

Operating CFW900 in a CANopen network using PLC300

CFW900

Application Note

CFW900-CCAN-W CANopen Application Note

CFW900

Document: 10011592585

Revision: 00

Publication Date: 11/2023

SUMMARY OF REVISIONS

The information below describes the reviews made in this manual.

Version	Revision	Description
-	R00	First Edition.

CONTENTS

1	ABOUT THIS DOCUMENT	1-1
1.1	REFERENCED DOCUMENTS	1-1
1.2	DISCLAIMER	1-1
1.3	TRADEMARKS	1-1
1.4	ARCHITECTURE	1-2
1.5	CFW900	1-2
1.6	PLC300	1-2
1.7	PASSIVE NETWORK COMPONENTS	1-2
2	SLAVE CONFIGURATION	2-1
2.1	CANOPEN INTERFACE	2-1
2.2	LOCAL/REMOTE	2-1
2.3	COMMUNICATION ERROR	2-2
3	CANOPEN NETWORK CONFIGURATION	3-1
3.1	CREATE WPS RESOURCE FOR PLC300	3-1
3.2	IMPORT EDS FILE	3-1
3.3	ADD CFW900 TO THE NETWORK CONFIGURATION	3-1
3.4	CONFIGURE SLAVE PDOS	3-2
3.5	CONFIGURE MASTER PDOS	3-4
3.6	CONFIGURE ERROR CONTROL	3-6
3.7	DOWNLOAD CONFIGURATION	3-7
4	CONTROL AND MONITORING	4-1
4.1	VIEW AND EDIT CYCLIC DATA	4-1
4.2	LADDER LOGIC FOR ACYCLIC DATA TRANSFER	4-4

1 ABOUT THIS DOCUMENT

The purpose of this application note is to provide practical instructions on configuring CFW900 frequency converter with accessory CFW900-CCAN-W for operation in a CANopen network using the PLC300.

This document is meant for trained personnel working with the described equipment and CANopen networks, and requires basic knowledge of automation and programmable logic controllers, in particular about WPS software.

1.1 REFERENCED DOCUMENTS

This application note was developed based on the following documents and tools:

Document	Version	Source
CFW900 User's Manual	10008985516 / 03	WEG
CFW900 Programming Manual	10008985492 / 05 (1.08.XX)	WEG
CFW900 CANopen User's Guide	10009145043 / 05 (1.08.XX)	WEG
WPS	3.00	WEG
Planning and Installation Manual - DeviceNet Cable System	PUB00027R1	ODVA

1.2 DISCLAIMER

The information in this application note is given as a hint and orientation for the use of WEG Automation products. WEG Automation does not assume any liability arising out of the application or use of any product, product function or instructions describe herein, neither does not convey any license under its patents rights, nor the rights for the others.

WEG reserves the right to make changes without further notice to any product and information herein to improve reliability, function or design.

1.3 TRADEMARKS

CANopen[®] and CiA[®] are registered community trademarks of CAN in Automation.

Windows[®] is a registered trademark of Microsoft Corporation.

1.4 ARCHITECTURE

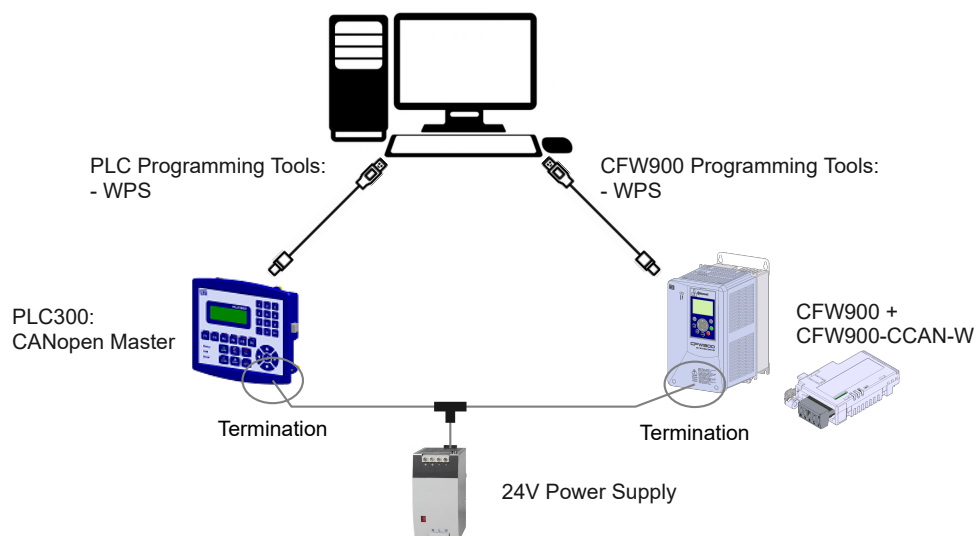


Figure 1.1: Network Components for this Application

1.5 CFW900

- Equipment: CFW900 with CFW900-CCAN-W accessory installed in slot A
- Version: 1.08.01
- EDS file: CO_CFW900_V108XX.eds
- Programming tools:
 - WPS version 3.00

1.6 PLC300

- CPU: PLC300HP version 4.14
- Programming tools:
 - WPS version 3.00

1.7 PASSIVE NETWORK COMPONENTS

For passive components, such as cables, we recommend using certified products for DeviceNet networks. Please refer to the product documentation for information about the proper network installation.

2 SLAVE CONFIGURATION

This section describes the main configurations for operating CFW900 frequency converter in a CANopen network. Some of the described configurations are only available if CFW900-CCAN-W accessory is properly installed.

Refer to the CFW900 programming manual for the necessary configurations related to other functions, such as motor configuration, protections, etc.

2.1 CANOPEN INTERFACE

For frequency converter CFW900 operation in the CANopen network, it is necessary to program the protocol, address (or Node-ID), and baud rate. For this application, the following configurations have been made:

- C9.8.1 CAN/CANopen/DNet Protocol: CANopen. ❶
- C9.8.2 CAN/CANopen/DNet Address: 2. ❷
- C9.8.3 CAN/CANopen/DNet Baud Rate: 500 Kbps. ❸

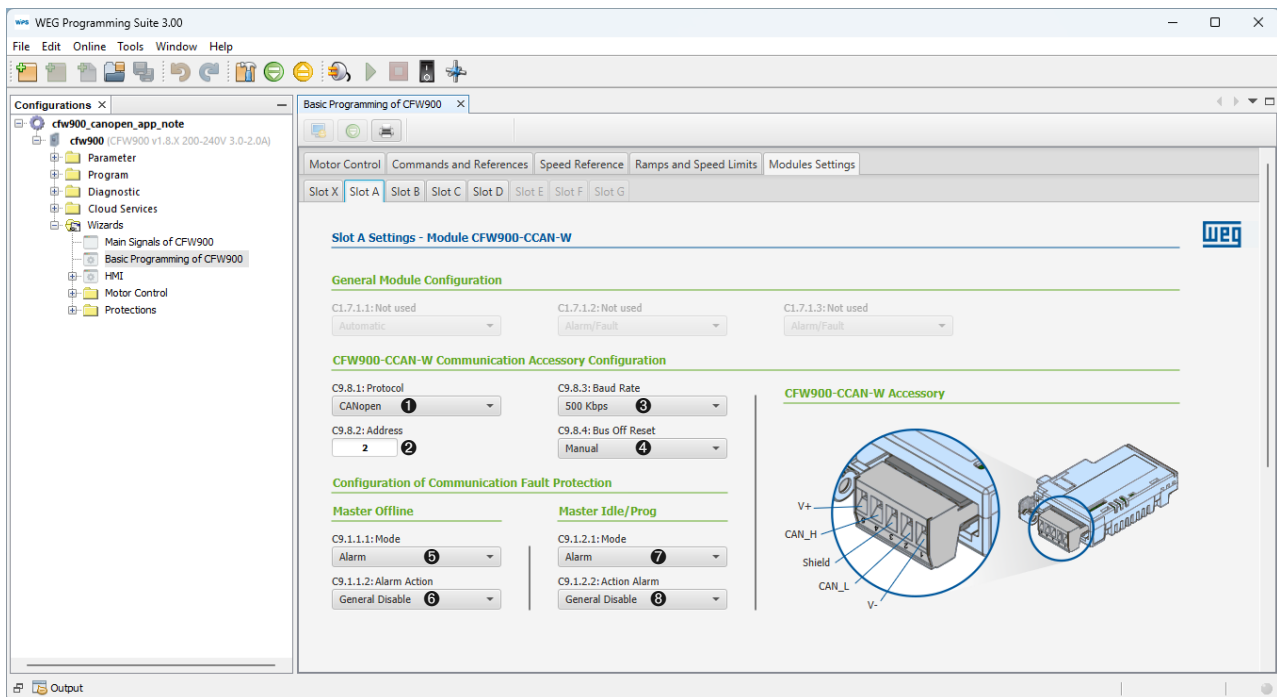


Figure 2.1: WPS - CAN Accessory Configuration

2.2 LOCAL/REMOTE

CFW900 has two operating modes: Local and Remote 1/Remote 2. For each operating mode, it is necessary to define the source that it will use to receive commands, like start/stop, direction of rotation and speed reference. For this application, considering CFW900-CCAN-W accessory installed in slot A, the following control sources have been defined:

- Local: keypad will control CFW900 in local mode.
- Remote: slot A (CFW900-CCAN-W) will control CFW900 in remote mode.
- Remote 1/Remote 2 transition: the definition if the device is in Remote 1 or Remote 2 will also be controlled by slot A commands.

Based on this, the following configuration have been programmed for remote operating mode:

SLAVE CONFIGURATION

- C4.1.1 LOC/REM Mode Config. Command mode: Remote 1. ❶

In this case, the change between Remote 1 and Remote 2 modes is done by the CANopen network. Hence, the following settings have been made for Remote 1 operating mode:

- C4.2.1.1 R1 Command Config. General Enable: CAN/CO/DN. ❷
- C4.2.1.2 R1 Command Config. Run/Stop: CAN/CO/DN. ❸
- C4.2.1.3 R1 Command Config. Direction of Rotation: CAN/CO/DN. ❹
- C4.2.1.4 R1 Command Config. JOG: CAN/CO/DN. ❺

Furthermore, the speed reference source is also defined by the CANopen network:

- C4.3.1.2.1 Speed Ref. Source Remote 1 Mode: CAN/CO/DN. ❻

Remote 2 will not be used in this application, therefore, the configurations for this mode have been programmed as not used. ❶ ❷ ❸ ❹ ❺

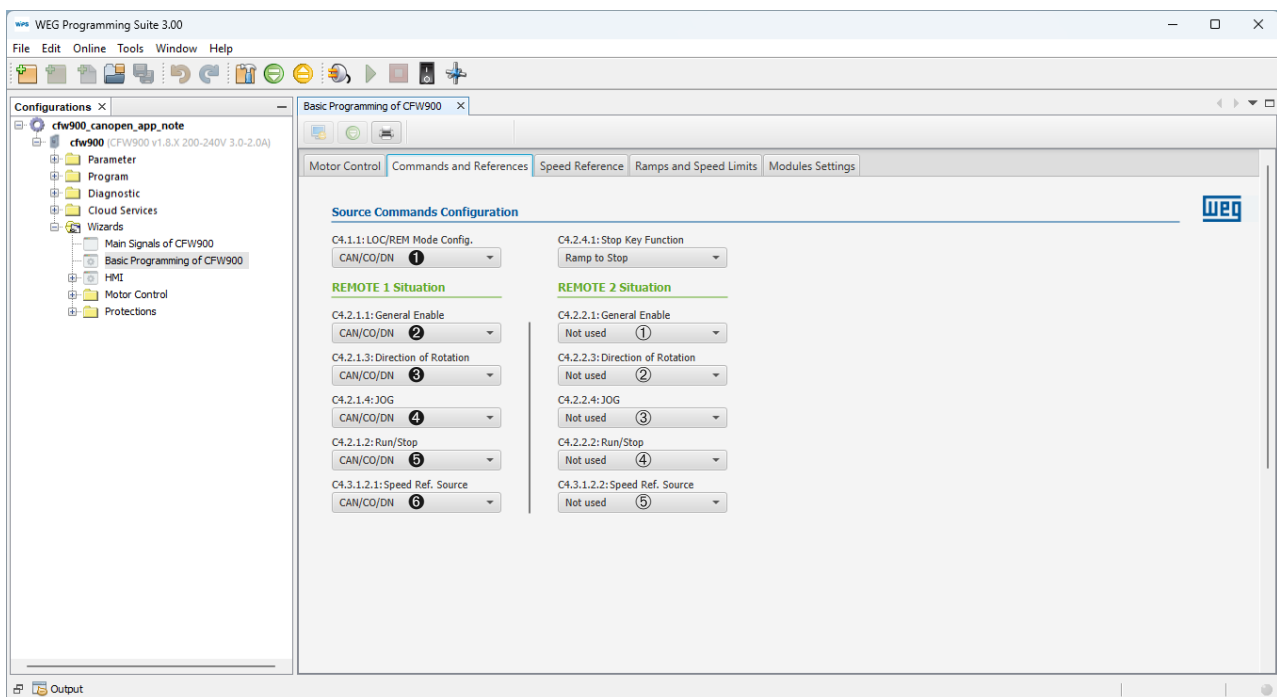


Figure 2.2: WPS - Local/Remote Configuration

2.3 COMMUNICATION ERROR

It is important to define the action CF900 must take in case of communication error. For this application, a communication error should lead to an alarm indication. If CF900 was running the motor via network command, CF900 should also perform a general disable.

Based on this, the following configurations have been programmed (refer to figure 2.1):

- C9.8.4 CAN/CANopen/DNet Bus Off Reset: Manual. ❹
- C9.1.1.1 Master Offline Mode: Alarm. ❺
- C9.1.1.2 Master Offline Alarm Action: General Disable. ❻
- C9.1.2.1 Master Idle/Prog Mode: Alarm. ❼
- C9.1.2.2 Master Idle/Prog Alarm Action: General Disable. ❸

3 CANOPEN NETWORK CONFIGURATION

For CANopen configuration, use WPS software. Based on this, the main steps are described below.

3.1 CREATE WPS RESOURCE FOR PLC300

Add a new resource for the PLC300 CANopen master to the WPS Configuration. Then select the CANopen folder and open WPSCAN by double-clicking the CANopen configuration file.

- Select “PLC300 (Rev. 4.10)” as the network master.
- Change baudrate to 500 kbit/s and click OK.

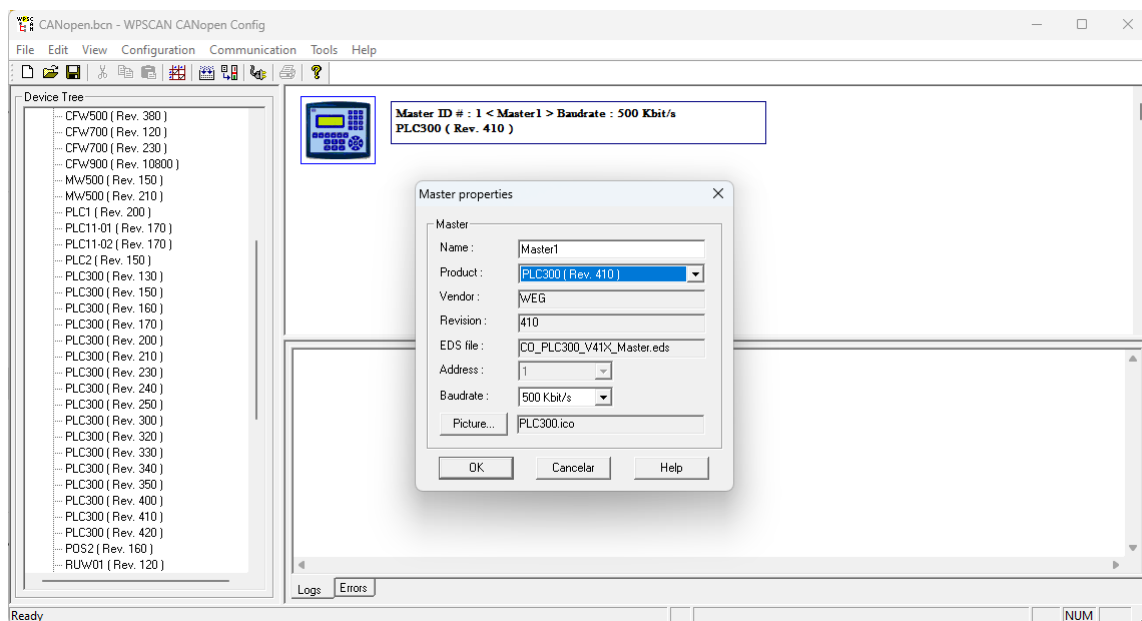


Figure 3.1: Add Master Device

3.2 IMPORT EDS FILE

Import the EDS file using the Tools menu. The EDS file must match CFW900 firmware version, otherwise communication will not be possible.

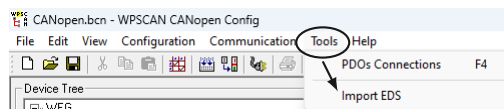


Figure 3.2: Register CFW900 EDS File

3.3 ADD CFW900 TO THE NETWORK CONFIGURATION

Add the slave device by dragging and dropping the “CFW900 (Rev. 10800)” available on the list of devices into the network structure area.

CANOPEN NETWORK CONFIGURATION

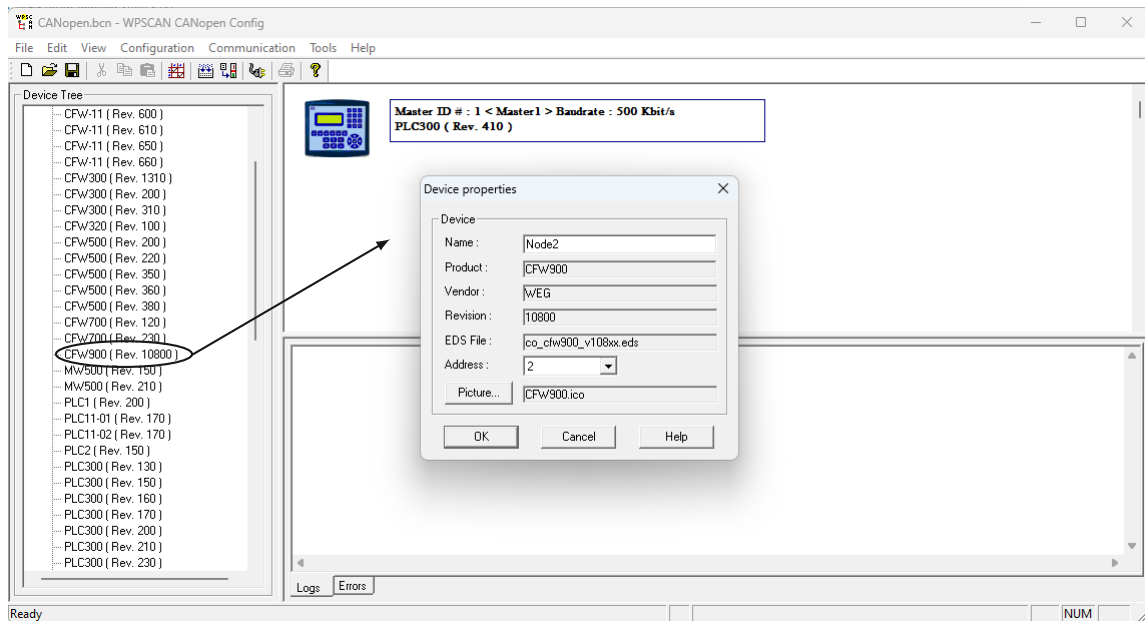


Figure 3.3: Add Slave Device

3.4 CONFIGURE SLAVE PDOS

The CFW900 has a set of PDOs where it is possible to define any device data to exchange within the network. There is an appendix at CANopen User's Guide describing the entire list of device data which can be programmed to PDOs.

Parameter	Description	Range of values	Decimal places	Index	Net Id	Size
S1 Status\Inverter						
S1.1	Status					
S1.1.1	Inverter	0 = Ready 1 = Run 2 = Undervoltage 3 = Fault 4 = Configuration 5 = STO 6 = Power Off 7 = Disabled 8 = SS1 9 = Self-Tuning 10 = Sleep		2006h	6	enum

Figure 3.4: List of Available Data Described at CFW900-CCAN-W CANopen User's Guide

For each application, it is necessary to look at this appendix and define which data to communicate. Considering CFW900-CCAN-W accessory installed in slot A, for this application, CFW900 will exchange the following data:

Mapped Inputs	Net Id	Size	Index
S5.1.1 Status and Commands Status Word 1	680	16bit	22A8h
S5.1.2 Status and Commands Speed	681	s16bit	22A9h

Mapped Outputs	Net Id	Size	Index
S5.7.2 CAN/CANopen/DNet Control Word	684	8bit	22ACh
S5.7.3 CAN/CANopen/DNet Speed Reference	685	s16bit	22ADh

After choosing the data, select the PDOs indicated below for a transfer of data every 100 ms.

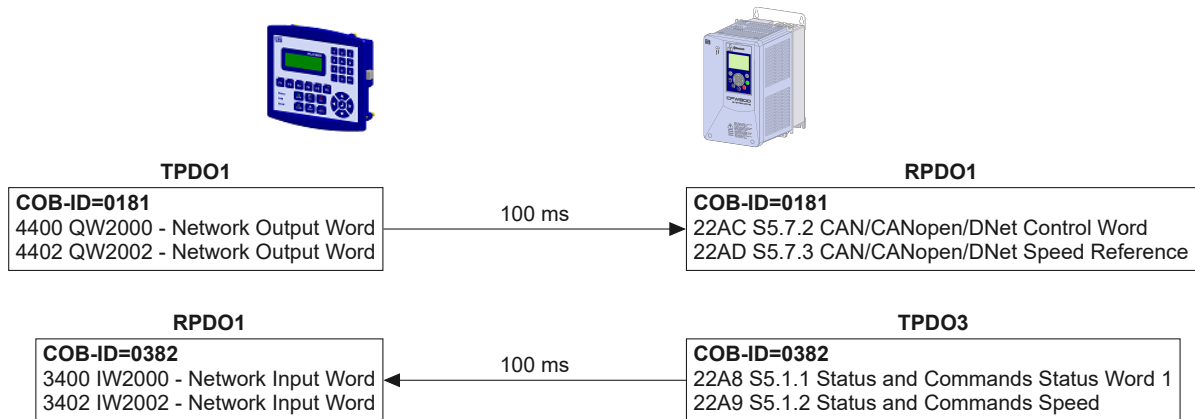


Figure 3.5: PDO Configuration

Based on this data, the following configurations have been programmed into the slave node:

Receive PDO configuration:

- In the PDOs configuration window, set 181h as the COB-ID for Receive PDO 1. This is the master's Transmit PDO 1 COB-ID. ❶
- The RPDO1 default configuration for the mapped objects is: ❷
 - "22AC S5.7.2 CAN/CANopen/DNet Control Word".
 - "22AD S5.7.3 CAN/CANopen/DNet Speed Reference".
- Disable Receive PDO 2 to 4. ❸

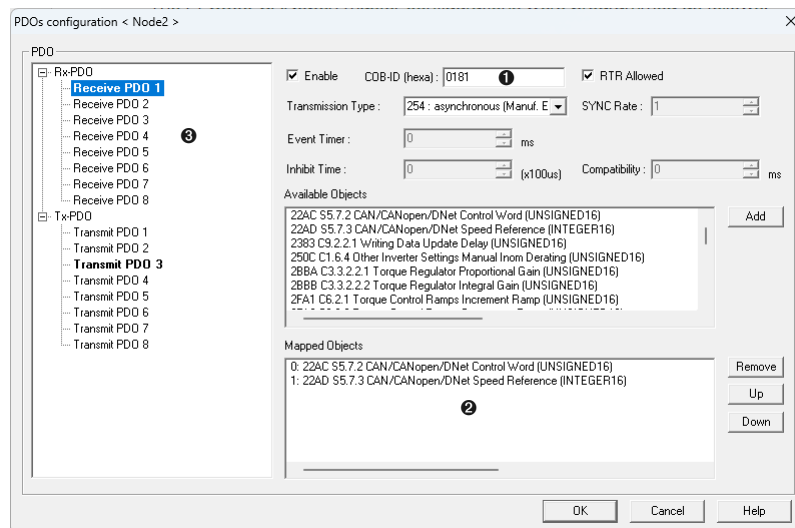


Figure 3.6: Slave's Receive PDO Configuration

Transmit PDO configuration:

- Set 100 ms as the Transmit PDO 3 Event Timer. ❶
- The TPDO3 default configuration for the mapped objects is: ❷
 - "22A8 S5.1.1 Status and Commands Status Word 1".
 - "22A9 S5.1.2 Status and Commands Speed".
- Disable Transmit PDO 1, 2 and 4. ❸

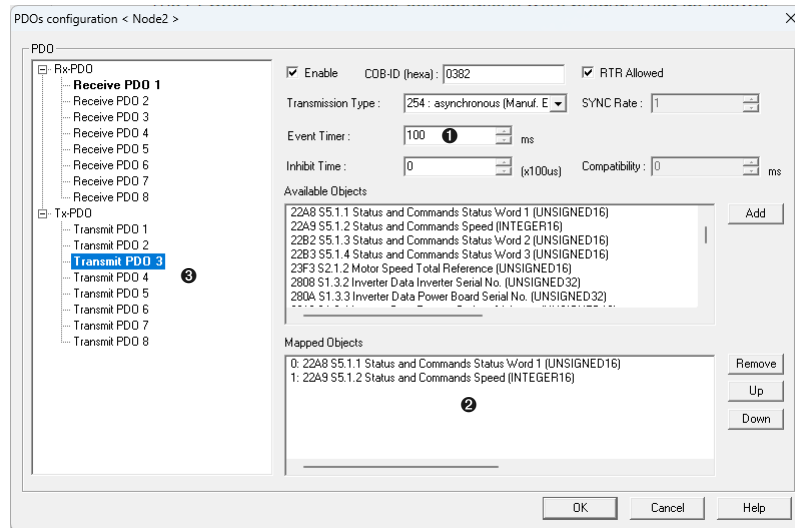


Figure 3.7: Slave's Transmit PDO Configuration

3.5 CONFIGURE MASTER PDOS

Now configure the master's PDOs to receive information from the slave's Transmit PDO and send commands to the slave's Receive PDO.

Use the PLC300 Manufacturer Specific objects to configure the PDOs. For this application, the following PLC300 objects have been used to link the CFW900 objects:

Table 3.1: Master/Slave Object Relationship

Slave data	Master Data
22A8 S5.1.1 Status and Commands Status Word 1	3400 IW2000 - Network Input Word
22A9 S5.1.2 Status and Commands Speed	3402 IW2002 - Network Input Word
22AC S5.7.2 CAN/CANopen/DNet Control Word	4400 QW2000 - Network Output Word
22AD S5.7.3 CAN/CANopen/DNet Speed Reference	4402 QW2002 - Network Output Word

Therefore, the PLC300 CANopen master configurations have been programmed as follows:

Receive PDO configuration:

- In the PDOs configuration window, set 382h as the COB-ID for Receive PDO 1. This is the slave's Transmit PDO 3 COB-ID. ❶
- The RPDO1 configuration for the mapped objects is: ❷
 - “3400 IW2000 - Network Input Word”.
 - “3402 IW2002 - Network Input Word”.
- Disable Receive PDO 2 to 4. ❸

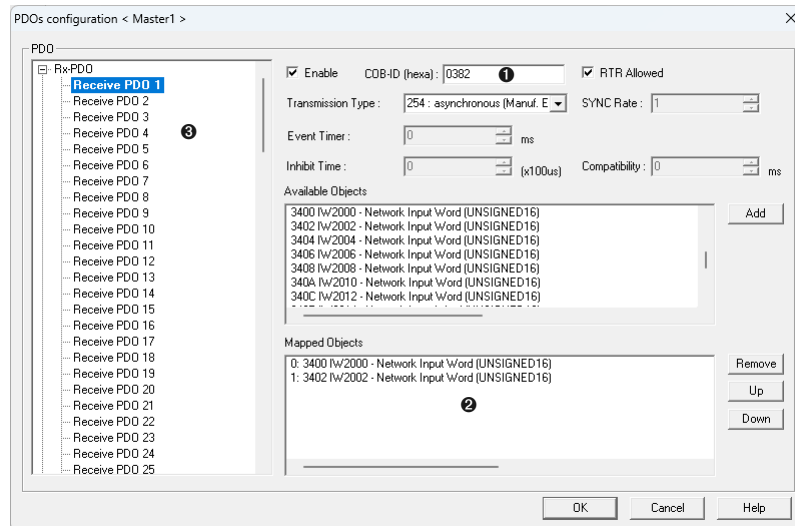


Figure 3.8: Master's Receive PDO Configuration

Transmit PDO configuration:

- Set 100 ms as the Transmit PDO 1 Event Timer. ①
- TPDO1 configuration for the mapped objects is: ②
 - “4400 QW2000 - Network Output Word”.
 - “4402 QW2002 - Network Output Word”.
- Disable Transmit PDO 2 to 4. ③

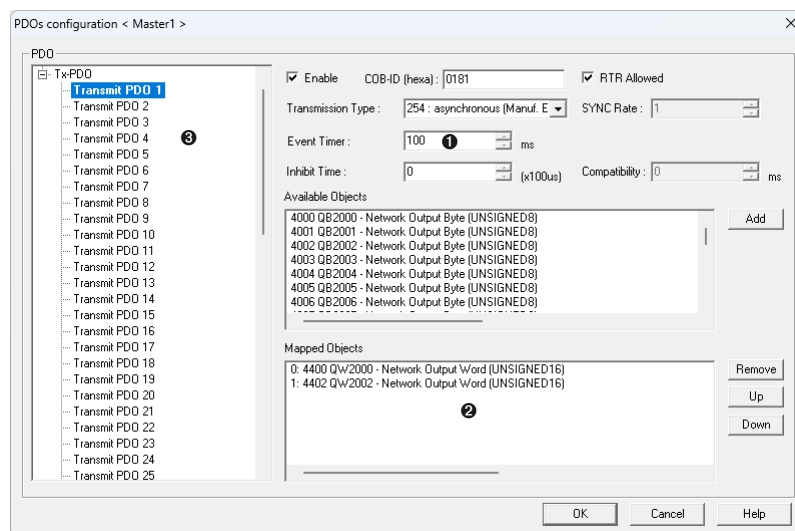
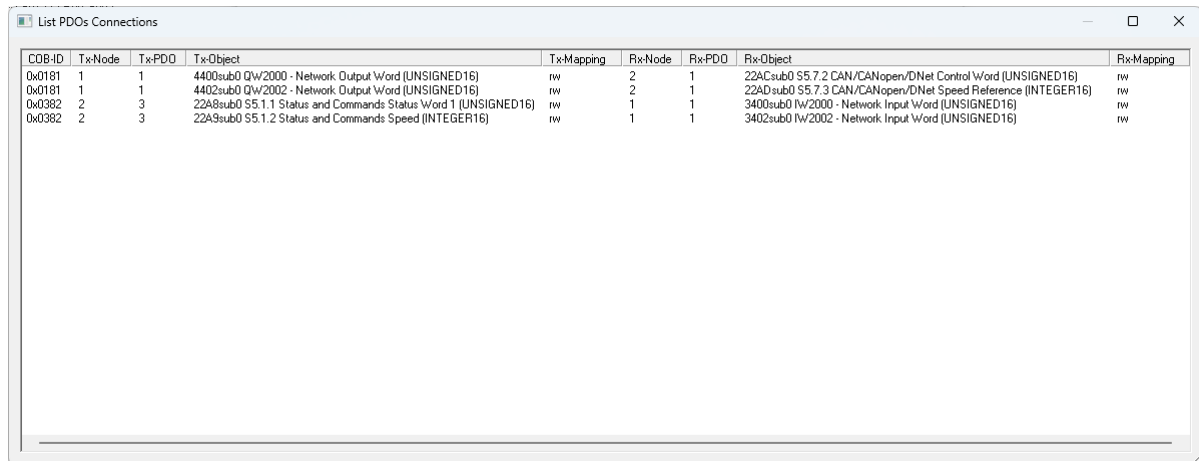


Figure 3.9: Master's Transmit PDO Configuration

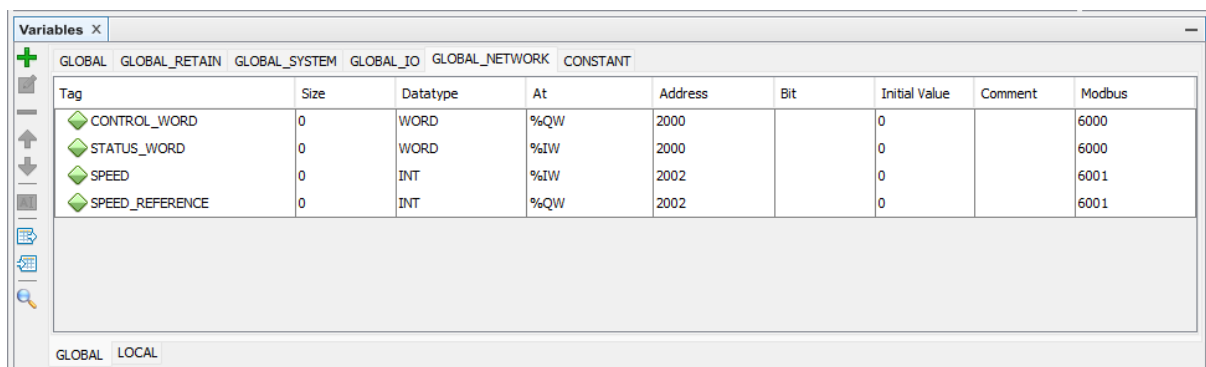
Verify the PDO connections in the menu “Tools / PDOs Connections”, press the “List Connections” button.



COB-ID	Tx-Node	Tx-PDO	Tx-Object	Tx-Mapping	Rx-Node	Rx-PDO	Rx-Object	Rx-Mapping
0x0181	1	1	4400sub0 Qw2000 - Network Output Word (UNSIGNED16)	rw	2	1	22ACsub0 S5.7.2 CAN/CANopen/DNet Control Word (UNSIGNED16)	rw
0x0181	1	1	4402sub0 Qw2002 - Network Output Word (UNSIGNED16)	rw	2	1	22ADsub0 S5.7.3 CAN/CANopen/DNet Speed Reference (INTEGER16)	rw
0x0382	2	3	22A8sub0 S5.1.1 Status and Commands Status Word 1 (UNSIGNED16)	rw	1	1	3400sub0 Iw2000 - Network Input Word (UNSIGNED16)	rw
0x0382	2	3	22A9sub0 S5.1.2 Status and Commands Speed (INTEGER16)	rw	1	1	3402sub0 Iw2002 - Network Input Word (UNSIGNED16)	rw

Figure 3.10: Master/Slave PDO Connections

The PLC300 Manufacturer Specific objects are mapped to the GLOBAL_NETWORK memory area. If you don't find them, create new variables in that datatype and address. Using these data it is possible to design a PLC program, creating variables representing device information and a PLC logic to manipulate such data according to the application.

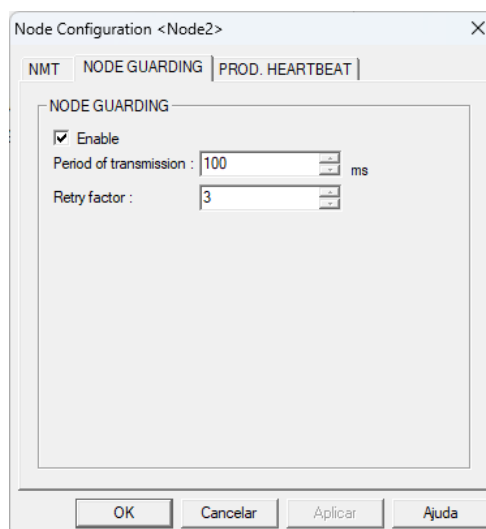


Tag	Size	Datatype	At	Address	Bit	Initial Value	Comment	Modbus
CONTROL_WORD	0	WORD	%QW	2000		0		6000
STATUS_WORD	0	WORD	%IW	2000		0		6000
SPEED	0	INT	%IW	2002		0		6001
SPEED_REFERENCE	0	INT	%QW	2002		0		6001

Figure 3.11: Location of Communication Data

3.6 CONFIGURE ERROR CONTROL

Click on the slave and select "Configuration". In the tab "NODE GUARDING" enable the node guarding service. This service is used to identify interruptions in the communication both by the master and the slave as well.



Node Configuration <Node2>

NMT | **NODE GUARDING** | PROD. HEARTBEAT

NODE GUARDING

☒ Enable

Period of transmission : 100 ms

Retry factor : 3

OK Cancelar Aplicar Ajuda

Figure 3.12: Node Guarding Service

3.7 DOWNLOAD CONFIGURATION

Once the configuration is finished, download it to PLC from the WPS software. Go to the “Download Resource” option under the “Online” menu.

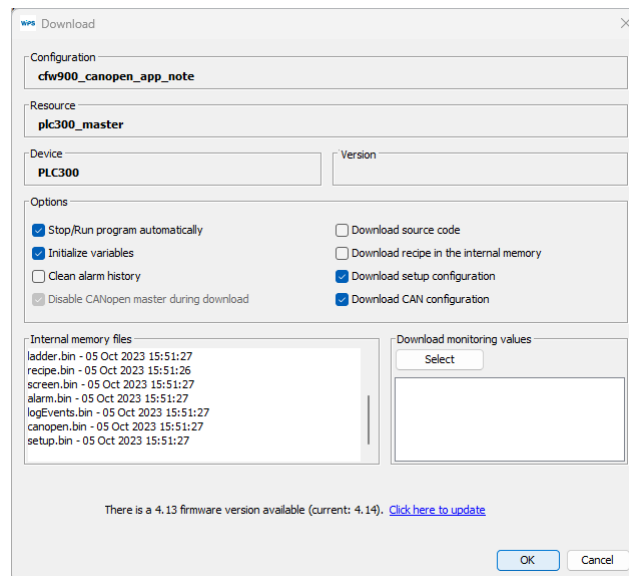


Figure 3.13: Downloading WPS Configuration to the PLC300

After transmitting the master configuration, monitor the devices to check if the device is connected to the network and effectively communicating. Go to the “Online Monitoring” option under the “Communication” menu in WPSCAN CANopen.

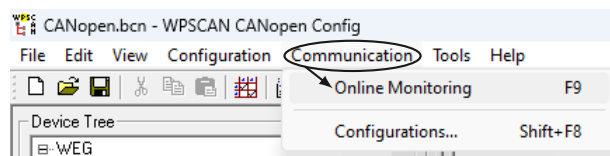


Figure 3.14: Online Monitoring

The indicators of all devices should be green, meaning that communication is active and without errors.

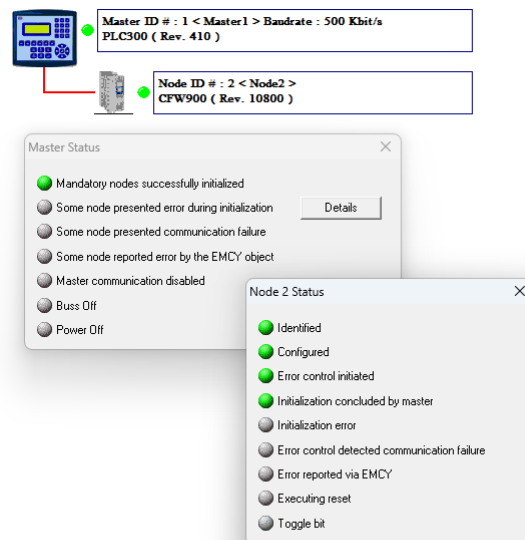


Figure 3.15: Viewer Online Monitoring

4 CONTROL AND MONITORING

Once network configuration is done, use WPS to access device data. Based on this, the main steps are described below.

4.1 VIEW AND EDIT CYCLIC DATA

Under the “Configurations” tab, right-click “cfw900_canopen_app_note/plc300_master/Diagnostic/Monitoring Variable” and add a new file. Click at the plus sign, select the GLOBAL_NETWORK variables and press OK to add them to the monitoring file.

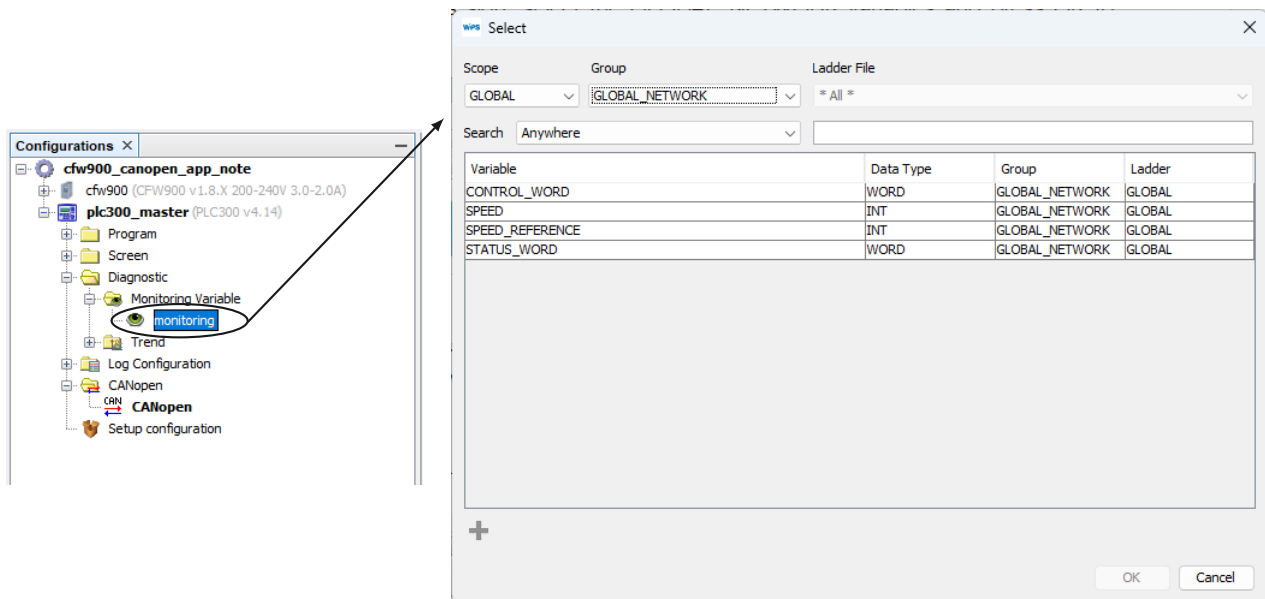


Figure 4.1: Monitoring Variable File

It is now possible to check input and write output data directly at controller memory. For inputs, as described in table 3.1, it is programmed to read Status Word and Actual Speed:

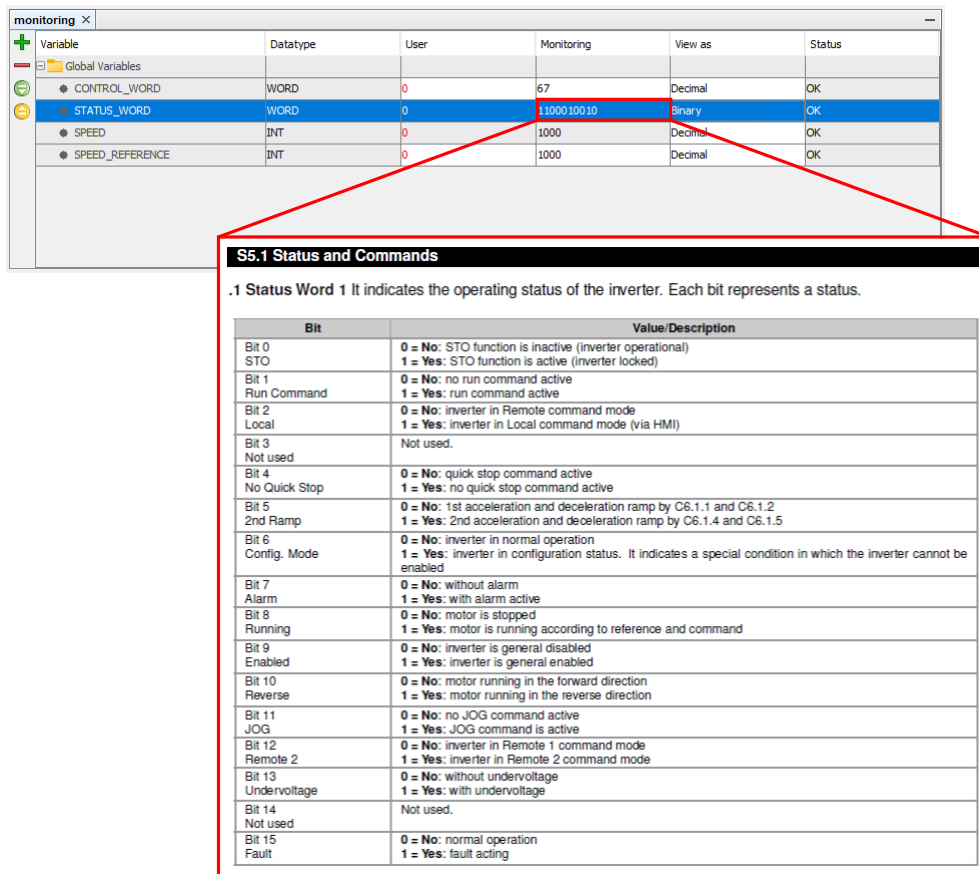


Figure 4.2: Read Status Word, with Highlight how as Described at CFW900 CANopen Documentation

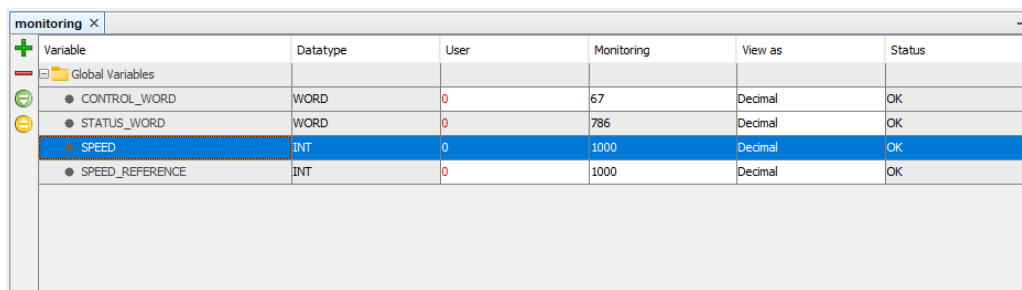


Figure 4.3: Read Speed

Therefore, the informations read about Status Word 1 and Speed in this case are:

■ STATUS_WORD: value 1100010010b.

- Bit 0 = 0 (STO).
- Bit 1 = 1 (Run Command).
- Bit 2 = 0 (Local).
- Bit 3 = 0 (Not Used).
- Bit 4 = 1 (No Quick Stop).
- Bit 5 = 0 (2nd Ramp).
- Bit 6 = 0 (Config. Mode).
- Bit 7 = 0 (Alarm).
- Bit 8 = 1 (Running).
- Bit 9 = 1 (Enabled).

- Bit 10 = 0 (Reverse).
- Bit 11 = 0 (JOG).
- Bit 12 = 0 (Remote 2).
- Bit 13 = 0 (Undervoltage).
- Bit 14 = 0 (Not Used).
- Bit 15 = 0 (Fault).

■ SPEED: value 1000.

For output, as described in table 3.1, it is programmed to write Control Word and Speed Reference informations:

The screenshot shows a 'monitoring' window with a table of variables. The 'CONTROL_WORD' variable is highlighted, showing a value of 1000011 in binary. Below this, a detailed view of the 'S5.7 CAN/CANopen/DNet' parameter is shown, explaining the bits of the control word.

Variable	Datatype	User	Monitoring	View as	Status
CONTROL_WORD	WORD	0	1000011	Binary	OK
STATUS_WORD	WORD	0	786	Decimal	OK
SPEED	INT	0	1000	Decimal	OK
SPEED_REFERENCE	INT	0	1000	Decimal	OK

S5.7 CAN/CANopen/DNet

.2 Control Word It indicates the status of the control word via CAN interface. This parameter can only be changed via CAN interface. For other sources, only read access is allowed.

For the commands written in this parameter to be executed, the inverter must be programmed to be commanded via CAN/CO/DN. This programming is done through menu C4.

Each bit of this word represents a command that can be executed on the inverter.

Bit	Value/Description
Bit 0 Enable Ramp	0 = No: stops the motor by deceleration ramp 1 = Yes: the motor turns according to the acceleration ramp until reaching the speed reference value
Bit 1 General Enable	0 = No: disables the inverter completely, interrupting the motor power supply 1 = Yes: enables the inverter completely, allowing the operation of the motor
Bit 2 Run Reverse	0 = No: runs the motor in the direction of the reference signal (forward) 1 = Yes: runs the motor in the opposite direction of the reference signal (reverse)
Bit 3 Enable JOG	0 = No: disables the JOG function 1 = Yes: enables the JOG function
Bit 4 R1/R2 Mode	0 = R1: selects the Remote 1 command mode 1 = R2: selects the Remote 2 command mode
Bit 5 2nd Ramp	0 = No: 1st ramp acceleration and deceleration according to parameters C6.1.1 and C6.1.2 1 = Yes: 2nd ramp acceleration and deceleration according to parameters C6.1.4 and C6.1.5
Bit 6 No Quick Stop	0 = No: enables quick stop 1 = Yes: disables quick stop
Bit 7 Fault Reset	0 = No: not used 1 = Yes: in the transition, if in a fault state, it resets the fault

Figure 4.4: Write Control Word, with Highlight how as Described at CFW900 CANopen Documentation

The screenshot shows the same 'monitoring' window. The 'SPEED_REFERENCE' variable is now highlighted, showing a value of 1000 in decimal.

Variable	Datatype	User	Monitoring	View as	Status
CONTROL_WORD	WORD	0	67	Decimal	OK
STATUS_WORD	WORD	0	786	Decimal	OK
SPEED	INT	0	1000	Decimal	OK
SPEED_REFERENCE	INT	0	1000	Decimal	OK

Figure 4.5: Write Speed Reference

Therefore, the informations write in Control Word and Speed Reference in this case are:

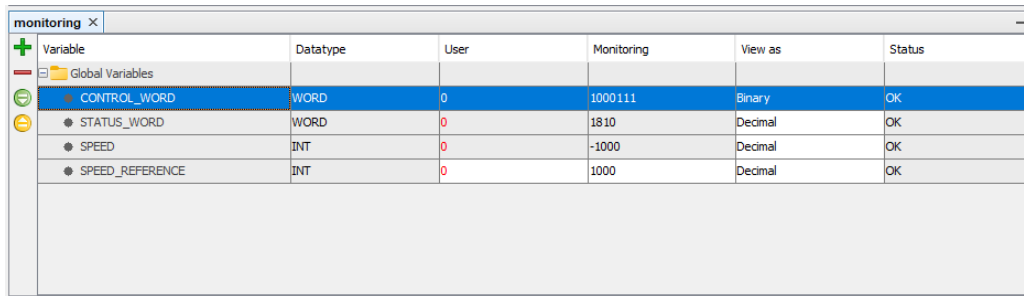
- CONTROL_WORD: value 1000011b.
 - Bit 0 = 1 (Enable Ramp).
 - Bit 1 = 1 (General Enable).
 - Bit 2 = 0 (Run Reverse).
 - Bit 3 = 0 (Enable JOG).

CONTROL AND MONITORING

- Bit 4 = 0 (R1/R2 Mode).
- Bit 5 = 0 (2nd Ramp).
- Bit 6 = 1 (No Quick Stop).
- Bit 7 = 0 (Fault Reset).

■ SPEED_REFERENCE: value 1000.

For example, when changing the value of Bit 2 (Run Reverse) in Control Word and keeping Speed Reference value, it is possible to notice that the Status Word and Speed has also changed.



Variable	Datatype	User	Monitoring	View as	Status
Global Variables					
CONTROL_WORD	WORD	0	1000111	Binary	OK
STATUS_WORD	WORD	0	1810	Decimal	OK
SPEED	INT	0	-1000	Decimal	OK
SPEED_REFERENCE	INT	0	1000	Decimal	OK

Figure 4.6: Example Write and Read

4.2 LADDER LOGIC FOR ACYCLIC DATA TRANSFER

In addition to the cyclic data transmitted through PDO's, it is also possible to transfer data acyclically via SDO's. Using this type of communication, you can access any parameter of the equipment.

Access to this type of data is commonly done using instructions for reading or writing data, which should indicate the index and subindex of the desired parameter. Moreover, there is an appendix in the CANopen User's Guide that describes the entire list of device data that can be accessed through the CANopen interface, this list also indicates the index of each data.

For example, is possible to read the value of S5.1.3 - Status and Commands Status Word 2. In this case:

Parameter	Description	Range of values	Decimal places	Index	Net Id	Size
S5.1.3	Status Word 2	Bit 0 = Self-tuning Bit 1 = Not used Bit 2 = Pre-Charge OK Bit 3 = SF Reduction Bit 4 = Not used Bit 5 = Decel. Ramp Bit 6 = Acc. Ramp Bit 7 = Freeze Ramp Bit 8 = Setpoint OK Bit 9 = DC Voltage Limitation Bit 10 = Current Limitation Bit 11 = Torque Limitation Bit 12 = Ride-Through Bit 13 = Flying Start Bit 14 = DC Braking Bit 15 = PWM pulses		22B2h	690	16bit

Figure 4.7: CFW900 CANopen Documentation Describing Index for Acyclic Access

Once defined the information for the acyclic access, add a CO_SDORed block to the PLC logic. Then, configure the following informations:

- NodeID#: slave address (for this example, CFW900 at address 2). ❶
- Index# as described by CFW900 CANopen documentation (for this example, WORD#16#22B2). ❷
- SubIndex# is always 0 (zero) for the CFW900. ❸

- Size#: size in bytes of the Value output variable (for this example, 2 bytes). ④
- Timeout#: waiting time in ms for the arrival of data starting from the beginning of the request (for this example, 500 ms). ⑤
- Value: a variable to store the read value (must be compatible with the data size of the reading object). ⑥

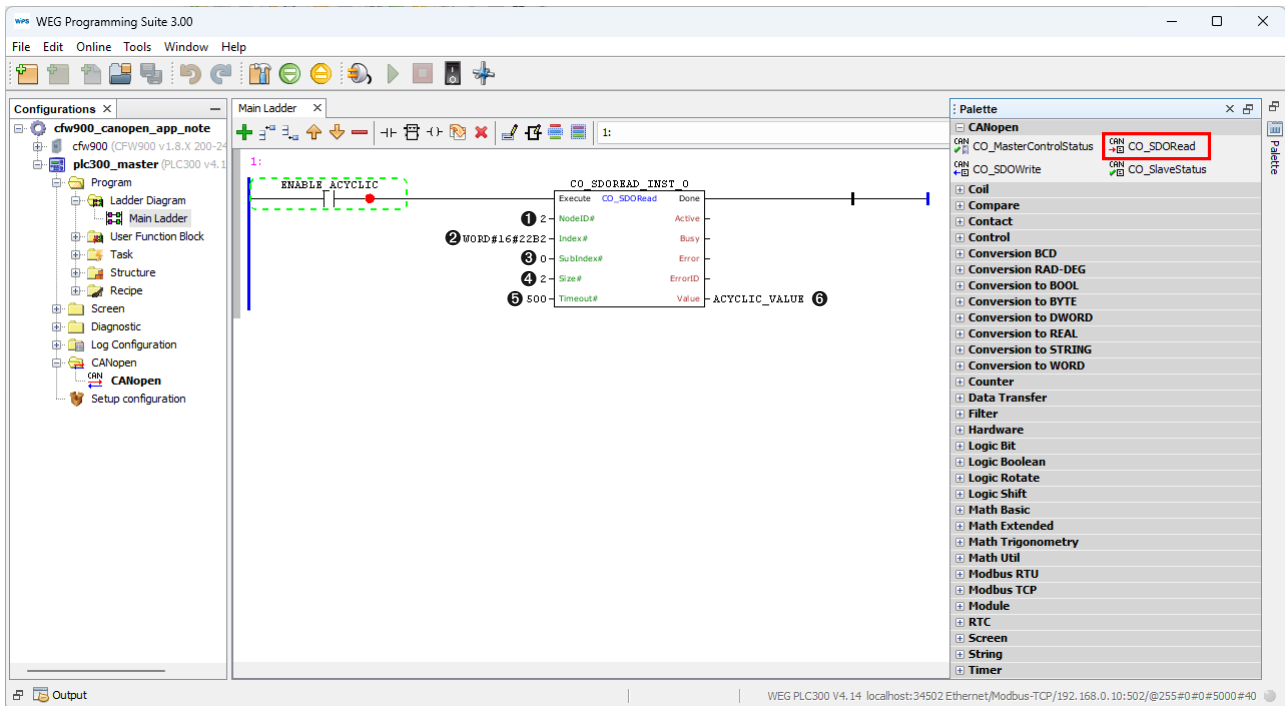


Figure 4.8: Acyclic Message Configuration

After that, activate CO_SDOREAD_INST_0 block and execute input ❶ to send an acyclic request to slave. For this example, the value read from S5.1.3 is “4” ❷, that is:

- STATUS_WORD_2: value 100b.
 - Bit 0 = 0 (Self-tuning).
 - Bit 1 = 0 (Not Used).
 - Bit 2 = 1 (Pre-Charge OK).
 - Bit 3 = 0 (SF Reduction).
 - Bit 4 = 0 (Not Used).
 - Bit 5 = 0 (Decel. Ramp).
 - Bit 6 = 0 (Acc. Ramp).
 - Bit 7 = 0 (Freeze Ramp).
 - Bit 8 = 0 (Setpoint OK).
 - Bit 9 = 0 (DC Voltage Limitation).
 - Bit 10 = 0 (Current Limitation).
 - Bit 11 = 0 (Torque Limitation).
 - Bit 12 = 0 (Ride-Through).
 - Bit 13 = 0 (Flying Start).
 - Bit 14 = 0 (DC Braking).
 - Bit 15 = 0 (PWM Pulses).

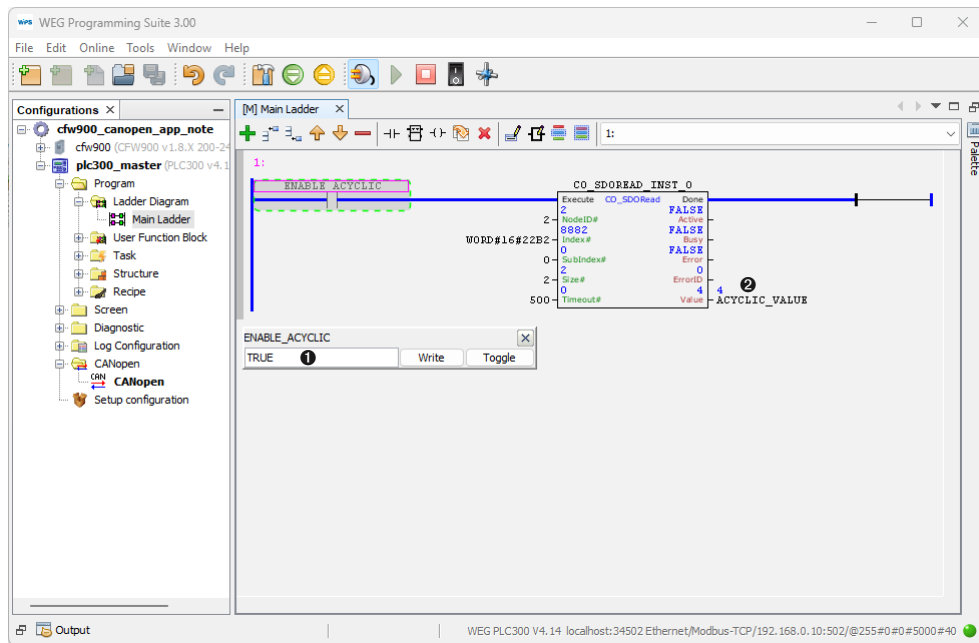


Figure 4.9: Main Ladder Sending an Acyclic Message to the Slave



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