

Modbus/RTU - Network Master PLC500, PLC500ED, PLC500MC PLC410

Application Note





Application Note

PLC410, PLC500, PLC500ED, PLC500MC

Document: 10010165648

Revision: 01

Publication Date: 12/2024

The information below describes the reviews made in this manual.

Version	Revision	Description
1.4.0	R01	Information update and inclusion of PLC models PLC410, PLC500ED and PLC500MC.
1.0.0	R00	First edition.

1	INTRODUCTION	1-1 1-1 1-2 1-2 1-2
2	RS485 INTERFACE 2.1 RS485 SERIAL INDICATION 2.2 MODBUS/RTU NETWORK INSTALLATION 2.3 EXAMPLE MODBUS/RTU NETWORK ARCHITECTURE	2-1 2-1 2-2 2-3
3	PROJECT IN CODESYS 3.1 CREATING A PROJECT IN CODESYS 3.2 ADDING MASTER DEVICE 3.3 ADDING SLAVE DEVICE	3-1 3-1 3-2 3-3
4	MODBUS/RTU CONFIGURATION 4.1 TERMINATION RESISTORS 4.2 NETWORK CONFIGURATION 4.3 MODBUS CHANNEL CONFIGURATION	4-1 4-1 4-1 4-2
5	MONITORING 5.1 MONITORING VARIABLES 5.2 COMMUNICATION ERRORS	5-1 5-1 5-2

1 INTRODUCTION

This Application Note aims to assist in the use of the **Modbus/RTU** protocol on WEG PLCs, models PLC410, PLC500, PLC500ED, and PLC500MC. Throughout this document, the PLC500 is used as an example. However, the information presented is applicable to the other PLC models described earlier.

This document presents information about the Modbus/RTU protocol, description of the RS485 interface, installation recommendations, and a guide to establish Modbus/RTU communication between the PLC500 (master) and the PLC300 (slave) in **CODESYS**[®].

For more information about the hardware, interfaces, and communication protocols, please refer to the User Manual of the respective product, available on the WEG website. For a more in-depth and detailed description of the operation of the Modbus/RTU network, access the online help at CODESYS Online Help.



ATTENTION!

This application note is intended for professionals trained in industrial networks. The installation and configuration of devices should be performed according to the manufacturer's manual.



NOTE!

This application note is intended for the CODESYS programming tool. It is recommended to use **CODESYS V3.5 SP19** or higher. It is also recommended to use the latest available version of the configuration libraries for the Modbus/RTU network.

1.1 TERMS AND DEFINITIONS USED

CODESYS: Programming software for the PLC500. A programming platform that allows the development, configuration, and monitoring of solutions for industrial automation and system integration.

kbps: Kilobits per second.

m: meter.

ms: millisecond = 0.001 seconds.

s: second.

USB: stands for "Universal Serial Bus", a type of serial communication interface designed to operate based on the "Plug and Play" concept.

1.2 NUMERIC REPRESENTATION

Decimal numbers are represented using digits without any prefix. Hexadecimal numbers are represented with the prefix '0x'.

1.3 MODBUS/RTU DOCUMENTS

It is recommended to consult the documents related to the Modbus/RTU network presented in Table 1.1 on page 1-2.

Document	Version	Source
MODBUS Application Protocol Specification, December 28th 2006	V1.1b	MODBUS.ORG
MODBUS over Serial Line Specification, December 20th 2006.	V1.02	MODBUS.ORG
MODBUS Protocol Reference Guide, June 1996.	Rev. J	MODICON

Table	1.1:	Reference	Documents.
1 01010		1 (010101100	Doodinionto.

1.4 IMPORTANT NOTICE ABOUT CYBERSECURITY AND COMMUNICATIONS

WEG PLCs, models PLC410, PLC500, PLC500ED, and PLC500MC, have the ability to connect and exchange information through networks and communication protocols. While they have been designed and tested to ensure proper operation with other automation systems using the protocols mentioned in this manual, it is essential that the customer understands the responsibilities associated with information and cybersecurity when using this equipment.

Therefore, it is the customer's full responsibility to adopt defense-in-depth strategies and implement policies and measures to ensure the security of the system as a whole, including communications sent and received by the equipment. These measures include, but are not limited to, the installation of firewalls, antivirus and antimalware programs, data encryption, authentication control, and physical access control for users.

WEG and its affiliates are not responsible for damages or losses arising from cybersecurity breaches, including, but not limited to, unauthorized access, intrusion, data leakage and/or theft, denial of service, or any other form of security violation. The use of this product in conditions for which it was not specifically designed is not recommended and may cause damage to the product, network, or automation system.

In this regard, it is crucial that the customer understands that external interventions through third-party programs, such as sniffers or programs with similar actions, have the potential to cause interruptions or restrictions in the functionality of the equipment.

1.5 TRADEMARKS

All other registered trademarks are the property of their respective owners.

2 RS485 INTERFACE

Modbus/RTU communication is performed through the RS485 interface, as shown in Figure 2.1 on page 2-1 for the PLC500 and PLC410. The pinout of the interface is described in Table 2.1 on page 2-1.



Figure 2.1: RS485 connector pinout on the PLC500 and PLC410.

Pin	Label	Function
1	COM	Common (connected to the negative terminal of the Modbus/RTU network)
2	B+	B+ communication signal
3	A-	A- communication signal
4	NC	Not connected (may receive the positive terminal of the Modbus/RTU network)
5	SH	Cable shielding

The RS485 interface is isolated with its own internal power supply, meaning the pin 4 (NC) does not need to be connected. However, it is recommended that pin 1 (COM) be connected to the master and other slaves to maintain the same voltage reference for the Modbus/RTU network.

The PLC410, PLC500, PLC500ED, and PLC500MC allow connection of up to 246 Modbus/RTU slaves. However, for connecting more than 32 devices to the RS485 interface, a repeater must be used.

2.1 RS485 SERIAL INDICATION

The PLC500 and PLC410 have LEDs on the front panel used to indicate the status of the RS485 (SERIAL) interface. The LED color indications shown in Figure 2.2 on page 2-1 are described in Table 2.2 on page 2-1.



Figure 2.2: LED indicators of the PLC500 and PLC410.

SERIAL LED	STATUS	DESCRIPTION			
OFF	No configuration.	There is no Modbus/RTU communication configuration in the application.			
GREEN	Modbus/RTU communication without failure.	Modbus/RTU communication configured in the application and all elements running without errors/failures.			
ORANGE	Modbus/RTU communication with partial failure.	Modbus/RTU communication configured in the application with some elements having errors/failures.			
RED	Modbus/RTU communication with total failure.	Modbus/RTU communication configured in the application and all elements having errors/failures.			

Table 2.2:	Operation	of the	RS485	interface	indicator	LED
------------	-----------	--------	-------	-----------	-----------	-----

2.2 MODBUS/RTU NETWORK INSTALLATION

The Modbus/RTU network, like many industrial communication networks, due to its frequent use in harsh environments with high electromagnetic interference, requires certain precautions to ensure a low communication error rate during its operation.



ATTENTION!

For passive components of the network such as cables, connectors, termination resistors, and power supplies, it is recommended to use only components certified for industrial applications.

The RS485 interface of the PLC500 programmable logic controller can communicate using the rates defined in Table 2.3 on page 2-2.

Communication Rate
1200 kbps
2400 kbps
4800 kbps
9600 kbps
19200 kbps
38400 kbps
57600 kbps
115200 kbps

Table 2.3: Supported RS485 communication rates.

For connecting the **A-** and **B+** signals, shielded twisted pair (STP) cables should be used. For equipment requiring external power for the RS485 interface, the positive and negative pair should also be twisted.

The maximum cable length depends on the communication rate. For a rate of 9600 kbps and an AWG 26 (or larger) wire gauge, the maximum length is 1000 meters. An AWG 24 wire is sufficient for Modbus data in most cases. Branches should be no longer than 20 meters.

For interconnecting the various network nodes, it is recommended to connect the equipment directly to the main line, without using branches. During cable installation, the cables should not be routed near power cables, as this increases the likelihood of transmission errors due to electromagnetic interference.

An adequate connection to the grounding system is essential to reduce electromagnetic interference in industrial environments. It is important to avoid grounding the cable at multiple points, especially where there are grounds with different voltage potentials.



ATTENTION!

Grounding the negative terminal of the PLC500 power supply and pin 1 of the RS485 interface may cause damage to the device due to loss of isolation in the power supply.



DANGER!

Defective grounding installations can lead to death and malfunction of the Modbus/RTU network.

2.3 EXAMPLE MODBUS/RTU NETWORK ARCHITECTURE

In this application note, an example is given with a PLC500 as the network master and a PLC300 as the network slave. Figure 2.3 on page 2-3 presents the components and architecture of the Modbus/RTU network. A computer programs the PLC500 in CODESYS via a USB cable. The PLC300 also has an isolated RS485 interface with its own internal power supply.



Figure 2.3: Network components.



NOTE!

The PLC300 is configured by default to operate as a Modbus/RTU master. To change this configuration, refer to the documentation on the product page available on the WEG website.

3 PROJECT IN CODESYS

This chapter presents the steps on how to create a project and add the master and slave devices of the Modbus/RTU network in CODESYS.

3.1 CREATING A PROJECT IN CODESYS

To configure and define a Modbus/RTU network, you must first create the project including the PLC500 programmable logic controller. In the CODESYS software, create a new project using the **Standard project** template, choose the directory and name of the application. Then, select the PLC500 device and the desired programming language, as shown in Figure 3.1 on page 3-1.

管 New Project		×		
Categories	Templates	Standard project w Standar C_PRG	d Project You are about to create a new standard project. This wizard will cre objects within this project: - One programmable device as specified below - A program PLC PRG in the language specified below - A project sak which calls PLC PRG - A reference to the newest version of the Standard library currently	× eate the following y installed.
Name PLC500_modbusR1 Location C:\Users\user\Doc	U_master	✓	Device PLCS00 (WEG) PLC_PRG in Ladder Logic Diagram (LD)	× ×
	ОК	Cancel	ОК	Cancel

Figure 3.1: Creating the project in CODESYS.



NOTE!

If the PLC410, PLC500, PLC500ED, or PLC500MC device is not yet available in the CODESYS options, you must download and install the configuration file. Refer to the **Product Manual** for steps and necessary configurations.

With the PLC500 device selected, the project tree will be displayed, as shown in Figure 3.2 on page 3-1.



Figure 3.2: Initial project tree of the PLC500.

3.2 ADDING MASTER DEVICE

In the next step, the device **Modbus_Master_COM_Port** should be added. Right-click on the **RS485** icon and select **Add Device**. A new window will open. Expand the **Miscellaneous** icon using the **+** symbol to reveal the **Modbus Master COM Port** device, as shown in Figure 3.3 on page 3-2. With the device selected, click on **Add Device**.

Devices	•	ά Χ		
PLC500_modbusRTU_ma	ster	•		
E Device (PLC500)				
PLC Logic				
= 💭 Application	1			
Library f	fanag	er	Add F	Device X
	5 (PRC	5) 		
Task Co	Tack		Name M	Modbus_Master_COM_Port
		96	Action	
Setup (Setup)			Anno	
Z I Os (I/Os)			Abbe	
Expansions (Exp	ansior	is)	String for	or a full text search Vendor <all vendors=""> ~</all>
ETH1 (ETH1)			Name	Verder Vering Description
ETH2 (ETH2)				vendor version beschprion
CAN (CAN)				mscelaneous
m RS485 (RS485)	V	Cut		In moutos master, com part wild Tusto. A device that works as a moutos master on a senar com part or a wild FLC
	00 135	Carry		
		Сору		
	••••	Paste		
	×	Delete		
		Refactoring		
	æ	Properties	🕑 Group	up by category Display all versions (for experts only) Display outdated versions
		Add Object		Namer Modius Master COM Port
	\cong	Add Folder	، ۳ I	Vendor: WEG
		Add Device		Categories: Version: 43.0.0
		Disable Device		Order Number: -
		Update Device	· · · ·	Description: A device that works as a modulus master on a senar COM Port of a WEG PLC
	ĥ	Edit Object		
	-	Edit Object With	RS485	a selected device as last child of
		Edit IO mapping	- O (Yr	(ou can select another target node in the navigator while this window is open.)
		Import mappings from CSV		
		Export mappings to CSV		Add Device Close
1	_			

Figure 3.3: Adding Modbus_Master_COM_Port master device.

The device will be added to the project tree right below the **RS485** interface, as shown in Figure 3.4 on page 3-2.



Figure 3.4: Modbus_Master_COM_Port device added to the project tree.

3.3 ADDING SLAVE DEVICE

Right-clicking on the **Modbus_Master_COM_Port** icon adds the slave device **Modbus_Server_COM_Port** to the Modbus/RTU network. In this case, expand the **Modbus** and **Modbus Serial Slave** icons using the **+** symbol until the **Modbus Server COM Port** device appears, as shown in Figure 3.5 on page 3-3. With the device selected, click **Add Device**.



Figure 3.5: Adding Modbus_Server_COM_Port slave device.

At this point, the **RS485** interface should contain the items shown in Figure 3.6 on page 3-3. More slave devices can be added by following the same procedures.



Figure 3.6: Modbus_Server_COM_Port device added to the project tree.

4 MODBUS/RTU CONFIGURATION

This chapter presents the steps to configure a Modbus/RTU network with a PLC500 master and PLC300 slave in CODESYS.

4.1 TERMINATION RESISTORS

For the proper operation of the Modbus/RTU network, it is essential to use termination resistors. You can choose to use an external resistor or enable the internal resistor directly in CODESYS. In the PLC500, the internal resistor can be configured through the **Setup** \rightarrow **Setup Parameters** \rightarrow **Termination Resistors** \rightarrow **RS485** object, as shown in Figure 4.1 on page 4-1.

Setup Parameters	Parameter	Туре	Value	Default Value	Unit	Description
	🗄 🗀 Firmware					
LC Setup IEC Objects	🖳 📴 Date and Time					
Setup I/O Mapping	Termination Resistors					
etap 1/0 Mapping	🔷 🖗 RS485	Enumeration of BYTE	Connected \lor	Not Connected		RS485 termination resistor configuration
	🕸 CAN	Enumeration of BYTE	Not Connected	Not Connected		CAN termination resistor configuration
	🗉 🗀 Ethernet 1		Connected			
	🗉 🚞 Ethernet 2					
	🗄 🗀 Default Route					
	😟 🗀 LEDs Indication					

Figure 4.1: Termination resistor configuration page.

NOTE!

In the PLC300, the internal resistor can be added using switch S1. For more information, refer to the product documentation available on the WEG website.

4.2 NETWORK CONFIGURATION

The network must be configured on the master device to ensure that slave addresses are correctly addressed. Each slave device must be configured with a unique address and operate at the same transmission rate; otherwise, communication failures may occur. Double-clicking on **RS485** sets the communication configuration for the master device, as shown in Figure 4.2 on page 4-1.

		216-2		Derdalt Value	onic	Description
	🔷 🖗 Baudrate	Enumeration of UDINT	9600	9600		Baudrate of the serial port.
itus	🖤 < Parity	Enumeration of STRING	'EVEN'	'EVEN'		Parity for messages on the serial port.
ormation	DataBits	USINT	8	8		Number of data bits
ormation	StopBits	USINT	1	1		Number of stop bits
	🖳 🖗 ComPort	USINT(11)	1	1		COM port number to use for the serial communication

Figure 4.2: Initial settings for the master device.



ATTENTION!

Only use **ComPort** 1 for the PLC500; using another port will result in a communication error.

Double-click on **Modbus_Master_COM_Port** and make sure that the **Auto-restart communication** option is enabled, as shown in Figure 4.3 on page 4-2.

General	Modbus RTU/ASCII			
ModbusGenericSerialMaster I/O Mapping	Transmission mode	O RTU	◯ ASCII	ng n
ModbusGenericSerialMaster IEC	Response timeout (ms)	1000		•••
Objects	Time between frames (ms)	10		
Modbus Generic Serial Master Parameters	Auto-restart communication	on		
Status				
Information				

Figure 4.3: Option to automatically restore Modbus/RTU communication.

Double-clicking on **Modbus_Server_COM_Port** sets the configuration for the slave device, as shown in Figure 4.4 on page 4-2. As an example, the slave address is set to 1, and the **Response timeout** is set to 1000 ms.

Modbus_Server_COM_Port	×		-
General	Modbus RTU/ASCII		
Modbus Server Channel	Server address [1247]	1	🐝 odbus
Modbus Server Init	Response timeout (ms)	1000	~
Modbus Generic Serial Server Parameters			
Modbus Generic Serial Server IEC Objects			
Status			
Information			

Figure 4.4: Settings for communication with the slave device.

4.3 MODBUS CHANNEL CONFIGURATION

Variables and memory addresses configured for a slave must be declared in the memory ranges reserved for transmission variables. The declaration of transmitted addresses is done in the **Modbus Server Channel** tab, as shown in Figure 4.5 on page 4-3.

Modbus_Server_COM_Port X									
General	Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length	Comment
Modbus Server Channel									
Modbus Server Init									
ModbusGenericSerialServer Parameters									
ModbusGenericSerialServer IEC Objects									
Status									
Information									
	Move Up	Mov	e Down			Add Chann	el De	lete	Edit

Figure 4.5: Tab to add Modbus/RTU network channels.

For each application, you can define the transmission addresses and which data is relevant. As an example, the PLC500 writes a value to the digital outputs and reads the value from the digital inputs of the PLC300 every 100 ms. In Figure 4.6 on page 4-3, the schematic representation of this network configuration is shown.



Figure 4.6: Schematic representation of Modbus channel configuration.

To add a new channel, click on **Add channel...**, after which the window shown in Figure 4.7 on page 4-3 appears. To write to the 9 digital outputs of the PLC300, configure channel 0 with the parameters shown in the figure.

and men	-			
Vame	Channel 0			
Access type	Write Multiple Coils (Fund	tion (Code 15)	~
Trigger	Cyclic	\sim	Cycle time (ms)	100
Comment	Digital Outputs PLC300			
READ Register				
Offset	0x0000			~
Length	0			
Error handling	Keep last value	\sim		
WRITE Register				
Offset	0x3E80			~
Length	9			

Figure 4.7: Channel 0 configuration example for Modbus/RTU network.

For reading the 10 digital inputs from the PLC300, configure channel 1 with the parameters shown in Figure 4.8 on page 4-4.

lodbus Channel		
Channel		
Name	Channel 1	
Access type	Read Discrete Inputs (Function Code 2)	\sim
Trigger	Cyclic \checkmark Cycle time (ms)	100
Comment	Digital Inputs PL300	
READ Register		
Offset	0x3E80	~
Length	10	
Error handling	Keep last value \checkmark	
WRITE Register		
Offset	0x0000	\sim
Length	0	

Figure 4.8: Channel 1 configuration example for Modbus/RTU network.



After these procedures, the channels are added in the **Modbus Server Channel** tab, as shown in Figure 4.9 on page 4-4.

Modbus_Server_COM_Port >	۲									
General		Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length	Comment
	0	Channel 0	Write Multiple Coils (Function Code 15)	Cyclic, t#100ms				16#3E80	9	Digital Outputs PLC300
Modbus Server Channel	1	Channel 1	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#3E80	10	Keep last value			Digital Inputs PLC300
Modbus Server Init										
ModbusGenericSerialServer Parameters										
ModbusGenericSerialServer I/O Mapping										
ModbusGenericSerialServer IEC Objects										
Status										
Information										
		Move Up	Move Down				Add	Channel	Delet	e Edit

Figure 4.9: Channels added in the example Modbus/RTU network.

The available function codes to access different types of data for the Modbus protocol in CODESYS are:

- Read Coils (code 1): reading a block of bits of the coil type.
- Read Discrete Inputs (code 2): reading a block of bits of the discrete input type.
- Read Holding Registers (code 3): reading a block of holding type registers.
- Read Input Registers (code 4): reading a block of input type registers.
- Write Single Coil (code 5): writing to a single bit of the coil type.
- Write Single Register (code 6): writing to a single holding register.
- Write Multiple Coils (code 15): writing to a block of bits of the coil type.

- Read Input Registers (code 16): writing to a block of holding type registers.
- Mask Write Registers (code 22): writing to a holding type register using a mask.
- **Read/Write Multiple Registers (code 23):** reading and writing to a block of holding type registers.
- Read Device Identification (code 43): identifying the device model.

NOTE!

The availability of Modbus addresses and the way data types can be accessed may vary from device to device. Therefore, during configuration, the **Product Manual** should be used to obtain information about the available addresses.

5 MONITORING

This chapter presents some methods for monitoring Modbus/RTU network variables and diagnosing communication errors in CODESYS.

5.1 MONITORING VARIABLES

Variables declared in the network can be monitored using two methods. The first option is to add them to the CODESYS program and monitor the variable values online. The second method only requires enabling **Always Update Variables** in the footer of the **ModbusGenericSerialServer I/O Mapping** tab, as indicated by the red arrow in Figure 5.1 on page 5-1.

Seneral	Find		Filter Show all		-	🕂 Add FB for IO C	hannel → Go to Instance
Modhua Saryar Channal	Variable	Mapping	Channel	Address	Туре	Default Value	Unit Description
Houbus Server Channel			Channel 0	%QB22	ARRAY [01] OF BYTE		Write Multiple Coils
Modbus Server Init	🗐 – 🍢		Channel 0[0]	%QB22	BYTE		Write Multiple Coils
	- V DO1_PLC30	D 🍾	Bit0	%QX22.0	BOOL	FALSE	0x3E80
ModbusGenericSerialServer	- ⁵ DO2_PLC30	D 🍾	Bit1	%QX22.1	BOOL	FALSE	0x3E81
		D 🍫	Bit2	%QX22.2	BOOL	FALSE	0x3E82
Apping	^K DO4_PLC30	D 🍾	Bit3	%QX22.3	BOOL	FALSE	0x3E83
ModbusGenericSerialServer IEC	- V DO5_PLC30	D 🍾	Bit4	%QX22.4	BOOL	FALSE	0x3E84
Dbjects	- ⁵ DO6_PLC30	D 🍫	Bit5	%QX22.5	BOOL	FALSE	0x3E85
	⁵ DO7_PLC30	D 🍫	Bit6	%QX22.6	BOOL	FALSE	0x3E86
status	- ⁵ DO8_PLC30	D 🍫	Bit7	%QX22.7	BOOL	FALSE	0x3E87
nformation	📄 🖻 🍢		Channel 0[1]	%QB23	BYTE		Write Multiple Coils
	\$ DO9_PLC30	D 🍫	Bit0	%QX23.0	BOOL	FALSE	0x3E88
	😑 🍫		Channel 1	%IB74	ARRAY [01] OF BYTE		Read Discrete Inputs
	🔍 🖹 🖹 🖗		Channel 1[0]	%IB74	BYTE		Read Discrete Inputs
	DI1_PLC300	**	Bit0	%IX74.0	BOOL		0x3E80
	- 👋 DI2_PLC300	**	Bit1	%IX74.1	BOOL		0x3E81
	DI3_PLC300	**	Bit2	%IX74.2	BOOL		0x3E82
	- 🏷 DI4_PLC300	***	Bit3	%IX74.3	BOOL		0x3E83
	* DI5_PLC300	**	Bit4	%IX74.4	BOOL		0x3E84
	- 👋 DI6_PLC300	**	Bit5	%IX74.5	BOOL		0x3E85
	DI7_PLC300	**	Bit6	%IX74.6	BOOL		0x3E86
	DI8_PLC300	***	Bit7	%IX74.7	BOOL		0x3E87
	🖹 🁋		Channel 1[1]	%IB75	BYTE		Read Discrete Inputs
	* DI9_PLC300	**	Bit0	%IX75.0	BOOL	Z	0x3E88
	🏷 DI10_PLC30	0 🍫	Bit1	%IX75.1	BOOL		0x3E89
			Rese	t Mapping	Always update variables	Use parent device s	etting
						Use parent device se	etting

Figure 5.1: Option to always update Modbus/RTU variable values with the bus cycle task.

Figure 5.2 on page 5-1 shows the CODESYS variable display screen for the Modbus/RTU slave used through online monitoring.

Device [connected] (PLC500)	General	Find		Filter Show all				 Add FB for IO Channel * Go to Instance 			
PLC Logic	Modbus Server Channel	Variable		Ma	Channel Channel 0	Address %0822	Type ARRAY (0., 1) OF BYTE	Default Value	Current Value Only subelements updated	Prepared Value Un	t Description Write Multiple Coil
· (11) Ubrary Manager IPIC_PRG (PRG) 응 25 Task Configuration IPIC 중 MainTask	Modbus Server Init	6.5	¢		Channel 0[0]	%Q822	BYTE		181		Write Multiple Coll
			- * DO1_PLC300	*	Bit0	%QX22.0	BOOL	FALSE	TRUE		0x3E80
	ModbusGenericSerialServer Parameters		- ** DO2_PLC300	*	Bit1	%QX22.1	BOOL	FALSE	FALSE		0x3E81
			• DO3_PLC300	*	Bit2	%QX22.2	BOOL	FALSE	TRUE		0x3E82
PLC_PRG	Mapping		- V DO4_PLC300	*	Bit3	%QX22.3	BOOL	FALSE	FALSE		0x3E83
- 🖸 🛷 Setup (Setup)	Modbue Generic Serial Server IEC		• 005_PLC300	*	Bit4	%QX22.4	BOOL	FALSE	TRUE		0x3E84
🖸 💳 I_Os (I/Os)	Objects		- ** DO6_PLC300	*	BitS	%QX22.5	BOOL	FALSE	TRUE		0x3E85
* G III Expansions (Expansions)	Status		DO7_PLC300	*	Bit6	%QX22.6	BOOL	FALSE	FALSE		0x3E86
- 😏 🛐 ETH1 (ETH1)			DO8_PLC300	*	Bit7	%QX22.7	BOOL	FALSE	TRUE		0x3E87
- 😔 🛗 ETH2 (ETH2)	Information	8-5	2		Channel 0[1]	%Q823	BYTE		0		Write Multiple Co
🚱 🔟 CAN (CAN)			> DO9_PLC300	×.	Bit0	%QX23.0	BOOL	FALSE	FALSE		0x3E88
= 😳 🔟 RS485 (RS485)		8-**			Channel 1	%IB74	ARRAY [01] OF BYTE		Only subelements updated		Read Discrete In
B- 😏 👔 Modbus_Master_COM_Port (Modbus Master, COM Port)		🗎 🗐 🖓	>		Channel 1[0]	%IB74	BYTE		167		Read Discrete In
- 🔂 🚮 Modbus_Server_COM_Port (Modbus Server, COM Port)			- * DI1_PLC300	*	BitO	%EX74.0	BOOL		TRUE		0x3E80
			DI2_PLC300	×	Bit1	%IX74.1	BOOL		TRUE		0x3E81
			* DI3_PLC300	×	Bit2	%IX74.2	BOOL		TRUE		0x3E82
			- * DI4 PLC300	*	Bit3	%IX74.3	BOOL		FALSE		0x3E83
			- * DI5_PLC300	*	Bit4	%IX74.4	BOOL		FALSE		0x3E84
			- * DI6_PLC300	*	BitS	%EX74.5	BOOL		TRUE		0x3E85
			* DI7_PLC300	×	Bit6	%IX74.6	BOOL		FALSE		0x3E86
			* DI8_PLC300	*	Bit7	%IX74.7	BOOL		TRUE		0x3E87
		8-4	2		Channel 1[1]	%1875	BYTE		0		Read Discrete In
			- 🍫 DI9_PLC300	*	Bit0	%EX75.0	BOOL		FALSE		0x3E88
			* DI 10_PLC300	*	Bit1	%EX75.1	BOOL		FALSE		0x3E89

Figure 5.2: Monitoring Modbus/RTU variables in online mode in CODESYS.

5.2 COMMUNICATION ERRORS

The monitoring of network states can be done in **Devices**, indicating the state of each communication step and reporting the status (**Status**). When connection issues are encountered, as shown in Figure 5.3 on page 5-2, check if the cables are properly connected, the state of the SERIAL LED, and then review the configurations.



Figure 5.3: Communication error indication.

In online mode, access the **Status** tab of the **Modbus_Server_COM_Port** as shown in Figure 5.4 on page 5-2. In this location, CODESYS will inform which issue is preventing communication. In this case, the cable was not properly connected to the slave device.

Modbus_Server_COM_Port	ĸ			
General	ModbusGenericSerialS	:	Not running The error has been deared.	
Modbus Server Channel	Last diagnostic message			Acknowledge
Modbus Server Init	Server Diag Request Counter	1352	Server's diagnostic information	
ModbusGenericSerialServer Parameters	Error Counter	57		
ModbusGenericSerialServer I/O Mapping	Timestamp	DT#2024-2-27-1:19:53		
ModbusGenericSerialServer IEC Objects	Error Code	U RESPONSE TIMEOUT	There was no response in time	
Status				
Information				

Figure 5.4: Communication error indication.

After reconnecting the cable and waiting for a moment, network communication is restored, and the device icons turn green. A warning icon is shown on the device, indicating that an error was declared, as shown in Figure 5.5 on page 5-2.



Figure 5.5: Indication of a declared error in the Modbus/RTU network.

To remove the warning, simply click **Acknowledge**, as indicated in Figure 5.6 on page 5-3.

neral	ModbusGenericSerialS	:	Running The error has been dea	red.
dbus Server Channel	Last diagnostic message			Acknowledge
dbus Server Init	Server Diag	0505	Server's diagnostic information	\frown
dbusGenericSerialServer rameters	Error Counter	91		T [
dbusGenericSerialServer I/O pping	Timestamp	DT#2024-2-27-1:43:8		
dbusGenericSerialServer IEC jects	Channel Index Error Code	1 RESPONSE TIMEOUT	There was no response in time	
itus				
ormation				

Figure 5.6: Acknowledging the declared error in the Modbus/RTU network.

After acknowledging the warning and confirming the proper functioning of the network, the device icons will be displayed as shown in Figure 5.7 on page 5-3.



Figure 5.7: Modbus/RTU network running without errors or warnings between master and slave.

NOTE!

It is recommended to consult the documentation available in the **Library Manager**, or access the online help at CODESYS Online Help, for more information on configuring the Modbus/RTU network and available functions.



BRAZIL WEG DRIVES & CONTROLS - AUTOMAÇÃO LTDA. Av. Prefeito Waldemar Grubba, 3000 89256-900 - Jaraguá do Sul - SC Phone: 55 (47) 3276-4000 Fax: 55 (47) 3276-4060 www.weg.net/br