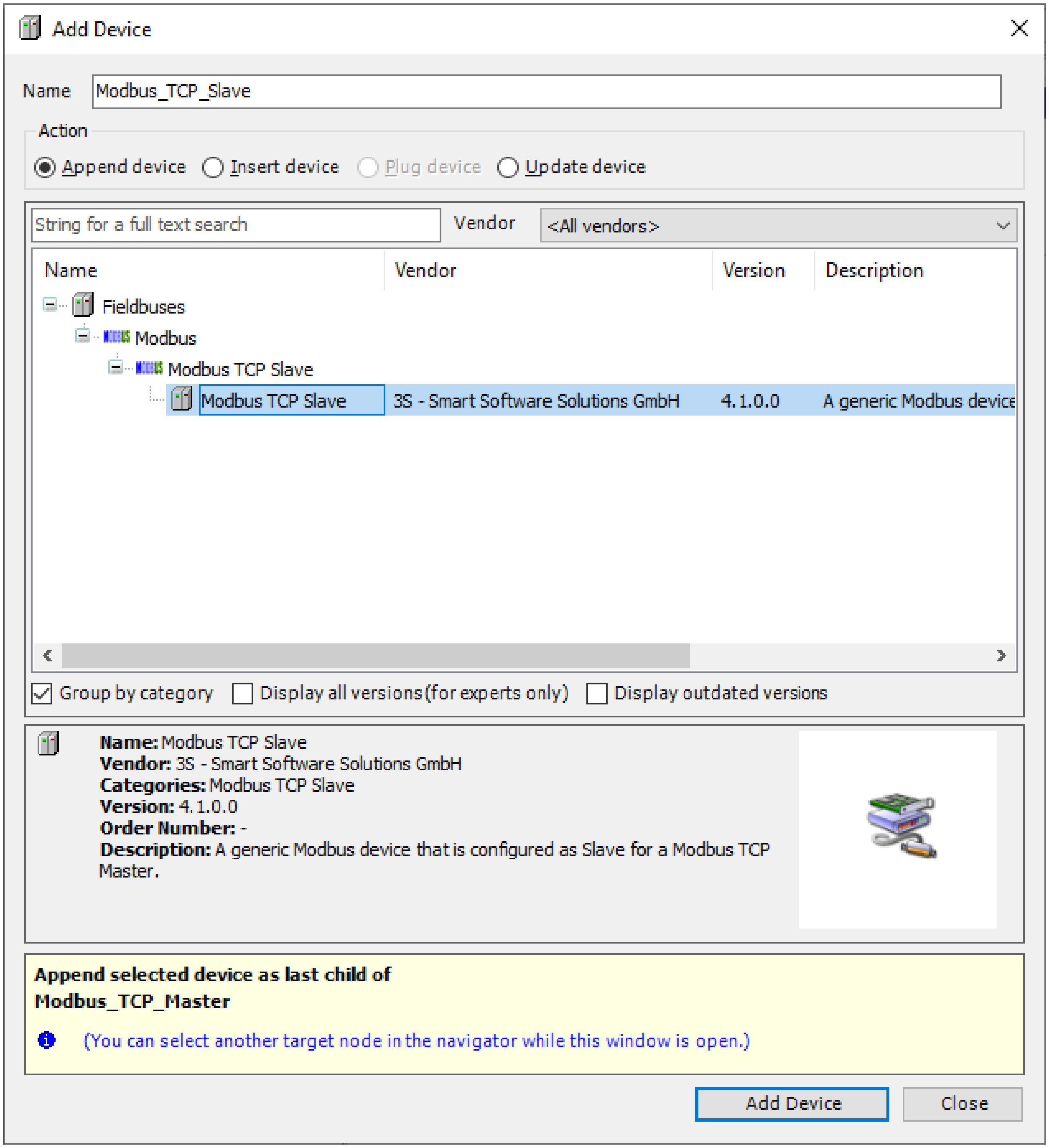
Modbus TCP

Response timeout (ms): 1000

Socket timeout (ms): 10

☑ Auto-reconnect

7. Now the slave needs to link to the master. Add the slave as a node element under the master in the device editor. In the device tree ⇒ Modbus_TCP_Master (right click) ⇒ Add device... ⇒ Modbus TCP Slave



Add Device

Name: Modbus_TCP_Slave

Action

● Append device ○ Insert device ○ Plug device ○ Update device

String for a full text search Vendor <All vendors>

Name	Vendor	Version	Description
Fieldbuses			
Modbus			
Modbus TCP Slave			
Modbus TCP Slave	3S - Smart Software Solutions GmbH	4.1.0.0	A generic Modbus device

☑ Group by category □ Display all versions (for experts only) □ Display outdated versions

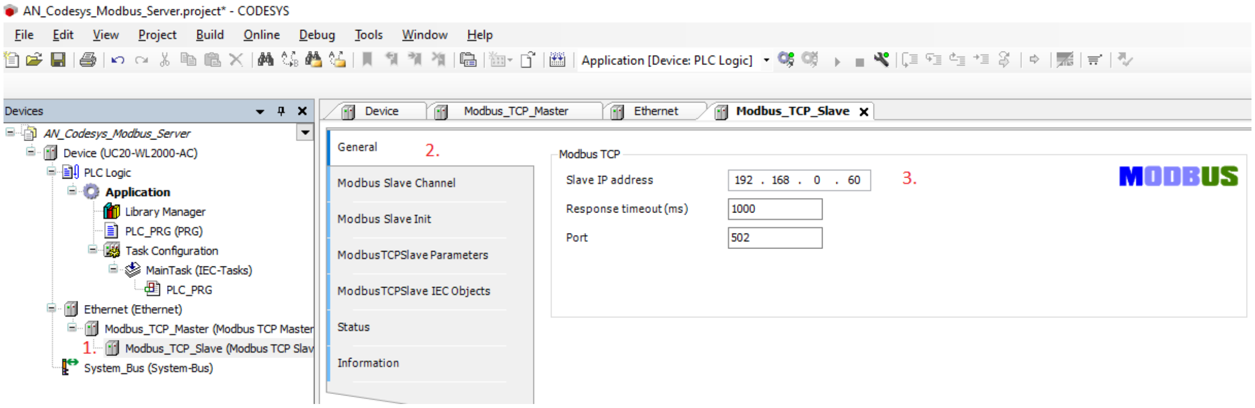
Name: Modbus TCP Slave
Vendor: 3S - Smart Software Solutions GmbH
Categories: Modbus TCP Slave
Version: 4.1.0.0
Order Number: -
Description: A generic Modbus device that is configured as Slave for a Modbus TCP Master.

Append selected device as last child of Modbus_TCP_Master

(You can select another target node in the navigator while this window is open.)

Figure 6: Modbus slave

8. Now there is a slave element in the device tree. Open the slave device editor and enter the IP address of the slave device. Also check if the same port is selected for both devices (default 502).



Modbus TCP

Slave IP address 192 . 168 . 0 . 60

Response timeout (ms) 1000

Port 502

Figure 7: IP Address

The General tab has the following settings for Modbus TCP:

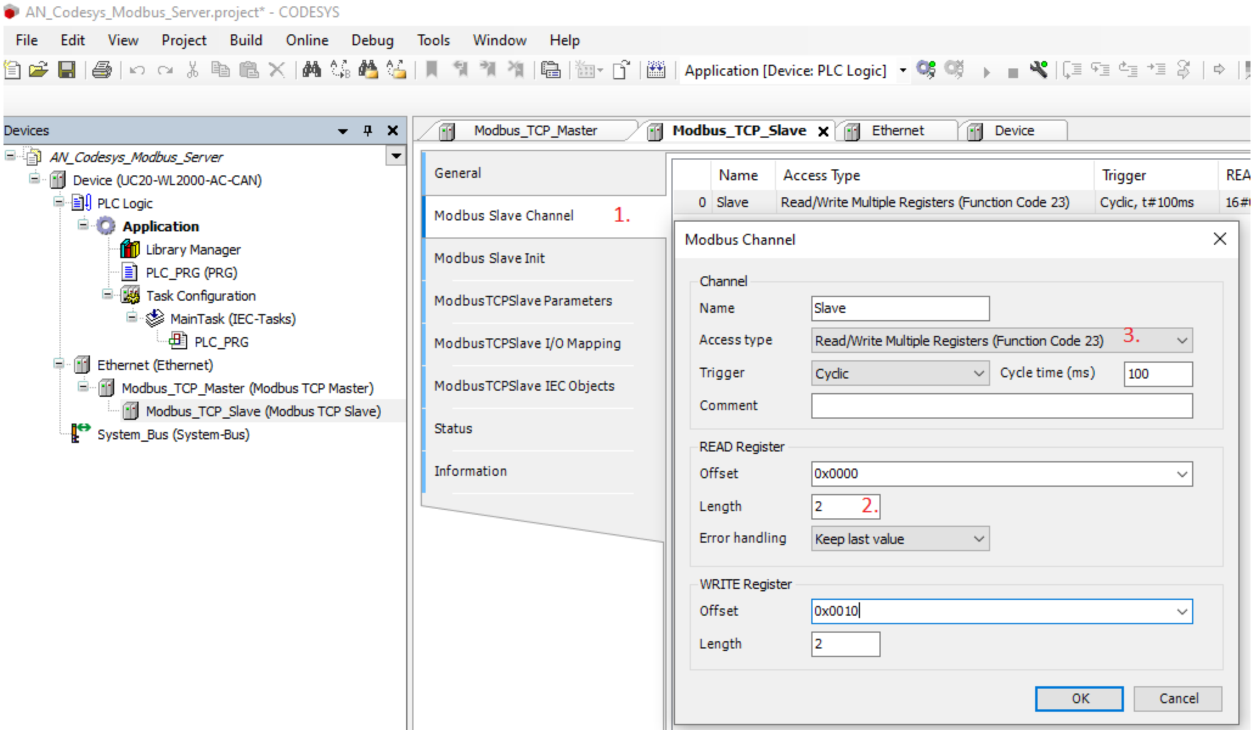
- Slave IP Address - the IP address of the slave.
- Response Timeout - the time interval in which the master waits for the response of the slave device. The timeout specified here overrides the general "Response Timeout" setting of the associated master.
- Port - Port number of the slave (default 502)

3.1 Function codes

Now the function codes to transmit data needs to be specified. To do this, add the channels in the tab "Modbus Slave Channel". There are several function codes. Please see the Modbus specification for more information.

In the example program, the function code 23 (Read/Write Multiple Registers) to read and write values simultaneously is used, the data length is 4 bytes (2 words), and the register address starts for reading at 0 and writing at 16#10.

Of course, you need to adjust the values according to the amount of data you need. Click in the menu Modbus Slave Channel ⇒ Add Channel... and you will see the following dialogue with the parameters already mentioned above.


AN_Codesys_Modbus_Server.project* - CODESYS
File Edit View Project Build Online Debug Tools Window Help
Application [Device: PLC Logic]
Devices
AN_Codesys_Modbus_Server
Device (UC20-WL2000-AC-CAN)
PLC Logic
Application
Library Manager
PLC_PRG (PRG)
Task Configuration
MainTask (IEC-Tasks)
PLC_PRG
Ethernet (Ethernet)
Modbus_TCP_Master (Modbus TCP Master)
Modbus_TCP_Slave (Modbus TCP Slave)
System_Bus (System-Bus)
Modbus_TCP_Master
Modbus_TCP_Slave
Ethernet
Device
General
Modbus Slave Channel
1.
Modbus Slave Init
ModbusTCPSlave Parameters
ModbusTCPSlave I/O Mapping
ModbusTCPSlave IEC Objects
Status
Information
2.
3.
OK
Cancel

	Name	Access Type	Trigger	REA
0	Slave	Read/Write Multiple Registers (Function Code 23)	Cyclic, t#100ms	16#

Modbus Channel

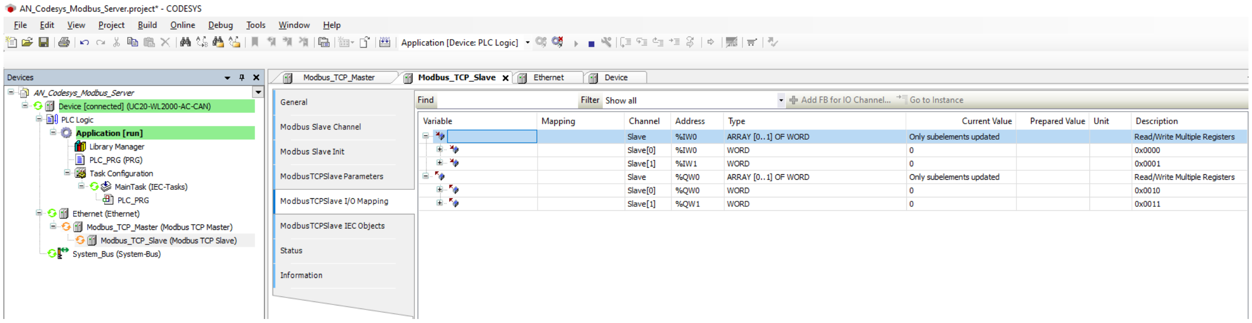
Channel
Name: Slave
Access type: Read/Write Multiple Registers (Function Code 23)
Trigger: Cyclic   Cycle time (ms): 100
Comment:

READ Register
Offset: 0x0000
Length: 2
Error handling: Keep last value

WRITE Register
Offset: 0x0010
Length: 2

Figure 8: Add Channel

The project for the Modbus TCP Master application is configured and now ready for download to the controller.



Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Unit	Description
		Slave	%IW0	ARRAY [0..1] OF WORD	Only subelements updated			Read/Write Multiple Registers
		Slave[0]	%IW0	WORD	0			0x0000
		Slave[1]	%IW1	WORD	0			0x0001
		Slave	%QW0	ARRAY [0..1] OF WORD	Only subelements updated			Read/Write Multiple Registers
		Slave[0]	%QW0	WORD	0			0x0010
		Slave[1]	%QW1	WORD	0			0x0011

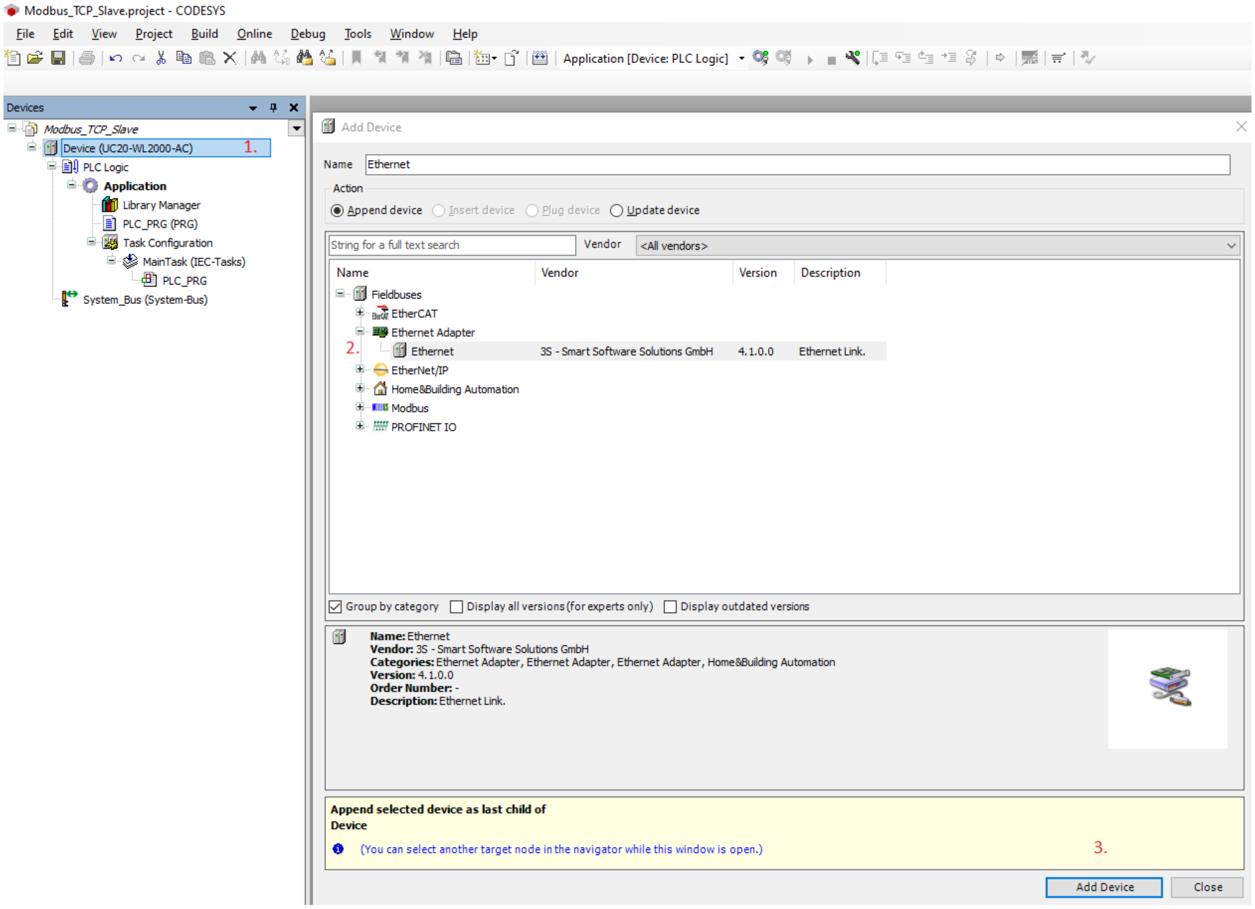
Note:

If the CODESYS license is not activated, the icon at Modbus TCP device is orange after download and login. With license the icon should be green.

4 Modbus TCP Slave (Server) Configuration

Open a second standard CODESYS project for the Modbus TCP Slave device.

1. Insert an "Ethernet node" in the device tree ⇒ Device (right-click) ⇒ Add device...



Modbus_TCP_Slave.project - CODESYS

Add Device

Name: Ethernet

Action: ● Append device ○ Insert device ○ Plug device ○ Update device

Name	Vendor	Version	Description
Ethernet	3S - Smart Software Solutions GmbH	4.1.0.0	Ethernet Link.

☑ Group by category □ Display all versions (for experts only) □ Display outdated versions

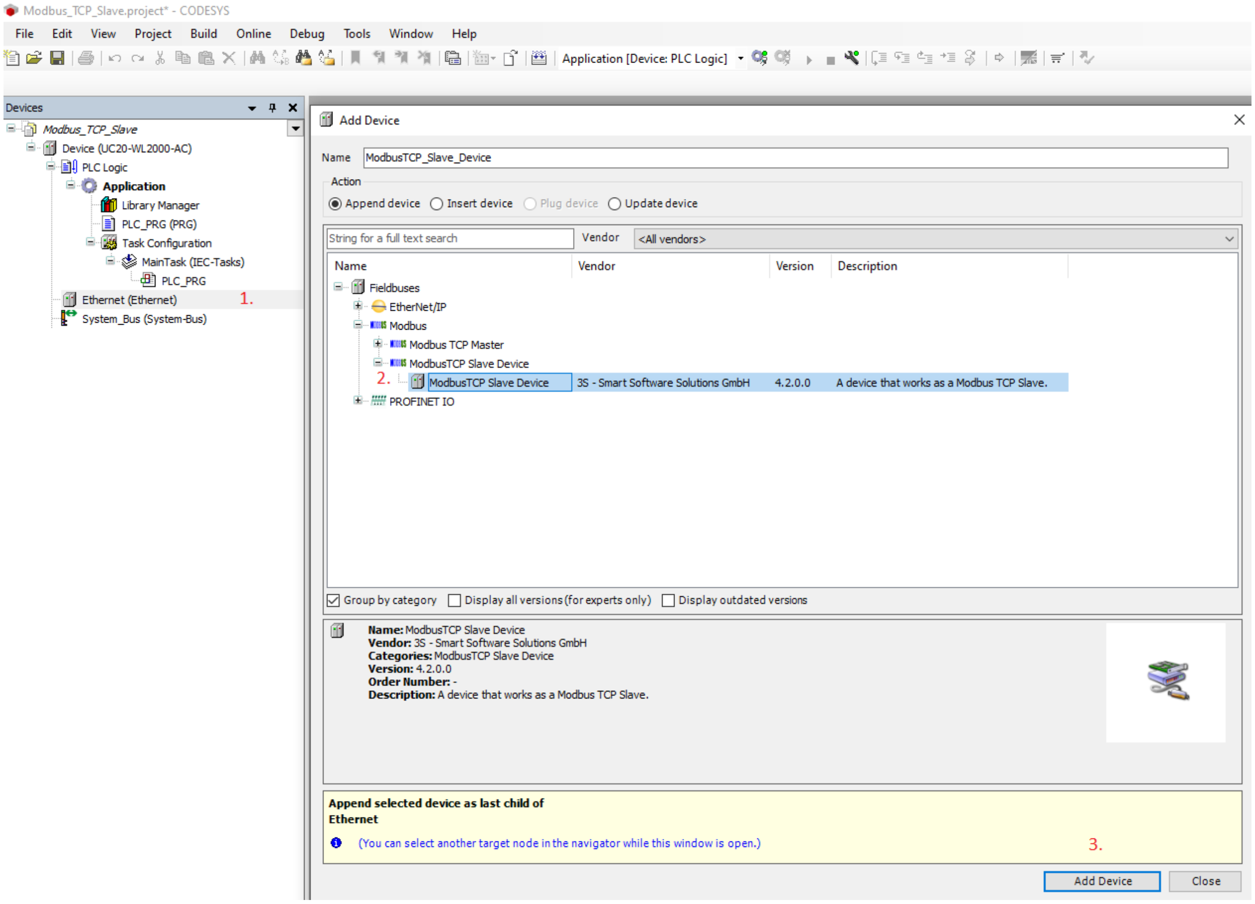
Name: Ethernet
Vendor: 3S - Smart Software Solutions GmbH
Categories: Ethernet Adapter, Ethernet Adapter, Ethernet Adapter, Home&Building Automation
Version: 4.1.0.0
Order Number: -
Description: Ethernet Link.

Append selected device as last child of
Device

(You can select another target node in the navigator while this window is open.)

Figure 9: Add Ethernet adapter

2. Now add the Modbus TCP slave device to the “ethernet adapter”. ⇒ Ethernet (select) ⇒ right-click ⇒ Attach Device... ⇒ Modbus ⇒ Modbus TCP Slave Device ⇒ Add Device ⇒ Close.



Name: ModbusTCP_Slave_Device

Action: ● Append device ○ Insert device ○ Plug device ○ Update device

String for a full text search

Vendor: <All vendors>

Name	Vendor	Version	Description
Fieldbuses			
EtherNet/IP			
Modbus			
Modbus TCP Master			
ModbusTCP Slave Device			
ModbusTCP Slave Device	3S - Smart Software Solutions GmbH	4.2.0.0	A device that works as a Modbus TCP Slave.
PROFINET IO			

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

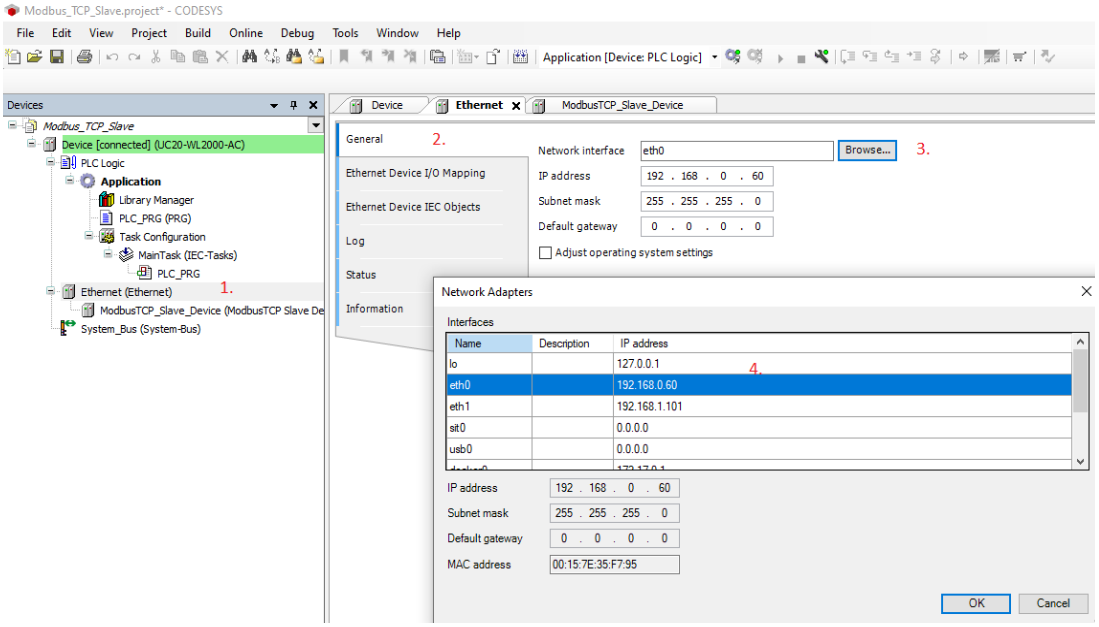
Name: ModbusTCP Slave Device
Vendor: 3S - Smart Software Solutions GmbH
Categories: ModbusTCP Slave Device
Version: 4.2.0.0
Order Number: -
Description: A device that works as a Modbus TCP Slave.

Append selected device as last child of
Ethernet

(You can select another target node in the navigator while this window is open.)

Figure 10: Add Modbus TCP client

3. Edit the network settings in the Ethernet node ⇒ Ethernet (select) ⇒ General ⇒ Network interface [...]. Select the Modbus communication port of your device.


Modbus_TCP_Slave.project* - CODESYS
File Edit View Project Build Online Debug Tools Window Help
Application [Device: PLC Logic]
Devices
Modbus_TCP_Slave
Device [connected] (UC20-WL2000-AC)
PLC Logic
Application
Library Manager
PLC_PRG (PRG)
Task Configuration
MainTask (IEC-Tasks)
PLC_PRG
Ethernet (Ethernet)
1.
ModbusTCP_Slave_Device (ModbusTCP Slave De
System_Bus (System-Bus)
Device
Ethernet
ModbusTCP_Slave_Device
General
2.
Ethernet Device I/O Mapping
Ethernet Device IEC Objects
Log
Status
Information
Browse...
3.
4.
OK
Cancel

Network interface: eth0
IP address: 192 . 168 . 0 . 60
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 0 . 0 . 0 . 0
☐ Adjust operating system settings

Network Adapters

Interfaces

Name	Description	IP address
lo		127.0.0.1
eth0		192.168.0.60
eth1		192.168.1.101
sit0		0.0.0.0
usb0		0.0.0.0
docker0		172.17.0.1

IP address: 192 . 168 . 0 . 60
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 0 . 0 . 0 . 0
MAC address: 00:15:7E:35:F7:95

Figure 11: Edit Network Settings

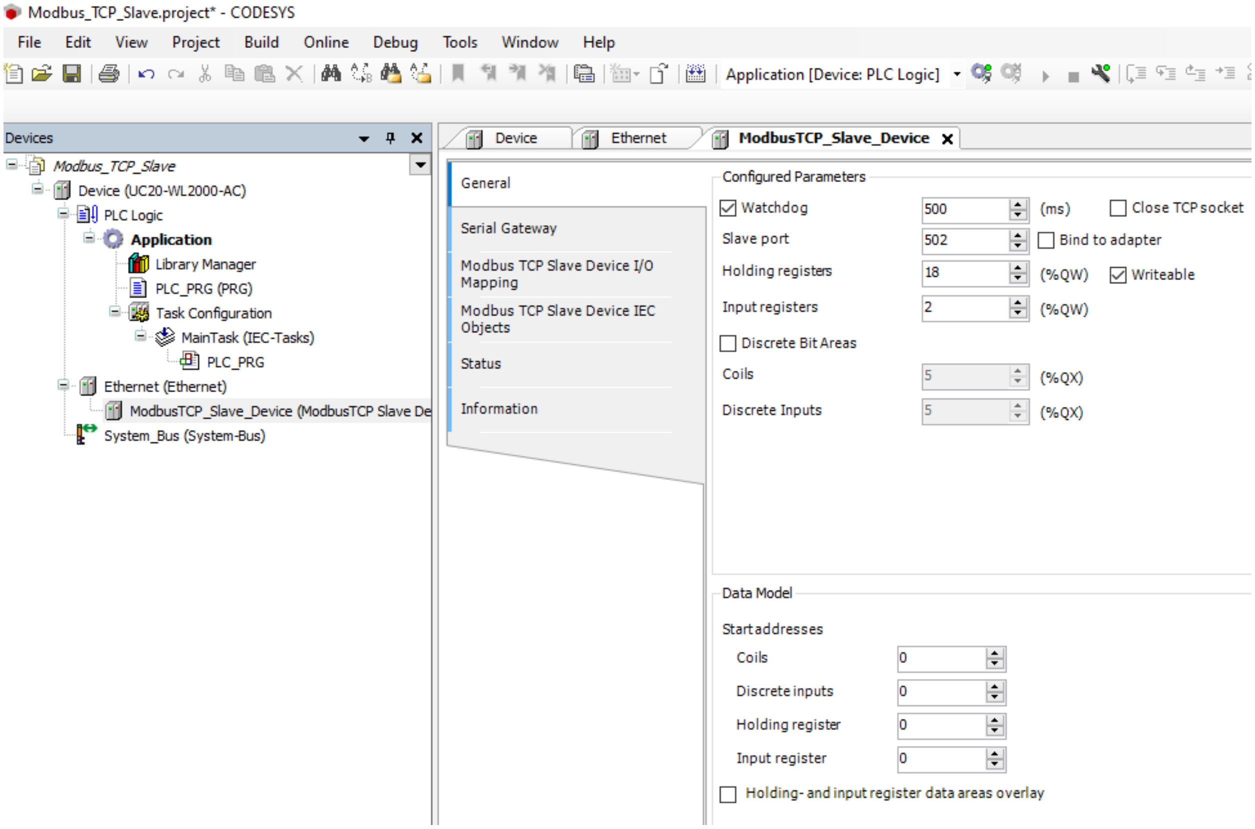
Note:
Use an IP address of the same subnet, but not the same IP Address.

4. Now, some Modbus Slave parameters need to be set. Open the configuration menu by double-clicking on the entry of the Modbus TCP slave in the device tree.

   The tab "General" has the following parameters:

- Watchdog - the maximum amount of time the slave unit expects a Modbus message.
- Slave Port - port number of the slave (default 502).
- Holding Registers - Number of holding registers
- Writeable – Write access to the holding register
- Input Registers - Number of input registers

Edit the settings of the slave device. In the example the parameters were selected to match the Modbus TCP Master settings.


Modbus_TCP_Slave.project* - CODESYS
File
Edit
View
Project
Build
Online
Debug
Tools
Window
Help
Application [Device: PLC Logic]
Devices
Device
Ethernet
ModbusTCP_Slave_Device
General
Serial Gateway
Modbus TCP Slave Device I/O Mapping
Modbus TCP Slave Device IEC Objects
Status
Information

Modbus_TCP_Slave
Device (UC20-WL2000-AC)
PLC Logic
Application
Library Manager
PLC_PRG (PRG)
Task Configuration
MainTask (IEC-Tasks)
PLC_PRG
Ethernet (Ethernet)
ModbusTCP_Slave_Device (ModbusTCP Slave De
System_Bus (System-Bus)

Configured Parameters

☑ Watchdog 500 (ms) □ Close TCP socket
Slave port 502 □ Bind to adapter
Holding registers 18 (%QW) ☑ Writeable
Input registers 2 (%QW)
□ Discrete Bit Areas
Coils 5 (%QX)
Discrete Inputs 5 (%QX)

Data Model

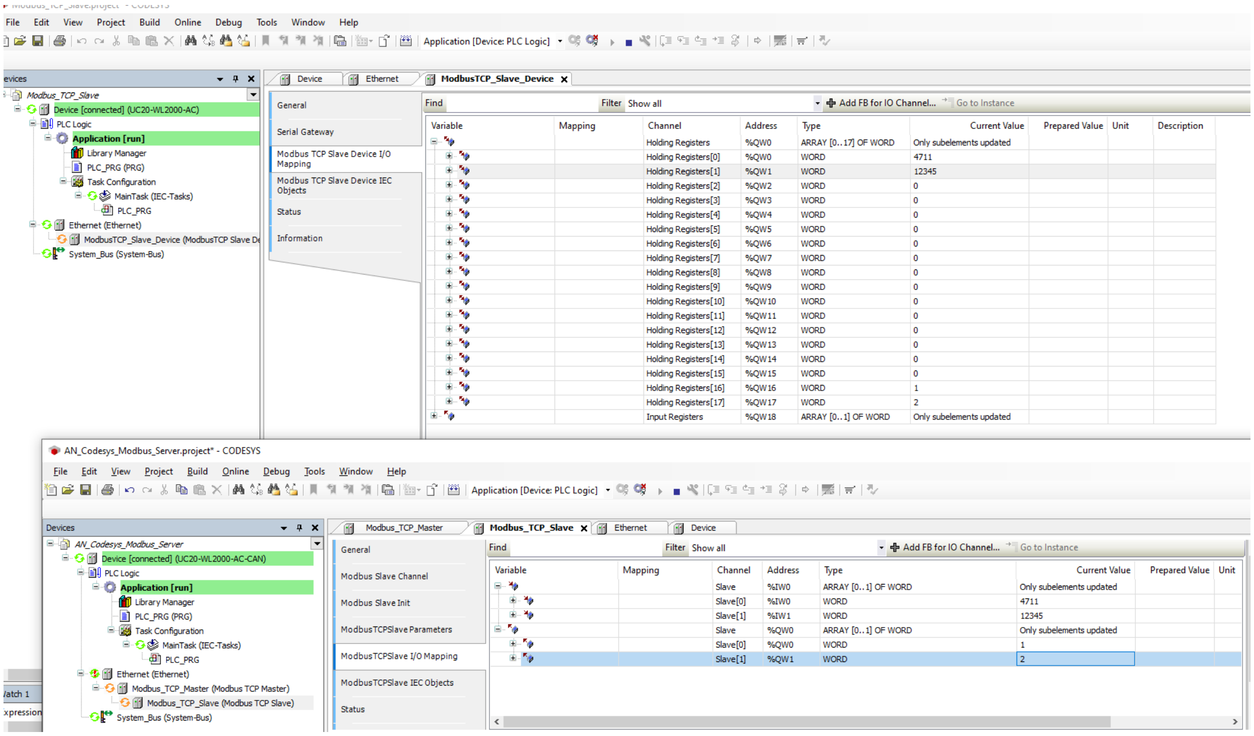
Start addresses
Coils 0
Discrete inputs 0
Holding register 0
Input register 0
□ Holding- and input register data areas overlay

Figure 12: Modbus Slave settings

Depending on which data model you choose, you must adjust the parameters there. If the memory blocks are separated, you must specify different start addresses for each function code (depending on the arrangement and size of the sections).
In the example the start addresses remains at "0". Now you can use the individual function codes to access a WORD or a BIT of your memory area.

5 Function test

The configuration is now complete. The program for master and slave can be loaded to the controllers. To write values to the registers and check the functionality, open the IO-mapping of both devices and login.


File Edit View Project Build Online Debug Tools Window Help
Application [Device: PLC Logic]
Devices
Modbus_TCP_Slave
Device [connected] (UC20-WL2000-AC)
PLC Logic
Application [run]
Library Manager
PLC_PRG (PRG)
Task Configuration
MainTask (IEC-Tasks)
PLC_PRG
Ethernet (Ethernet)
ModbusTCP_Slave_Device (ModbusTCP Slave De
System_Bus (System-Bus)
Device
Ethernet
ModbusTCP_Slave_Device
General
Serial Gateway
Modbus TCP Slave Device I/O Mapping
Modbus TCP Slave Device IEC Objects
Status
Information
Find
Filter Show all
Add FB for IO Channel...
Go to Instance
AN_Codesys_Modbus_Server.project* - CODESYS
AN_Codesys_Modbus_Server
Device [connected] (UC20-WL2000-AC-CAN)
Modbus_TCP_Master (Modbus TCP Master)
Modbus_TCP_Slave (Modbus TCP Slave)
Modbus_TCP_Master
Modbus_TCP_Slave
Modbus Slave Channel
Modbus Slave Init
ModbusTCPSlave Parameters
ModbusTCPSlave I/O Mapping
ModbusTCPSlave IEC Objects
Watch 1
Expression

Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Unit	Description
		Holding Registers	%QW0	ARRAY [0..17] OF WORD	Only subelements updated			
		Holding Registers[0]	%QW0	WORD	4711			
		Holding Registers[1]	%QW1	WORD	12345			
		Holding Registers[2]	%QW2	WORD	0			
		Holding Registers[3]	%QW3	WORD	0			
		Holding Registers[4]	%QW4	WORD	0			
		Holding Registers[5]	%QW5	WORD	0			
		Holding Registers[6]	%QW6	WORD	0			
		Holding Registers[7]	%QW7	WORD	0			
		Holding Registers[8]	%QW8	WORD	0			
		Holding Registers[9]	%QW9	WORD	0			
		Holding Registers[10]	%QW10	WORD	0			
		Holding Registers[11]	%QW11	WORD	0			
		Holding Registers[12]	%QW12	WORD	0			
		Holding Registers[13]	%QW13	WORD	0			
		Holding Registers[14]	%QW14	WORD	0			
		Holding Registers[15]	%QW15	WORD	0			
		Holding Registers[16]	%QW16	WORD	1			
		Holding Registers[17]	%QW17	WORD	2			
		Input Registers	%QW18	ARRAY [0..1] OF WORD	Only subelements updated			

Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Unit
		Slave	%IW0	ARRAY [0..1] OF WORD	Only subelements updated		
		Slave[0]	%IW0	WORD	4711		
		Slave[1]	%IW1	WORD	12345		
		Slave	%QW0	ARRAY [0..1] OF WORD	Only subelements updated		
		Slave[0]	%QW0	WORD	1		
		Slave[1]	%QW1	WORD	2		

Figure 13: Test

It is possible to force values for writing data to the slave (Holding Register 16 and 17) and it is also possible to read data from the slave (Register 0 and 1).