# Modbus/TCP - Network Master PLC500

**Application Note** 







## **Master Modbus/TCP - Application Note**

Series: PLC500 Language: English Document: 10010165984 / 00

Publication Date: 10/2022



The information below describes the reviews made in this manual.

Version	Revision	Description
-	R00	First edition

## Contents

1	INTRODUCTION	5
	1.1 MODBUS/TCP NETWORK COMPONENTS	5
	1.2 ETHERNET INTERFACE	5
	1.3 MODBUS/TCP NETWORK ARCHITECTURE	5
2	MODBUS/TCP SETTINGS	7
	2.1 CREATING A PROJECT IN CODESYS	7
3	MASTER MODBUS/TCP SETTINGS	9
	3.1 I/O SETTINGS	9
4	MONITORING	11
	4.1 VARIABLE MONITORING	11
	4.2 COMMUNICATION ERRORS	11

## 1 INTRODUCTION

This application manual describes how to operate the PLC500 as a Modbus/TCP **network master**. For the PLC500 product settings, follow the steps described in this document to configure the Modbus/TCP network properly.



#### ATTENTION!

This application manual is intended for professionals trained in industrial networks. The devices must be installed and configured according to manufacturer's manual.

#### 1.1 MODBUS/TCP NETWORK COMPONENTS

For the network passive components—cables, connectors, termination resistors, power supply—it is recommended to use only components certified for industrial applications. See the product documentation for information on the proper installation of the Ethernet network.

For a detailed description of the network operation Modbus/TCP and its settings, visit Codesys webpage at: https://help.codesys.com/.

#### **1.2 ETHERNET INTERFACE**

As indicated in Figure 1.1, Ethernet connections are used for Modbus/TCP communication. Initially, each port has the IPv4 address contained in Table 1.1, which can be changed at any time using the Codesys software or via website.



Figure 1.1: Ethernet interface connector.

Table 1.1: Default address for Ethernet ports.

Connection	Default IPv4 address
ETH1	192.168.1.10
ETH1	192.168.2.10

#### 1.3 MODBUS/TCP NETWORK ARCHITECTURE

Figure 1.2 shows the topology of the network used. The computer programs the two devices through the Ethernet interface, and the communication between the devices is done by the same way.



Figure 1.2: Network components.



Codesys:

#### NOTE!

This application manual is intended for the PLC500 and the Codesys programming tool, we recommend using the Codesys V3.5 SP18 or higher. If you need more information about the Modbus/TCP communication protocol, see its manuals.

## 2 MODBUS/TCP SETTINGS

#### 2.1 CREATING A PROJECT IN CODESYS

In the Codesys *software*, create a new project and choose the directory and application name. Then select the PLC500-WEG device and the desired programming language, as shown in Figure 2.1.

1 New Project X	
Categories       Templates         Libraries       Implates         Projects       Implates         Empty project       Implates         A project containing one device, one application, and an empty implementation for PLC_EBG.	Standard Project       ×         You are about to create a new standard project. This wizard will create the following objects within this project:       • One programmable device as specified below         • One program PLC_PRG in the language specified below       • A cyclic task which calle PLC_PRG         • A vicit cask which caller PLC_PRG       • A reference to the newest version of the Standard library currently installed.
Name Example Location C:\Users\user\Documents\CODESYS	Device         PLC500-WEG (WEG Drives & Controls)            PLC_PRG in         Ladder Logic Diagram (LD)            OK         Cancel





#### NOTE!

In case the device PLC500 is not available in the Cosys options yet, you must download and install the configuration file. See the **Product Manual** for the necessary steps and settings.

With the PLC500 device selected, you will have a project with the available networks already preset, as shown in Figure 2.2.



Figure 2.2: Network interfaces for the plc500.

In the ETH1 or ETH2 interface, add "Modbus TCP Master", as shown in Figure 2.3.



Figure 2.3: Codesys - Modbus/RTU Configuration Step 1.

#### In "Modbus TCP Master", add "Modbus TCP Slave", as shown in Figure 2.4.

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	1	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	AKK IDK DECLK CH	16= 6= 0	+= Q!	X
Devices 🗸 🗸	×	Nam	Modbus_TCP_Slave			
E→[] Example	•	Act	end device O Insert device O Plug device O Update de	device		
Expansions (Expansions)     Enti (ETH1)     Modus_TCP_Master (Modbus TCP Master)		Na But	Fieldbuses	1	/ersion	Description
CAN (CAN)	8 60 60	Copy	Modbus TCP Slave     Modbus TCP Slave     Software Solution	tions GmbH 4	. 1.0.0	A generic Modbus device that is configured as Slave for a Modbus TCP Mas
	×	Delete				
	6	Refactoring  Properties	ip by category 🔄 Display all versions (for experts only) 📃	Display outda	ted versi	< encoded and enco
		Add Object Add Folder	Name: Modbus TCP Slave Vendor: 35 - Smart Software Solutions GmbH			^
		Insert Device Disable Device	Version: 4.1.0.0 Order Number: - Description: A generic Modbus device that is configured as Sla	lave for a Modbus	TCP Mas	ter. 🗸
	ß	Edit Object Edit Object With	d selected device as last child of s_TCP_Master fou can select another target node in the navigator while this	is window is ope	n.)	
<		Edit IO mapping Import mappings from CSV Export mappings to CSV			_	Add Device Close

Figure 2.4: Codesys - Modbus/RTU Configuration Step 2.

At this moment, the chosen ETH interface should have the items indicated in Figure 2.5.



Figure 2.5: Project configuration in Codesys.



#### 3 MASTER MODBUS/TCP SETTINGS

The network must be set with valid addresses for each connected master and slave device. Define them with unique addresses and select the allowed IPv4 address range; otherwise, the network will have problems. In figure 3.1, the master device settings are defined, and the **Browse** box is used when selecting the Ethernet interface while defining the IPv4 address.

Ethernet X Modbus_	CP_Master	Modbus_TCP_Slave	Device						
General	Network interface	eth1	В	rowse					
Log	IP address	192 . 168 . 1 . 10	Network Ac	lanters	×				
Status	Subnet mask	255 . 255 . 255 . 0	Interferen						
Ethernet Device I/O Mapping	Default gateway	0.0.0.0	Name	Description	IP address				
Rhamat Davies IEC Objects	Adjust operating	g system settings	reserv1		127.0.0.1				
Ethemet Device IEC Objects			eth1		192.168.1.10				
Information			eth2		192.168.2.10				
			usb2		192.168.234.234				
			IP address Subnet mae Default gate MAC addre	192 . sk 255 . sway 0 . ss 00:01:	168       1       10         .255       .255       .0         .0       .0       .0         C028:53:13       OK       Cancel				

Figure 3.1: Network settings for the device programmed in Codesys.

After this step, the network for the slave device is configured. Each one must have its own address and be within the available IPv4 address range. The other settings, such as baud rate, will be managed automatically by the network.

Ethernet Modbus_TC	CP_Master Modbus_TC	P_Slave X	
General	Modbus TCP		
Modbus Slave Channel	Slave IP address	192 . 168 . 1 . 11	MODBUS
Modbus Slave Init	Response timeout (ms)	1000	
ModbusTCPSlave Parameters	Port	502	
ModbusTCPSlave I/O Mapping			
ModbusTCPSlave IEC Objects			
Status			
Information			



#### 3.1 I/O SETTINGS

Variables and memory addresses set for a slave must be declared in the memory intervals reserved for the transmission variables. Then, addresses are read and written in the slave device register. Figure 3.3 shows the Modbus functions for reading and writing these data to the slave device.







#### NOTE!

The availability of Modbus addresses may vary from device to device. During the configuration, refer to the product manual to obtain information on the available addresses.

Configuring the slave in **Modbus\_TCP\_Slave** in Figure 3.4, the functions are declared by going to **Modbus Slave Channel** and following the functions required for the project in Figure 3.3.

Device Ethernet Mo	odbus_TCP_Master	Modb	us_TCP_Slave	×						
General	lame Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Of	ifset Length	Comment		
Modbus Slave Channel	Modbus Channel					×				
Modbus Slave Init	Channel									
ModbusTCPSlave Parameters	Name	Channel 0								
ModbusTCPSlave IEC Objects	Access type	Read Holding	Registers (Funct	on Code 3)	~					
Status	Comment	Cyclic	~	Cycle time (m	s) <u>100</u>					
Information	READ Register									
	Length	1			~					
	Error handling	Keep last valu	ie v							
Mo	WRITE Register						Add Channel	Delete		Edit
Messages - Total 2 error(s), 8 warning(s), 0 messag	Offset	0x0000			~					<b>-</b> ₽ >
Save Project	Length	1								
Description				<u>0</u>	Cancel			Project	Object	Position
Eiguro 3	1. Declarat	ion of a	draccad	transm	itted by th	o notu	ork slava	device		

#### MONITORING 4

#### VARIABLE MONITORING 4.1

After setting the Modbus/TCP network and declaring the transmission addresses, we can monitor and control the data transmitted by the devices. You must declare the variables that you want to assign to the addresses. On the **Mapping** tab shown in Figure 4.1, it is possible to check and write the values in the network variables.



Figure 4.1: Declaration of variables transmitted by the network slave device.

The variables declared in the network can be monitored through two methods: first, to add the variable values to the Codesys program and monitor them online; second, to enable Always Update Variables at the bottom of the previous page, as shown in Figure 4.2.

General	Find		Filter Show a	ll		A	dd FB for IO Channel → 🗍 Go to Ir	istance
Modbus Slave Channel	Variable ⊞¥≱	Mapping	Channel Channel 0	Address %IW18	Type ARRAY [00] OF WORD	Unit	Description Read Holding Registers	
Modbus Slave Init	± - *		Channel 1	%QW14	ARRAY [00] OF WORD		Write Single Register	
ModbusTCPSlave Parameters								
ModbusTCPSlave I/O Mapping								
ModbusTCPSlave IEC Objects								
Status								
Status						Л		
Status Information					•	Û	,	
Status					•	Л V	,	
Status			Re	set Mapping	Always update variab		parent device setting	
Status			Re	set Mapping	Always update variab		parent device setting parent device setting led.1 (use hur code tack if ont used in	any task)
Status Information	Messages - Total 14 error	or(s), 8 warning(s), 0 mess	Re	set Mapping	Always update variab		parent device setting parent device setting led 1 (use bus cycle task if not used in led 2 (always in bus cycle task)	any task)

-igure 4.2: Selecting the option to always update the value of the

#### 4.2 **COMMUNICATION ERRORS**

The status of the networks in the Codesys software PLC500can be monitored in Devices, which indicates the status of each communication step and the Status, as shown in Figure 4.3. If you encounter connection problems, connect to the PLC500 and access the Status and Log tab within the created items; Codesys will report the problem hindering the communication.



Figure 4.3: Communication error indication.



#### NOTE!

If you encounter problems, check that the cables are properly connected and that the respective network LED is turned on.

When the settings are correct and the devices identify each other, the application should show as in Figure 4.4.



Figure 4.4: Communication correctly configured and devices communicating.



WEG Drives & Controls - Automation LTDA. Jaraguá do Sul - SC - Brazil Phone 55 (47) 3276-4000 - Fax 55 (47) 3276-4020 São Paulo - SP - Brazil Phone 55 (11) 5053-2300 - Fax 55 (11) 5052-4212 automacao@weg.net www.weg.net