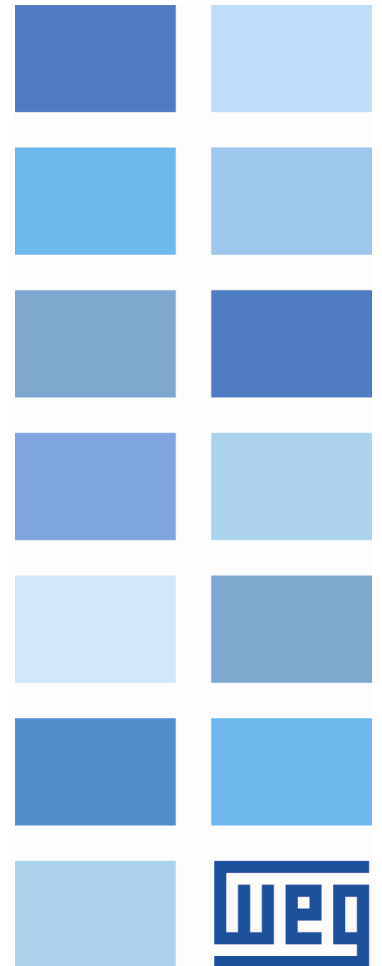


CANopen - Network Master

PLC500

Application Note





Master CANopen - Application Note

Series: PLC500

Language: English

Document: 10010165104 / 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 CANOPEN NETWORK COMPONENTS	5
1.2 CANOPEN INTERFACE	5
1.3 NETWORK ARCHITECTURE	5
2 MASTER CANOPEN CONFIGURATION	7
2.1 STARTING THE PROJECT IN CODESYS	7
3 CANOPEN SETTINGS	10
3.1 NETWORK SETTINGS	10
3.2 I/O SETTINGS	11
3.3 DECLARATION OF ADDRESSES	11
3.4 TERMINATION RESISTORS	12
4 MONITORING	13
4.1 VARIABLES MONITORING	13
4.2 COMMUNICATION ERRORS	13
4.3 CAN INDICATION LED	14

1 INTRODUCTION

This application manual describes how to operate the PLC500 as a network master. CANopen. For the PLC500 programmable controller settings, it is recommended to follow the steps contained in this document to configure network CANopen properly.

ATTENTION!

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

1.1 CANOPEN 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 CANopen network.

For a deeper and detailed description of the CANopen network operation and its settings, access the online help of Codesys at: <https://help.codesys.com>.

1.2 CANOPEN INTERFACE

Figure 1.1 shows the PLC500 with interface CANopen highlighted. The indication of the interface CANopen pins is described in Table 1.1.

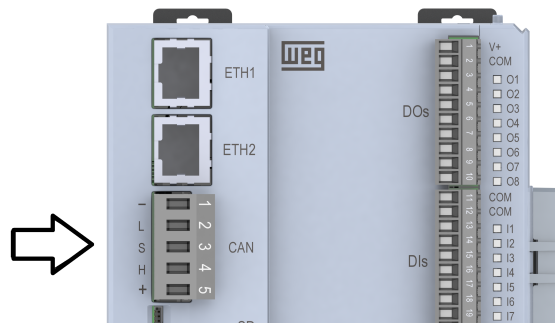


Figure 1.1: Indication of the CAN connector pins on the PLC500.

Pin	Name	Lettering	Function
1	COM	-	Common of the CAN network (connected to the negative pole of the network CANopen)
2	CAN_L	L	CAN_L communication signal
3	SHIELD	S	Cable Shield
4	CAN_H	H	CAN_H communication signal
5	NC	+	Not connected (it can receive the positive pole of network CANopen)

Table 1.1: Interface CANopen description.

1.3 NETWORK ARCHITECTURE

Figure 1.2 shows the components and architecture of network CANopen. This example shows the connection of a PLC500 (network master) and a RUW100 (network slave). The computer programs the devices, with CODESYS in the PLC500 and the WPS for the RUW100.

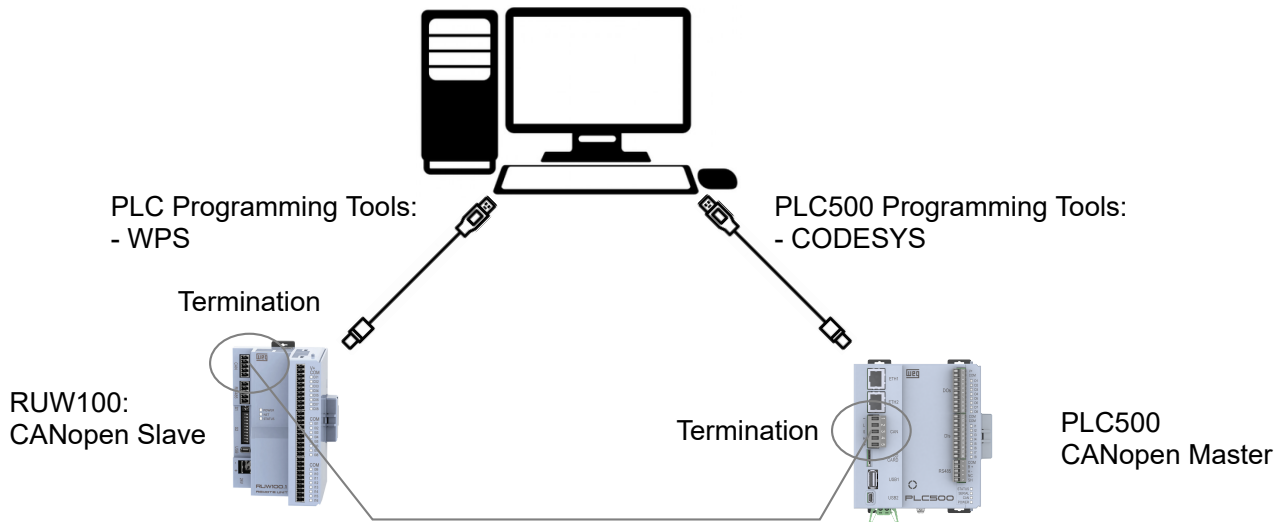


Figure 1.2: Network components and architecture CANopen.



NOTE!

This application manual is about the PLC500 and the **Codesys** programming tool—we recommend using **Codesys V3.5 SP18 or higher**. If you need more information about the CANopen communication protocol, refer to its manuals.

2 MASTER CANOPEN CONFIGURATION

2.1 STARTING THE PROJECT IN CODESYS

For a network CANopen settings and definitions, you must first create the project and include the PLC500 programmable controller. In the Codesys software, create a new project and choose the directory and name for the application. Then, select the PLC500-WEG device and the desired programming language, as shown in Figure 2.1.

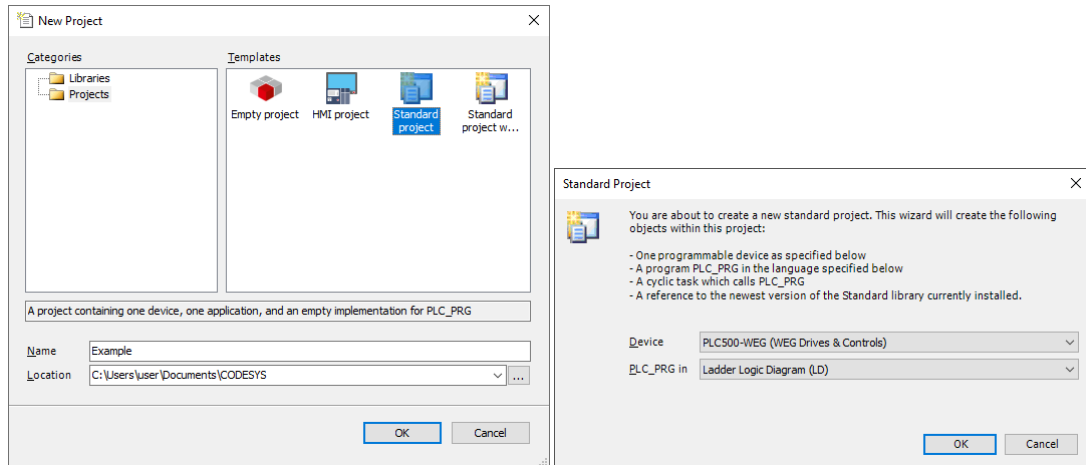


Figure 2.1: Project configuration in Codesys.



NOTE!

In case device PLC500 is not available in the Codesys 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 network interfaces already preset, as shown in Figure 2.2.

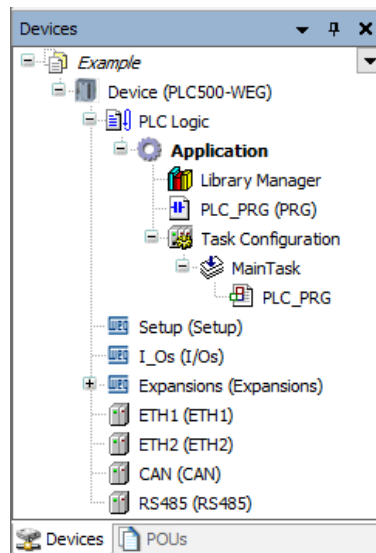


Figure 2.2: PLC500 interfaces.

In the next step, add the device **CANopen_Manager**, as shown in Fig. 2.3.

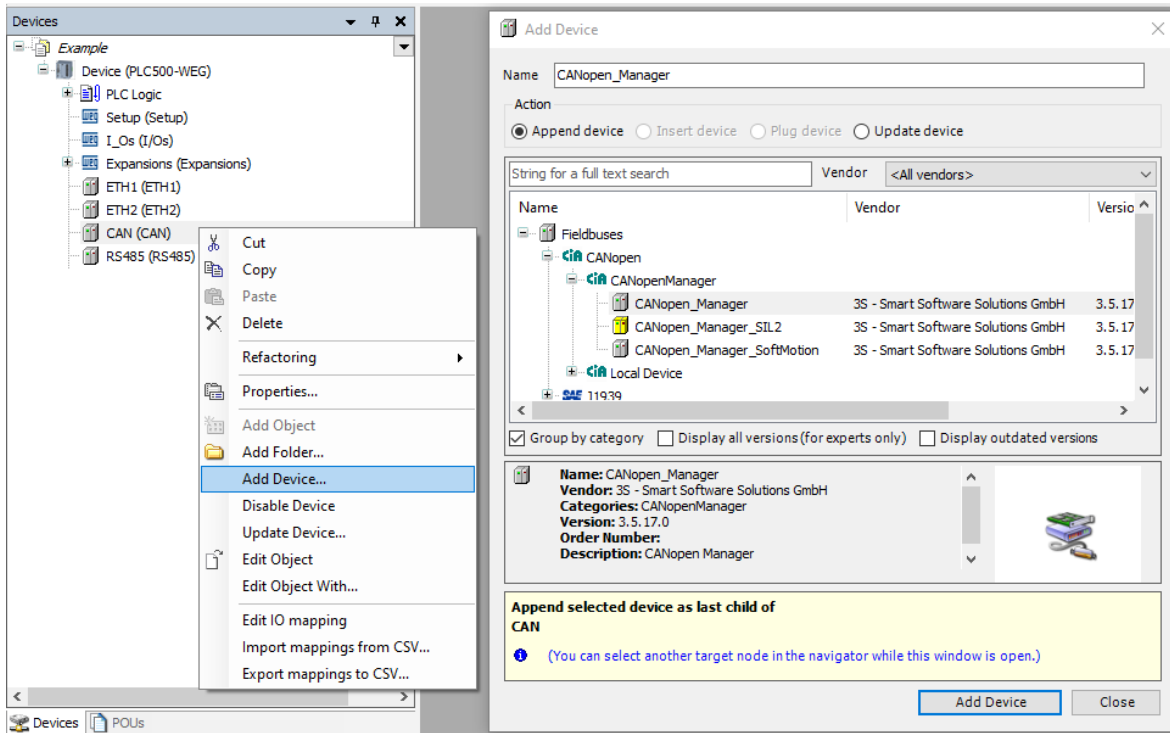


Figure 2.3: CANopen settings Step 1.

In **CANopen_Manager**, add the network slave device CANopen. Figure 2.4 shows the steps described above for this configuration. In this case, RUW100 is selected as an example of a network slave.

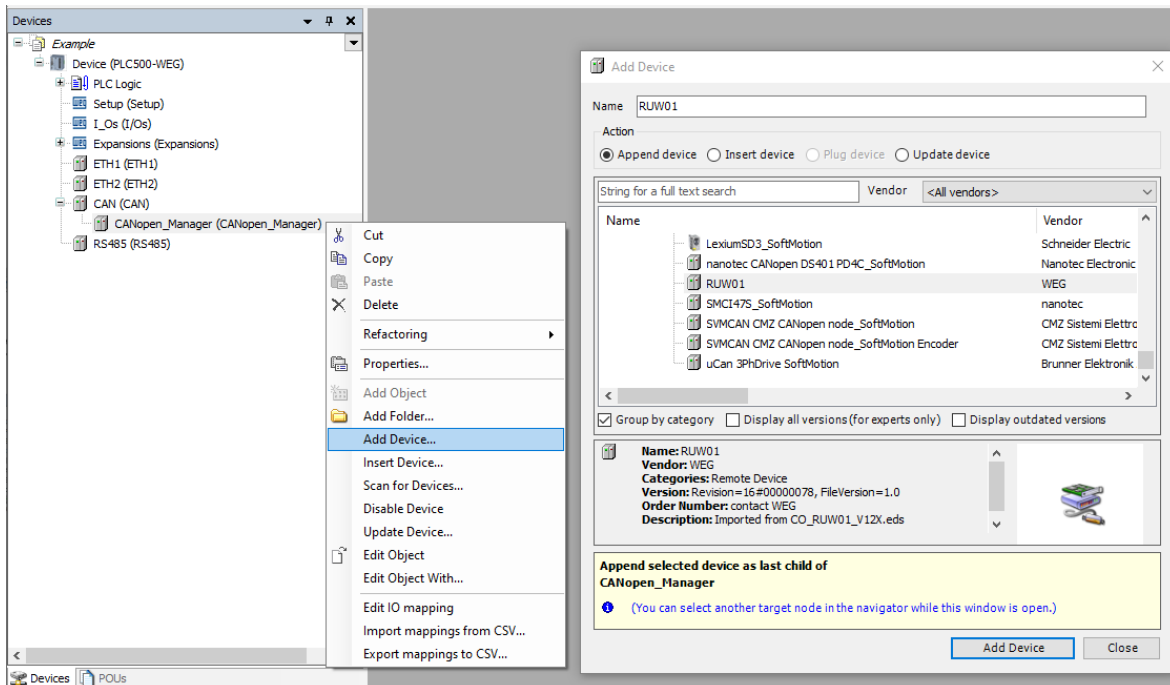


Figure 2.4: CANopen settings Step 2.

The first time some equipment is added as a network slave device CANopen, you must obtain the respective EDS file of the product and import it to Codesys. Install the EDS file in Codesys at: **Tools -> Device Repository -> Install**, and select the desired EDS file. All features can be found on the product page at <https://www.weg.net>.


NOTE!

Electronic Device Description (EDS) is the file needed to describe the slave device to the network CANopen master. The manufacturer must provide this file to allow the correct configuration of this device when the network slave option is available.

At this moment, the **CAN** interface should have the items indicated in Figure 2.5.

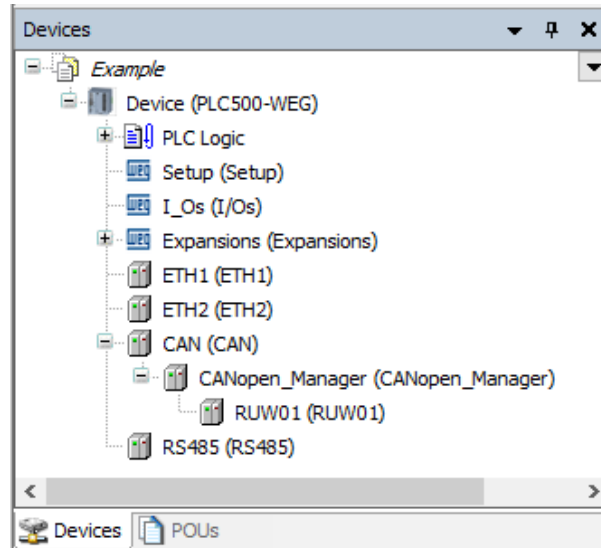


Figure 2.5: CAN interface working as master CANopen.

3 CANOPEN SETTINGS

3.1 NETWORK SETTINGS

The network must be configured on the master to serve the addresses defined on the network slaves. To that end, set the parameters and addresses that will be transmitted from the network master. Each slave device must be configured with a unique address and at the same baud rate; otherwise, the network will experience problems. In figure 3.1, the master device settings are configured.

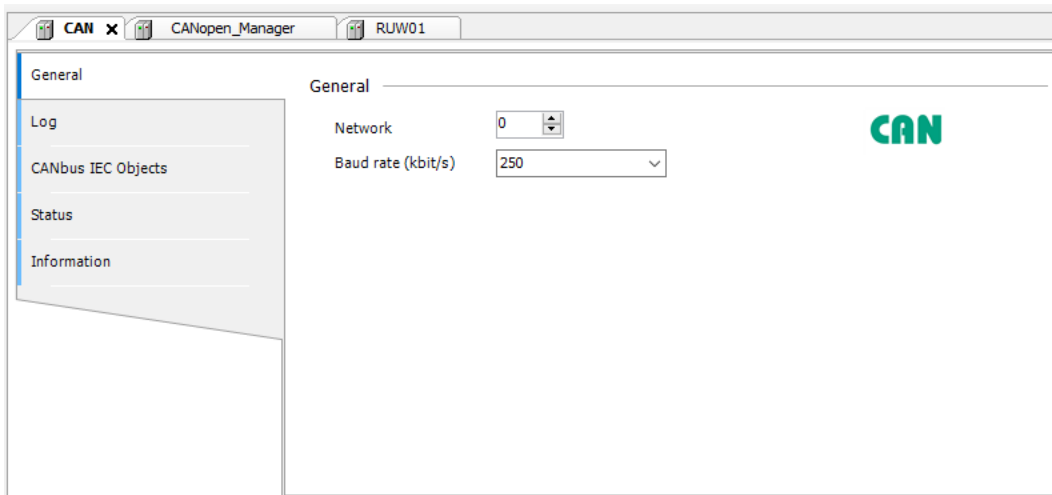


Figure 3.1: Master CANopenSettings.

The network settings for the slave device are shown in Figure 3.2. Each slave must also have its own address and be within the range of addresses available to slaves. Thus, the settings to start communication between master and slave devices are completed.

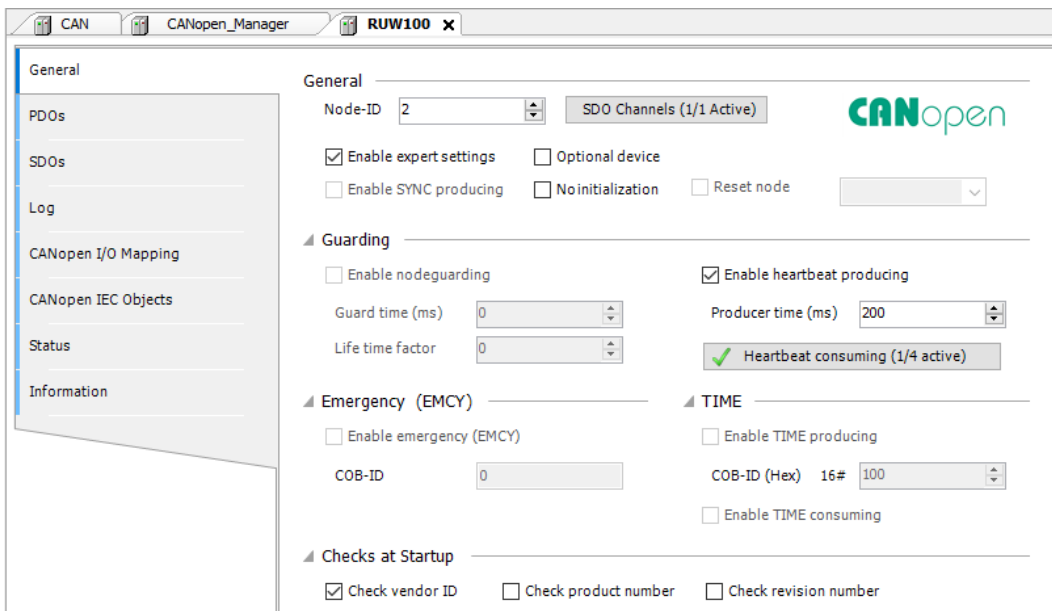


Figure 3.2: CANopen settings of the slave device.

As this example uses the RUW100 device, the address and rate are directly set on the DIP switches outside the product.



NOTE!

The configuration method may be different for each device, depending on the product type and programming software. Set the slave device according to the manual provided by the manufacturer.

3.2 I/O SETTINGS

Variables and addresses set for a slave must be programmed taking into account the master of that network CANopen. As shown in Figure 3.3, after importing the network CANopen slave into Codesys, the PDO set (Process Data Object) will be available according to each device.

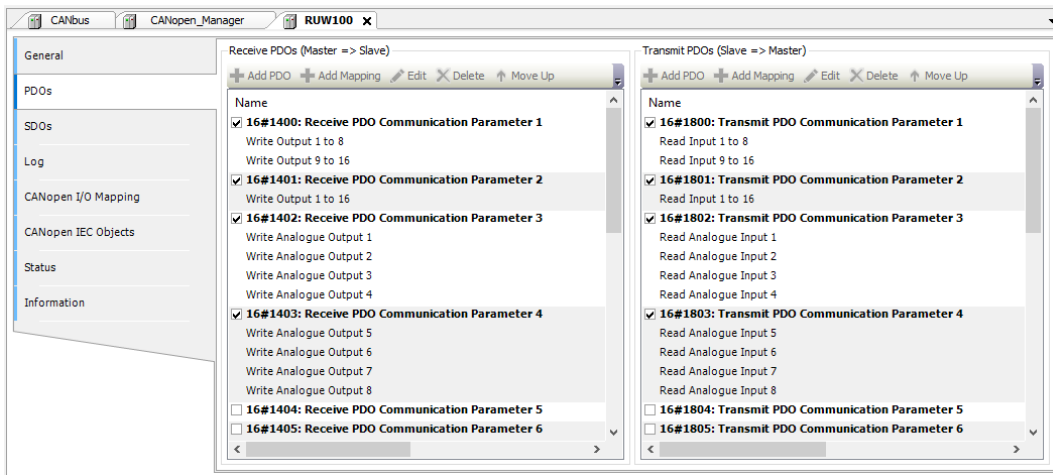


Figure 3.3: List of PDOs available on the slave device.

For the RUW100 device, we have 32 PDOs for sending and 32 PDOs for receiving data, which can be individually configured to transmit up to 8 bytes of data. These items are described in the RUW100 **User Manual**.



NOTE!

For each application, there may be specific settings and addresses. Thus, the manufacturer must inform the availability of PDOs according to the device. It is recommended to check the specific product manual for more details.

3.3 DECLARATION OF ADDRESSES

With the transmission addresses, you can define which data are relevant for the application. In the example of Figure 3.4, we write some value to the outputs and perform a reading right away.

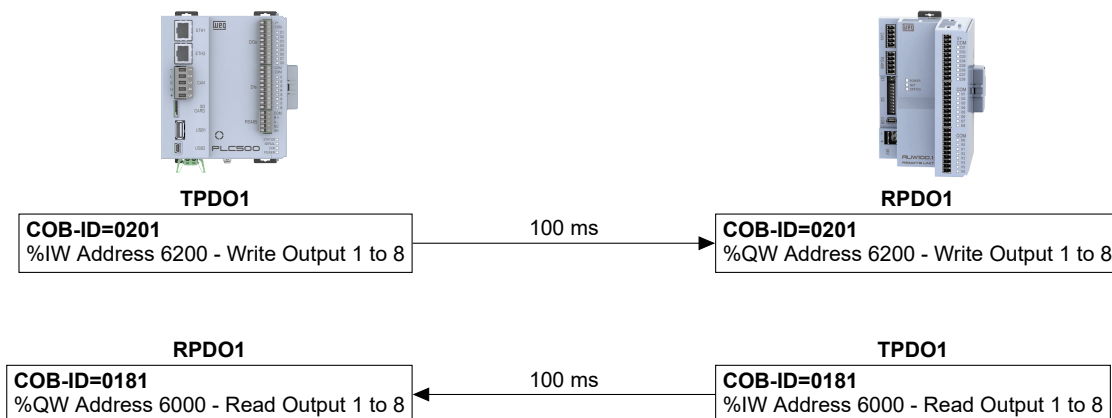


Figure 3.4: PDO settings.

3.4 TERMINATION RESISTORS

Network CANopen must have resistors installed at its terminations, as they are essential for the correct operation. The configuration of this item is done only through **Codesys**. For the PLC500, go to **setup** and then **Setup Parameters**, where the **Termination Resistors** fields are, as shown in Fig. 3.5.

Parameter	Type	Value	Default Value	Unit	Description
Firmware					
Firmware version	STRING	'Not connected'	'Not connected'		PLC's firmware version
Update available	STRING	'Not connected'	'Not connected'		Firmware version available in PLC for update
Update	Enumeration of BYTE	No	No		Select "Yes" to update PLC's firmware with the latest version
Date and Time					
Date	STRING	'2021-12-31'	'2021-12-31'		Read PLC date value, use the format '2021-12-31'
Time	STRING	'00:00:00'	'00:00:00'		Read PLC time value, use the format '12:59:59'
Termination Resistors					
RS485	Enumeration of BYTE	Not Connected	Not Connected		RS485 termination resistor configuration
CAN	Enumeration of BYTE	Not Connected	Not Connected		CAN termination resistor configuration

Figure 3.5: Configuration page for terminating resistors.

4 MONITORING

4.1 VARIABLES MONITORING

Figure 4.1 shows the Codesys variables view screen for the used network slave.

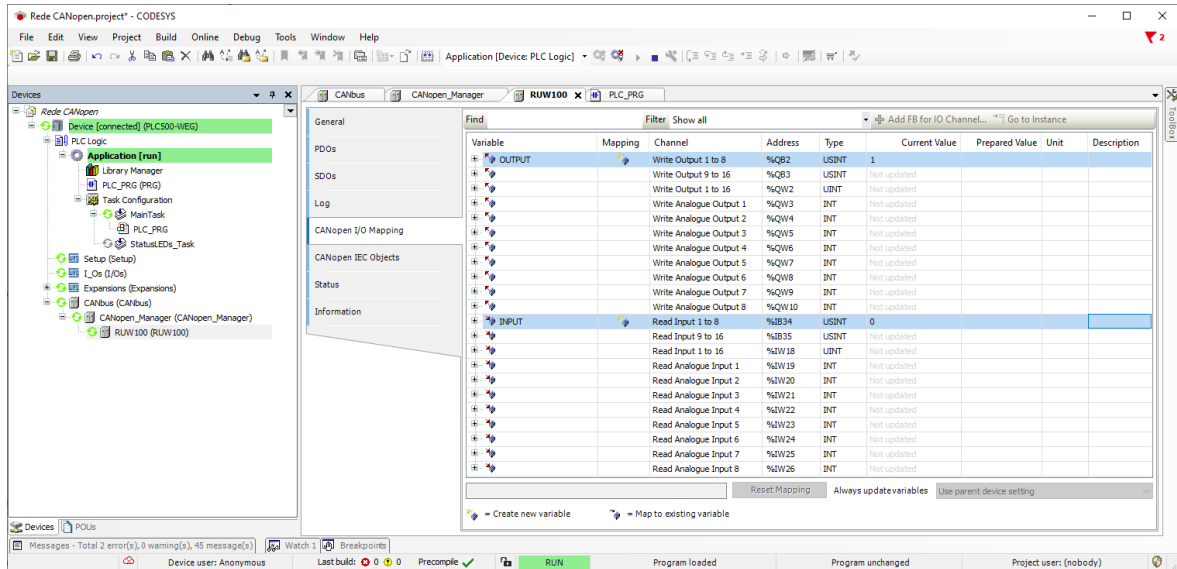


Figure 4.1: Monitoring variables in Codesys.

The variables declared in the network can be monitored using two methods: first, adding the variables to the Codesys program and monitoring the variable values online; second, enabling **Always Update Variables** at the bottom of the previous page, as indicated by the red arrow in Figure 4.2.

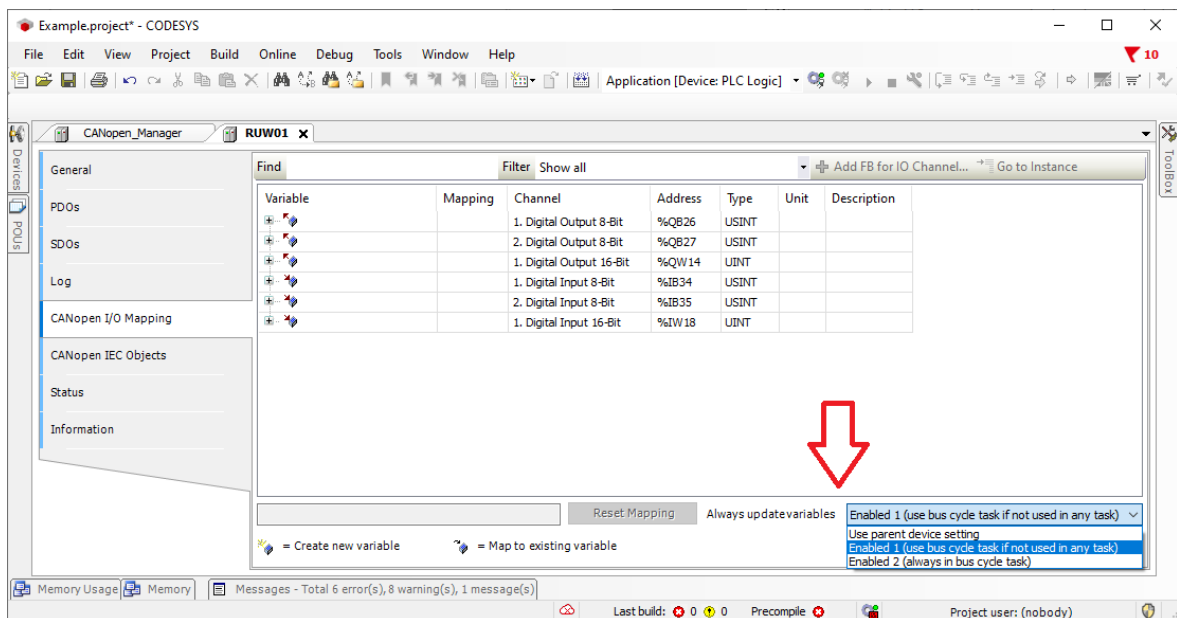


Figure 4.2: Monitoring variables in Codesys.

4.2 COMMUNICATION ERRORS

The network status can be monitored in **Devices**, which indicates the state of each communication step and reports the **(Status)**. When connection problems occur, as shown in Figure 4.3, check that the cables are properly connected, the status of the **CAN LED** at PLC500, and then review the settings.

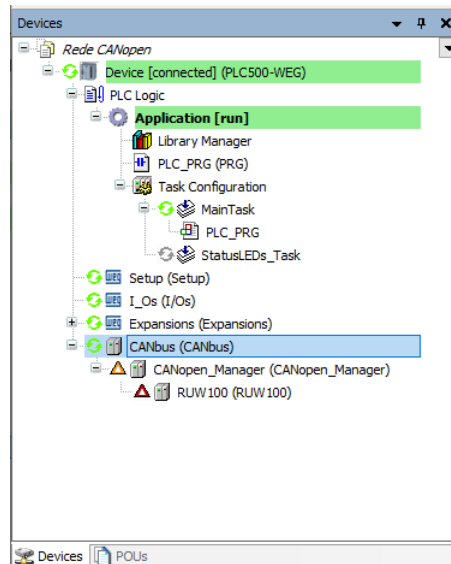


Figure 4.3: Indication of error in the communication with the slave device.

Connected to PLC500, access the **Status** and **Log** tabs, shown in Figure 4.4. There, Codesys will inform you which problem it is encountering to perform the communication. When the settings are correct, and the devices identify each other, we should check all other items in green.

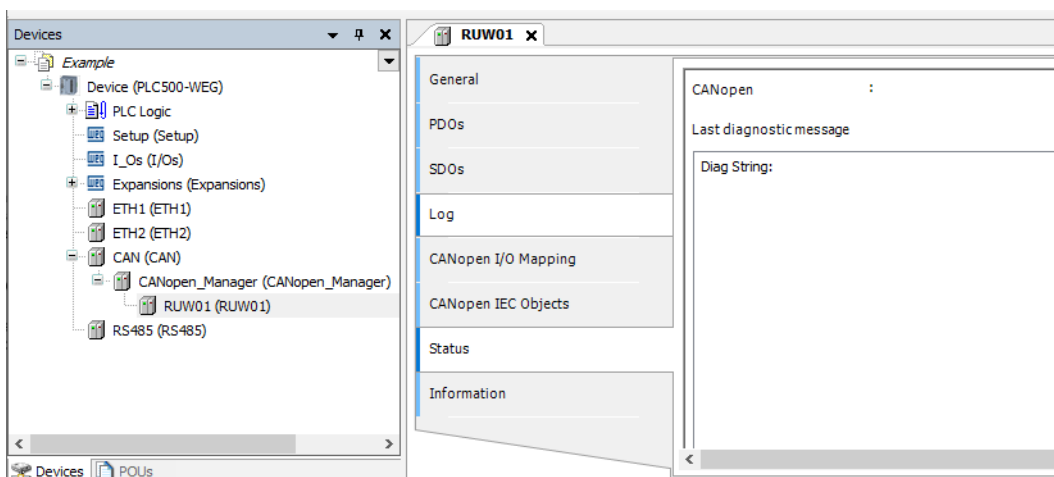


Figure 4.4: Status and Log pages contained in each of the items of the communication interfaces.

4.3 CAN INDICATION LED

The PLC500 programmable logic controller has LEDs on the front of the product that are used to indicate the interfaces. The LED color indication shown in Figure 4.5 is described in Table 4.1.

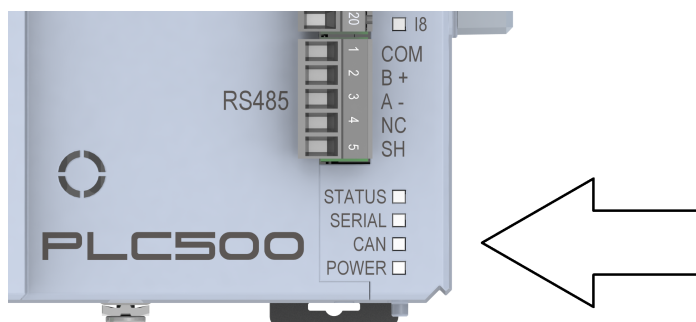


Figure 4.5: LEDs indicadores do PLC500.

Table 4.1: Operation of the CAN interface indication LED.

LED CAN	STATUS	DESCRIPTION
OFF	No settings.	There are no communication settings CANopen in the application.
GREEN	Communication CANopen without fault.	Communication CANopen configured in the application and all elements running without error/fault.
ORANGE	Communication CANopen with partial fault.	Communication CANopen configured in the application and with some elements containing error/fault.
RED	Communication CANopen with total fault.	Communication CANopen configured in the application and all elements running without error/fault.



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