Quick start-up guide
ABB Drives function blocks for Siemens PLC's

ABB

List of related manuals

Drive application firmware manuals and guides	Code (English)
FPBA-01 PROFIBUS DP adapter module User's manual	3AFE68573271
PROFIBUS DP Adapter Module RPBA-01 User's manual	3AFE64504215
ACS355 drives User's manual	3AUA0000066143
ACS850 Standard Control Program Firmware manual	3AUA0000045497
ACS880 primary control program Firmware manual	3AUA0000085967
ACSM1 speed and torque control program Firmware manual	3AFE68848261
ACSM1 motion control firmware manual	3AFE68848270
ACS550-01 Drives User's manual	3AUA0000001418
ACS800 Standard Control Program Firmware Manual	3AFE64527592

You can find manuals and other product documents in PDF format on the Internet. See section *Document library on the Internet* on the inside of the back cover. For manuals not available in the Document library, contact your local ABB representative.

3

Quick start-up guide

ABB Drives function blocks for Siemens PLC's

3AXD50000037864 Rev A

EN
EFFECTIVE: 2016-12-07

Contents

Introduction

This guide describes the configuration of ABB Drives function blocks for Siemens PLC's. ABB specific ready-made function blocks from the *ABB_DRIVES_LIB* are used.

Compatibility

This guide applies to the following libraries.

Libraries		Supported PCL series	ABB supported Drives
SIMATIC Manager version 5.5 or later	ABB_Drives_SIMACTIC_Library_ 3xx_4xx_PLC_V1.1	S7-300 & S7-400	ACS800, ACSM1, ACS350, ACS355, ACS550, ACS850 ACS880, ACS580, ACS380
TIA Portal version 11 and 13	ABB_Drives_TIA_Global_Library _3xx_4xx_PLC_V1.1	S7-300 & S7-400	
	ABB_Drives_TIA_Global_Library _12xx_15xx_PLC_V1.1	S7-1200 & S7-1500	

Limitations

The following limitations are valid for this guide:

- The drive must be equipped with PROFIBUS or PROFINET adapters
- PROFIBUS DP protocol: DP-V0 or DP-V1
- PROFIBUS DP communication profile: ABB drives
- All PROFIBUS DP, PROFINET and compatible ABB drives
- Application types: Speed/Frequency control or Torque control
- Supports PPO types with consistent data, example, PPO-06, 0 PKW + 10 PZD.
- All the libraries are encrypted with a passcode. If you want to edit the libraries, contact your local ABB representative for passcode.

Note! PPO types without consistent data, example, PPO-06, 0 PKW + (2+2+2+2+2) PZD or PPO-06, 0 PKW +NoCons. 10 PZD are **not** supported by *ABB_DRIVE_LIB*.

Note! The *ABB_DRIVES_LIB* can also be used for PROFINET control of ABB drives. When programming the function blocks, the drive should be considered as a PROFIBUS DP-V1 slave. Note that the hardware configuration (GSDML files) and the drive parameter settings for PROFINET slaves differ from PROFIBUS slaves.

9

Setup

The following figure shows an example of SIMATIC Manager/TIA portal with ABB drives and PROFIBUS module setup.

SIMATIC Manager/TIA portal

ETHERNET or serial connection

Siemens PLC

ABB drives

ABB PROFIBUS modules
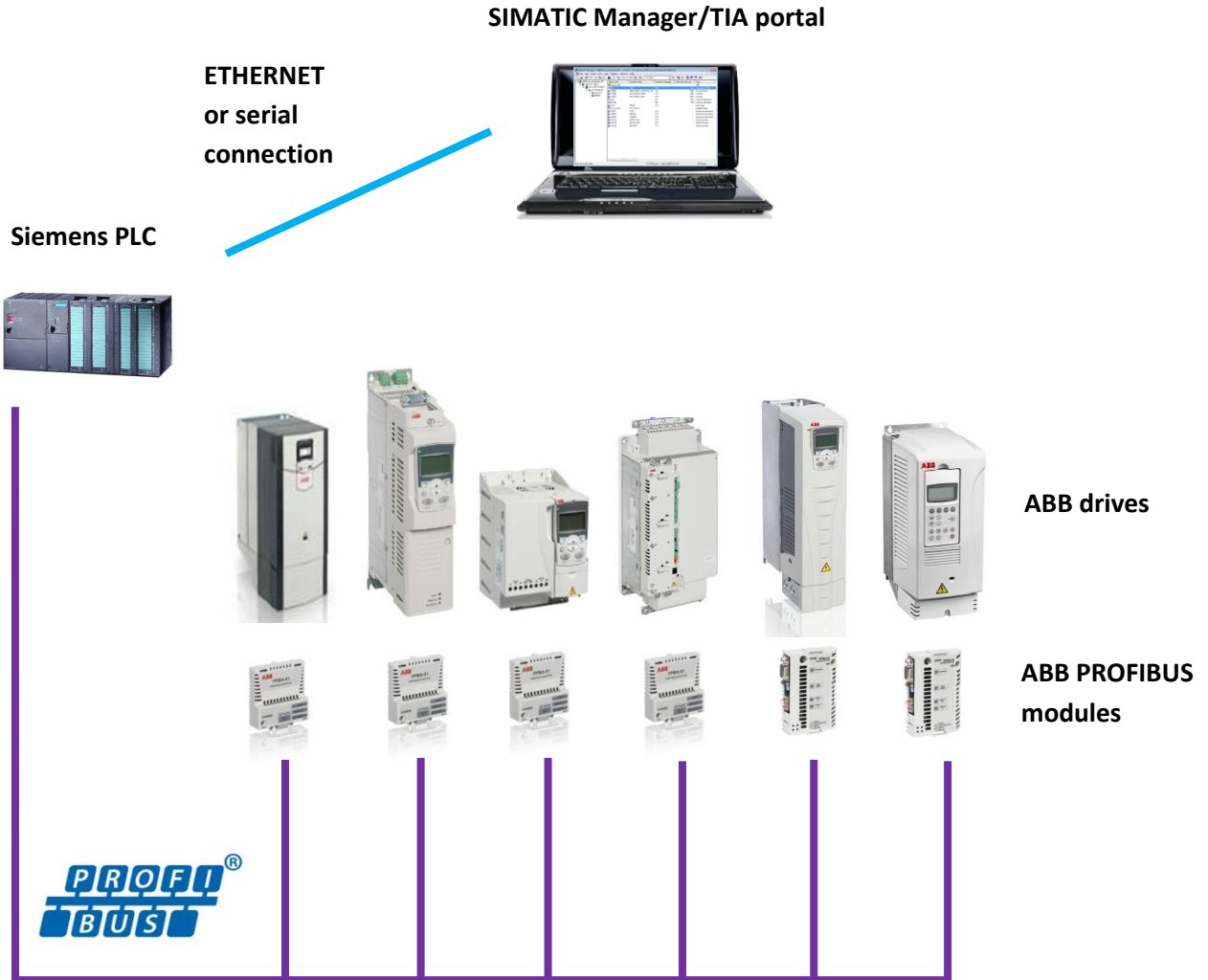
Cyber security disclaimer

This product is designed to be connected to and to communicate information and data via a network interface. It is Customer's sole responsibility to provide and continuously ensure a secure connection between the product and Customer network or any other network (as the case may be). Customer shall establish and maintain any appropriate measures (such as but not limited to the installation of firewalls, application of authentication measures, encryption of data, installation of anti-virus programs, etc) to protect the product, the network, its system and the interface against any kind of security breaches, unauthorized access, interference, intrusion, leakage and/or theft of data or information. ABB and its affiliates are not liable for damages and/or losses related to such security breaches, any unauthorized access, interference, intrusion, leakage and/or theft of data or information.

Disclaimer

ABB is not liable for personal injury, material damage or monetary losses due to non-functionality, incorrect behavior or misuse of application program and the function blocks created by ABB or third party.

Drive configuration

You can configure the drive for PROFIBUS DP or PROFINET control based on ABB specific drives library *ABB_DRIVE_LIB*.

Set the drive parameters from the drive's control panel or from drive-specific PC tool (DriveWindow Light for ACS355 and ACS550, DriveStudio for ACS850 and ACSM1, Drive Composer for ACS380, ACS580 and ACS880, and DriveWindow for ACS800).

Note! All parameter settings are based on drive default settings. If the drive was parametrized previously, return to default settings before continuing. It can be done by:

- Changing macro (and then changing back again) in parameter **99.02** for ACS355 and ACS550.
- Setting parameter **99.03** APPLIC RESTORE to **YES** in ACS800.
- Setting parameter **16.04** Param restore to **Restore defs** in ACS850 and ACSM1.
- Setting parameter **96.06** Parameter restore to **Restore defaults** in ACS380, ACS580 and ACS880.

Configuring ACS355 drives

Starting ACS355 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *98.02 COMM PROT SEL* to *EXT FBA*.
3. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02* and the communication profile in *51.05 = 1* (ABB drives).
4. With parameter *30.18 COMM FAULT FUNC*, select how the drive reacts to a fieldbus communication break.
5. With parameter *30.19 COMM FAULT TIME*, define the time between communication break detection and the selected action.
6. Define the process data transferred to and from the drive in the FBA-01 configuration parameter groups 54 and 55.
   Note! The adapter module sets the Status word and actual value automatically in parameters *54.01* and *54.02*, and Control word and reference in parameters *55.01* and *55.02*.
7. Validate the settings made in parameter groups 51, 54 and 55 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
8. Set the relevant drive control parameters to control the drive according to the application.

ACS355 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings are based on factory default settings.

Parameter	Description	Setting	Comment
98.02	COMM PROT SEL	EXT FBA	Activates fieldbus module
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.05	PROFILE	1	Communication profile ABB drives

51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
10.01	EXT 1 COMMANDS	COMM	Fieldbus interface as source for start and stop
11.03	REF1 SELECT	COMM	Fieldbus interface as source for speed reference
16.04	FAULT RESET SEL	COMM	Fieldbus interface as source for fault reset
(11.05)	REF1 MAX	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS355 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *98.02 COMM PROT SEL* to *EXT FBA*.
3. Set the FENA configuration parameters in group 51. At the minimum, set parameters *51.01 FBA A type* as *Ethernet*, *51.02 Protocol/Profile* as *PNIO ABB Pro (11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24.
4. Select how the drive reacts to a fieldbus communication break with parameter *30.18 COMM FAULT FUNC.*
5. Define the time between communication break detection and the selected action with parameter *30.19 COMM FAULT TIME.*
6. Define the process data transferred to and from the drive in the FBA-01 configuration parameter groups 54 and 55.
   Note! The adapter module sets the Status word and actual value automatically in parameters *54.01* and *54.02*, and Control word and reference in parameters *55.01* and *55.02*.
7. Validate the settings made in parameter groups 51, 54 and 55 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
8. Set the relevant drive control parameters to control the drive according to the application.

ACS355 Minimum required parameter settings for PROFINET

The minimum required parameter settings are based on factory default settings.

Parameter	Description	Setting	Comment
98.02	COMM PROT SEL	EXT FBA	Activates fieldbus module
51.01	FBA A type	Ethernet	Fieldbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
10.01	EXT1 COMMANDS	COMM	Fieldbus interface as a source for start and stop.
11.03	REF1 SELECT	COMM	Fieldbus interface as a source for

			speed reference
16.04	FAULT RESET SEL	COMM	Fieldbus interface as a source for fault reset
(11.05)	REF1 MAX	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

ACS355 More parameters read from drive to PLC (optional)

Parameter	Description	Setting	Comment
54.03 .. 54.10	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0104 = actual current; 0145 = motor temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)

ACS355 More parameters written from PLC to drive (optional)

Parameter	Description	Setting	Comment
55.03 .. 55.10	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2202 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)

Configuring ACS380 drives

Starting ACS380 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA A Enable* to *Enable*.
3. With parameter *50.02 FBA A comm loss func*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 FBA A comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for the rest of the parameters in group 50, starting from *50.04*.
6. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02 Node address* and the communication profile in *51.05 Profile* = **1** (ABB drives).
7. Define the process data transferred to and from the drive in FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01 FBA A data in1* and *53.01 FBA data out1*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Configure*.
9. Save the valid parameter values to permanent memory by setting parameter *96.07 Param save manually* to *Save*.
10. Set the relevant drive control parameters to control the drive according to the application.

ACS380 Minimum required parameter settings PROFIBUS DS

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA A Enable	Enable	Activates fieldbus module
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.05	PROFILE	1	Communication profile ABB drives
52.02	FBA data in2	Act1 16bit (5)	Actual speed as Data Word 2 from the drive
53.02	FBA data out2	Ref1 16bit (2)	Speed reference as Data Word 2 to the drive
51.27	FBA par refresh	Configure	Updates fieldbus settings (groups 50 to 57)
20.01	Ext 1 commands	Fieldbus A	Fieldbus interface as source for start and stop
22.11	Ext1 speed ref1	FB A ref1	Fieldbus interface as source for speed reference
31.11	Fault reset selection	P.06.01 bit 7	Fieldbus interface as source for fault reset
(46.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS380 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA A Enable* to *Enable*.
3. With parameter *50.02 FBA A comm loss func*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 FBA A comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for the rest of the parameters in group 50, starting from *50.04*.
6. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02 Node address* and the communication profile in *51.05 Profile* = 1 (ABB drives).
7. Define the process data transferred to and from the drive in FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01 FBA A data in1* and *53.01 FBA data out1*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Configure*.
9. Save the valid parameter values to permanent memory by setting parameter *96.07 Param save manually* to *Save*.
10. Set the relevant drive control parameters to control the drive according to the application.

ACS380 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA A Enable	Enable	Activates fieldbus module
51.01	FBA A ype	Ethernet	Filedbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
52.02	FBA data in2	Act1 16bit (5)	Actual speed as Data Word 2 from the drive
53.02	FBA data out2	Ref1 16bit (2)	Speed reference as Data Word 2 to the drive
20.01	Ext 1 commands	Fieldbus A	Fieldbus interface as source for start and stop
22.11	Ext1 speed ref1	FB A ref1	Fieldbus interface as source for speed reference
31.11	Fault reset selection	P.06.01 bit 7	Fieldbus interface as source for fault reset
(46.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

ACS380 More parameters read from drive to PLC (optional)

Parameter	Description	Setting	Comment
52.03 .. 52.12	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0107 = actual current; 3501 = motor est. temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 57)

ACS380 More parameters written from PLC to drive (optional)

Parameter	Description	Setting	Comment
53.03 .. 53.12	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2312 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 57)

Configuring ACS550 drives

Starting ACS550 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *98.02 COMM PROT SEL* to *EXT FBA*.
3. Set the RPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02*.
4. With parameter *30.18 COMM FAULT FUNC*, select how the drive reacts to a fieldbus communication break.
5. With parameter *30.19 COMM FAULT TIME*, define the time between communication break detection and the selected action.
6. Define the process data transferred to and from the drive in the RPBA-01 configuration parameter group 51.
   Note: The Status Word, Actual Speed, Control Word and Speed Reference are fixed in ACS550 and not necessary to set.
7. Validate the settings made in parameter group 51 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
8. Set the relevant drive control parameters to control the drive according to the application.

ACS550 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
98.02	COMM PROT SEL	EXT FBA	Activates fieldbus module
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.21	DP MODE	[DP MODE]	1 for DPV1, 0 for DPV0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (group 51)
10.01	EXT 1 COMMANDS	COMM	Fieldbus interface as source for start and stop
11.03	REF1 SELECT	COMM	Fieldbus interface as source for speed reference
16.04	FAULT RESET SEL	COMM	Fieldbus interface as source for fault reset
(11.05)	REF1 MAX	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS550 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *98.02 COMM PROT SEL* to *EXT FBA*.
3. Set the FENA configuration parameters in group 51. At the minimum, set the *51.01 FBA A type* as *Ethernet*, *51.02 Protocol/Profile* as *PNIO ABB Pro(11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24.
4. With parameter *30.18 COMM FAULT FUNC*, select how the drive reacts to a fieldbus communication break.
5. With parameter *30.19 COMM FAULT TIME,* define the time between communication break detection and the selected action.
6. Define the process data transferred to and from the drive in the RPBA-01 configuration parameter group 51.
   Note: The Status Word, Actual Speed, Control Word and Speed Reference are fixed in ACS550 and not necessary to set.
7. Validate the settings made in parameter groups 51, 54 and 55 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
8. Set the relevant drive control parameters to control the drive according to the application.

ACS550 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
98.02	COMM PROT SEL	EXT FBA	Activates fieldbus module
51.01	FBA A ype	Ethernet	Filedbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
10.01	EXT 1 COMMANDS	COMM	Fieldbus interface as source for start and stop
11.03	REF1 SELECT	COMM	Fieldbus interface as source for speed reference
16.04	FAULT RESET SEL	COMM	Fieldbus interface as source for fault reset
(11.05)	REF1 MAX	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

ACS550 More actual values or parameters to be read from drive to PLC (optional)

Parameter	Description	Setting	Comment
51.06,51.08 ,…..,51.20	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0104 = actual current; 0145 = motor temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (group 51)

ACS550 More parameters to be written from PLC to drive (optional)

Parameter	Description	Setting	Comment
51.05,51.07 ,…..,51.19	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2202 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (group 51)

Configuring ACS580 drives

Starting ACS580 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA A Enable* to *Enable*.
3. With parameter *50.02 FBA A comm loss func*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 FBA A comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for the rest of the parameters in group 50, starting from *50.04*.
6. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02 Node address* and the communication profile in *51.05 Profile* = 1 (ABB drives).
7. Define the process data transferred to and from the drive in FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01* and *53.01*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Configure*.
9. Save the valid parameter values to permanent memory by setting parameter *96.07 Parameter save maually* to *Save*.
10. Set the relevant drive control parameters to control the drive according to the application.

ACS580 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA A Enable	Enable	Activates fieldbus module
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.05	PROFILE	1	Communication profile ABB drives
52.02	FBA data in2	Act1 16bit (5)	Actual speed as Data Word 2 from the drive
53.02	FBA data out2	Ref1 16bit (2)	Speed reference as Data Word 2 to the drive
51.27	FBA par refresh	Configure	Updates fieldbus settings (groups 50 to 57)
20.01	Ext 1 commands	Fieldbus A	Fieldbus interface as source for start and stop
22.11	Ext1 speed ref1	FB A ref1	Fieldbus interface as source for speed reference
31.11	Fault reset selection	P.06.01 bit 7	Fieldbus interface as source for fault reset
(46.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS580 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA A Enable* to *Enable*.
3. With parameter *50.02 FBA A comm loss func*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 FBA A comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for the rest of the parameters in group 50, starting from *50.04*.
6. Set the FENA configuration parameters in group 51. At the minimum, set the *51.01 FBA A type* as *Ethernet*, *51.02 Protocol/Profile* as *PNIO ABB Pro(11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24.
7. Define the process data transferred to and from the drive in FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01* and *53.01*.
8. Validate the settings made in parameter groups 51, 54 and 55 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
9. Save the valid parameter values to permanent memory by setting parameter *96.07 Parameter save manually* to *Save*.
10. Set the relevant drive control parameters to control the drive according to the application.

ACS580 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA A Enable	Enable	Activates fieldbus module
51.01	FBA A ype	Ethernet	Filedbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
52.02	FBA data in2	Act1 16bit (5)	Actual speed as Data Word 2 from the drive
53.02	FBA data out2	Ref1 16bit (2)	Speed reference as Data Word 2 to the drive
20.01	Ext 1 commands	Fieldbus A	Fieldbus interface as source for start and stop
22.11	Ext1 speed ref1	FB A ref1	Fieldbus interface as source

			for speed reference
31.11	Fault reset selection	P.06.01 bit 7	Fieldbus interface as source for fault reset
(46.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

ACS580 More parameters read from drive to PLC (optional)

Parameter	Description	Setting	Comment
52.03 .. 52.12	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0107 = actual current; 3501 = motor est. temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 57)

ACS580 More parameters written from PLC to drive (optional)

Parameter	Description	Setting	Comment
53.03 .. 53.12	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2312 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 57)

Configuring ACS800 drives

Starting ACS800 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *98.02 COMM. MODULE LINK* to *FIELDBUS*.
3. Set the RPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02*.
4. With parameter *30.18 COMM FLT FUNC*, select how the drive reacts to a fieldbus communication break.
5. With parameter *30.19 MAIN REF DS T-OUT*, define the time between communication break detection and the selected action.
6. Define the process data transferred to and from the drive in the RPBA-01 configuration parameter group 51.
   Note: The Status Word, Actual Speed, Control Word and Speed Reference are configured as default.
7. Validate the settings made in parameter group 51 by setting parameter *51.27 FBA PAR REFRESH* to REFRESH.
8. Set the relevant drive control parameters to control the drive according to the application.

ACS800 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
98.02	COMM. MODULE LINK	FIELDBUS	Activates fieldbus module
98.07	COMM PROFILE	ABB DRIVES	Communication profile ABB drives
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.21	DP MODE	[DP MODE]	1 for DPV1, 0 for DPV0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
10.01	EXT 1 STRT/STP/DIR	COMM.CW	Fieldbus interface as source for start and stop
11.03	EXT REF1 SELECT	COMM.REF	Fieldbus interface as source for speed reference
16.04	FAULT RESET SEL	COMM.CW	Fieldbus interface as source for fault reset
(11.05)	EXT REF1 MAXIMUM	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS800 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *98.02 COMM. MODULE LINK* to *FIELDBUS*.
3. Set the FENA configuration parameters in group 51. At the minimum, set parameters *51.01 FBA A type* as *Ethernet*, *51.02 Protocol/Profile* as *PNIO ABB Pro (11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24.
4. With parameter *30.18 COMM FLT FUNC*, select how the drive reacts to a fieldbus communication break.
5. With parameter *30.19 MAIN REF DS T-OUT*, define the time between communication break detection and the selected action.
6. Define the process data transferred to and from the drive in the RPBA-01 configuration parameter group 51.
   Note: The Status Word, Actual Speed, Control Word and Speed Reference are configured as default.
7. Validate the settings made in parameter groups 51, 54 and 55 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
8. Set the relevant drive control parameters to control the drive according to the application.

ACS800 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
98.02	COMM. MODULE LINK	FIELDBUS	Activates fieldbus module
98.07	COMM PROFILE	ABB DRIVES	Communication profile ABB drives
51.01	FBA A ype	Ethernet	Filedbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
10.01	EXT 1 STRT/STP/DIR	COMM.CW	Fieldbus interface as source for start and stop
11.03	EXT REF1 SELECT	COMM.REF	Fieldbus interface as source for speed reference
16.04	FAULT RESET SEL	COMM.CW	Fieldbus interface as source for fault reset
(11.05)	EXT REF1 MAXIMUM	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

ACS800 More parameters read from drive to PLC (optional)

Parameter	Description	Setting	Comment
51.06,51.08 ,…..,51.20	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0104 = actual current; 0145 = motor temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (group 51)

ACS800 More parameters written from PLC to drive (optional)

Parameter	Description	Setting	Comment
51.05,51.07 ,…..,51.19	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2202 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (group 51)

Configuring ACS850 drives

Starting ACS850 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA* enable to *Enable.*
3. Select how the drive reacts to a fieldbus communication break with parameter *50.02 Comm loss func*, Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 Comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for parameters *50.04…50.11*.
6. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02* and the communication profile in *51.05* **= 1** (ABB drives).
7. Define the process data transferred to and from the drive in the FBA-01 configuration parameter groups 52 and 53.
   Note! The adapter module sets the Status word and actual value automatically in parameters *52.01* and *52.02*, and Control word and reference in parameters *53.01* and *53.02*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Refresh*.
9. Set the relevant drive control parameters to control the drive according to the application.

ACS850 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	Fba enable	Enable	Activates fieldbus module
51.05	PROFILE	1	Communication profile ABB drives
10.01	Ext1 start func	FB	Fieldbus interface as source for start and stop
21.01	Speed ref1 sel	FBA ref1	Fieldbus interface as source for speed reference
10.10	Fault reset sel	P.02.22 bit 8	Fieldbus interface as source for fault reset
(19.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS850 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA* enable to *Enable*.
3. Select how the drive reacts to a fieldbus communication break with the parameter *50.02 Comm loss func*, Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 Comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for parameters *50.04…50.11*.
6. Set the FENA configuration parameters in group 51. At the minimum, set parameters 51.01 FBA A type as Ethernet, *51.02 Protocol/Profile* as *PNIO ABB Pro (11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24.
7. Define the process data transferred to and from the drive in the FBA-01 configuration parameter groups 52 and 53.
   Note! The adapter module sets the Status word and actual value automatically in parameters *52.01* and *52.02*, and Control word and reference in parameters *53.01* and *53.02*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Refresh*.
9. Set the relevant drive control parameters to control the drive according to the application.

ACS850 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	Fba enable	Enable	Activates fieldbus module
51.01	FBA A type	Ethernet	Fieldbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
10.01	Ext1 start func	FB	Fieldbus interface as source for start and stop
10.10	Fault reset sel	P.02.22 bit 8	Fieldbus interface as source for fault reset
(19.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.
21.01	Speed ref1 sel	FBA ref1	Fieldbus interface as source for

			speed reference

ACS850 More parameters read from drive to PLC

Parameter	Description	Setting	Comment
52.03 .. 52.12	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0104 = actual current; 0117 = motor temp 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 53)

ACS850 More parameters written from PLC to drive

Parameter	Description	Setting	Comment
53.03 .. 53.12	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2202 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 53)

Configuring ACS880 drives

Starting ACS880 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA A Enable* to *Enable*.
3. Select how the drive reacts to a fieldbus communication break with parameter *50.02 FBA A comm loss func*. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. Define the time between communication break detection and the selected action with parameter *50.03 FBA A comm loss t out*.
5. Select application-specific values for the rest of the parameters in group 50, starting from *50.04*.
6. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02 Node address* and the communication profile in *51.05* = 1 (ABB drives).
7. Define the process data transferred to and from the drive in FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01* and *53.01*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Configure*.
9. Save the valid parameter values to permanent memory by setting parameter *96.07 Param save* to **Save**.
10. Set the relevant drive control parameters to control the drive according to the application.

ACS880 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA A Enable	Enable	Activates fieldbus module
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.05	PROFILE	1	Communication profile ABB drives
52.02	FBA data in2	Act1 16bit (5)	Actual speed as Data Word 2 from the drive
53.02	FBA data out2	Ref1 16bit (2)	Speed reference as Data Word 2 to the drive
51.27	FBA par refresh	Configure	Updates fieldbus settings (groups 50 to 57)
20.01	Ext 1 commands	Fieldbus A	Fieldbus interface as source for start and stop
22.11	Speed ref1 selection	FB A ref1	Fieldbus interface as source for speed reference
31.11	Fault reset selection	P.06.01 bit 7	Fieldbus interface as source for fault reset
(46.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACS880 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA A Enable* to *Enable*.
3. With parameter *50.02 FBA A comm loss func*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 FBA A comm loss t out*, define the time between communication break detection and the selected action.
5. Select application-specific values for the rest of the parameters in group 50, starting from *50.04*.
6. Set the FENA configuration parameters in group 51. At the minimum, set parameters *51.01 FBA A type* as Ethernet, *51.02 Protocol/Profile* as *PNIO ABB Pro (11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24).
7. Define the process data transferred to and from the drive in FBA-01 configuration parameter groups 52 and 53.
8. **Note:** The adapter module sets the Status word and Control word automatically in parameters *52.01* and *53.01*.
9. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA par refresh* to *Configure*.
10. Save the valid parameter values to permanent memory by setting parameter *96.07 Param save* to *Save*.
11. Set the relevant drive control parameters to control the drive according to the application.

ACS880 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA A Enable	Enable	Activates fieldbus module
51.01	FBA A ype	Ethernet	Filedbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
20.01	Ext 1 commands	Fieldbus A	Fieldbus interface as source for start and stop
22.11	Speed ref1 selection	FB A ref1	Fieldbus interface as source for speed reference
31.11	Fault reset selection	P.06.01 bit 7	Fieldbus interface as source for fault reset
(46.01)	Speed scaling	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be

31

			less or equal to drive parameter max speed/frequency.

ACS880 More parameters read from drive to PLC (optional)

Parameter	Description	Setting	Comment
52.03 .. 52.12	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0107 = actual current; 3501 = motor est. temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 57)

ACS880 More parameters written from PLC to drive (optional)

Parameter	Description	Setting	Comment
53.03 .. 53.12	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2312 = acceleration time 1
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 57)

Configuring ACSM1 drives

Starting ACSM1 drives for PROFIBUS DP

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA ENABLE* to *Enable*.
3. With parameter *50.02 COMM LOSS FUNC*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 COMM LOSS T OUT*, define the time between communication break detection and the selected action.
5. Select application-specific values for parameters *50.04…50.11*.
6. Set the FPBA-01 configuration parameters in group 51. At the minimum, set the required node address in parameter *51.02* and the communication profile in *51.05* = 1 (ABB drives).
7. Define the process data transferred to and from the drive in the FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01* and *53.01*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
9. Set the relevant drive control parameters to control the drive according to the application.

ACSM1 Minimum required parameter settings for PROFIBUS DP

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA ENABLE	Enable	Activates fieldbus module
51.02	NODE ADDRESS	[PB NODE]	PROFIBUS DP node address of the drive
51.05	PROFILE	1	Communication profile ABB drives
52.01	FBA DATA IN1	4	Status word as Data Word 1 from the drive
52.02	FBA DATA IN2	5	Actual speed as Data Word 2 from the drive
53.01	FBA DATA OUT1	1	Control word as Data Word 1 to the drive
53.02	FBA DATA OUT2	2	Speed reference as Data Word 2 to the drive
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 53)
10.01	EXT1 START FUNC	FBA	Fieldbus interface as source for start and stop
24.01	SPEED REF1 SEL	FBA REF1	Fieldbus interface as source for speed reference
10.08	FAULT RESET SEL	P.02.12 bit 8	Fieldbus interface as source for fault reset
(25.02)	SPEED SCALING	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

Starting ACSM1 drives for PROFINET

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter *50.01 FBA ENABLE* to *Enable*.
3. With parameter *50.02 COMM LOSS FUNC*, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter *50.03 COMM LOSS T OUT*, define the time between communication break detection and the selected action.
5. Select application-specific values for parameters *50.04…50.11*.
6. Set the FENA configuration parameters in group 51. At the minimum, set parameters *51.01 FBA A type* as *Ethernet*, *51.02 Protocol/Profile* as *PNIO ABB Pro (11)*, *51.04 IP configuration* as *Static*, and required IP address from parameter *51.05* to parameter *51.08* and *51.09 Subnet CDR* as 24.
7. Define the process data transferred to and from the drive in the FBA-01 configuration parameter groups 52 and 53.
   Note: The adapter module sets the Status word and Control word automatically in parameters *52.01* and *53.01*.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter *51.27 FBA PAR REFRESH* to *REFRESH*.
9. Set the relevant drive control parameters to control the drive according to the application.

ACSM1 Minimum required parameter settings for PROFINET

The minimum required parameter settings based on factory default settings.

Parameter	Description	Setting	Comment
50.01	FBA ENABLE	Enable	Activates fieldbus module
51.01	FBA A ype	Ethernet	Filedbus type
51.02	Protocol/Profile	PNIO ABB Pro (11)	Communication profile ABB drives
51.04	IP Configuration	Static (0)	-
51.05 ... 51.08	IP address	192.xxx.xx.xx or any other required IP address.	-
51.09	Subnet CDR	24	24 For subnet mask 255.255.255.0
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 51 to 55)
52.01	FBA DATA IN1	4	Status word as Data Word 1 from the drive
52.02	FBA DATA IN2	5	Actual speed as Data Word 2 from the drive
53.01	FBA DATA OUT1	1	Control word as Data Word 1 to the drive
53.02	FBA DATA OUT2	2	Speed reference as Data Word 2 to the drive
10.01	EXT1 START	FBA	Fieldbus interface as source for

	FUNC		start and stop
24.01	SPEED REF1 SEL	FBA REF1	Fieldbus interface as source for speed reference
10.08	FAULT RESET SEL	P.02.12 bit 8	Fieldbus interface as source for fault reset
(25.02)	SPEED SCALING	[Scale max]	Max speed/frequency scaling value (used in function block/visualization input *SPEED_REF_MAX*). Must be less or equal to drive parameter max speed/frequency.

ACSM1 More parameters read from drive to PLC (optional)

Parameter	Description	Setting	Comment
52.03 .. 52.12	Actual value or parameter of drive	GGii	GG = parameter group, ii = parameter index example, 0104 = actual current; 0117 = motor temp
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 53)

ACSM1 More parameters written from PLC to drive (optional)

Parameter	Description	Setting	Comment
53.03 .. 53.12	Parameter of drive	GGii	GG = parameter group, ii = parameter index example, 2503 = acceleration time
51.27	FBA PAR REFRESH	REFRESH	Updates fieldbus settings (groups 50 to 53)

SIMATIC Manager

You can configure S7-300 and S7-400 series PLC using SIMATIC Manager. This chapter provides the examples for configuring S7-300 series PLC.

Setting PG/PC interface

Set PG/PC interface for online connection between PC and PLC (Ethernet connection is used in this example). For Ethernet connections, set the IP address of the PC within the same subnet as the PLC. Default IP address of a Siemens PLC is *192.168.0.1*, so set the IP address of your Ethernet port between *192.168.0.2* and *192.168.0.255* (or change the IP address of the PC). See example below (Windows 7 environment).
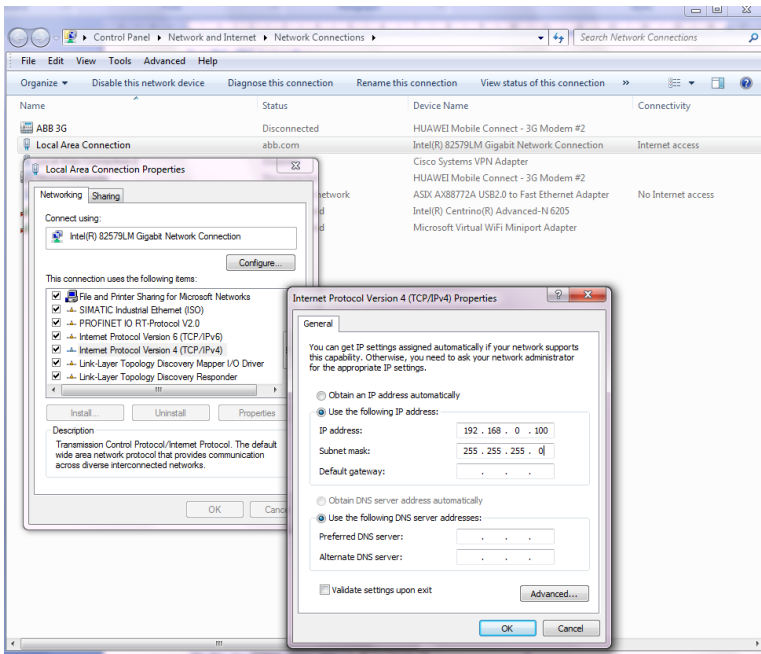
Figure 1 IP address

In the Options menu, click **Set PG/PC Interface** and choose the connected interface and click **OK**.
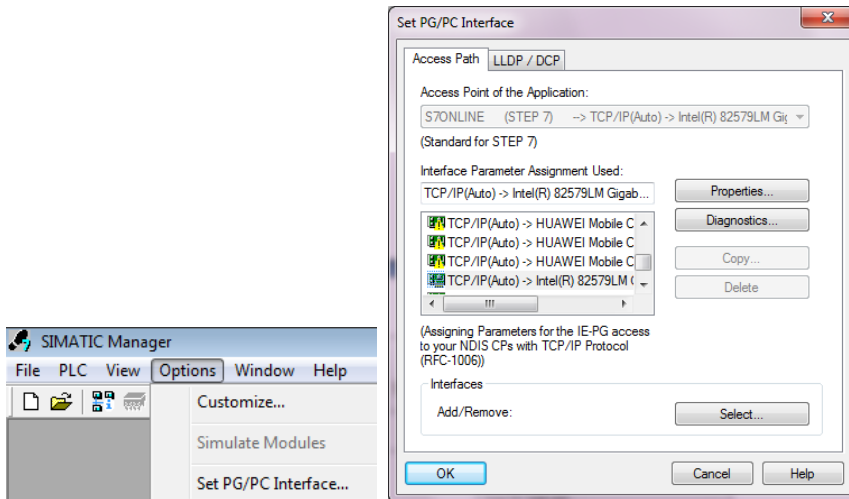
Note! For direct Ethernet connections (no routing), use *TCP/IP(Auto) -> XXXXXXXXXXXXXXX*.

Hardware configuration

In this example, a CPU 319F-3 PN/DP with integrated PROFIBUS DP and PROFINET IO interface is used. An ABB ACS850 drive with FPBA-01 PROFIBUS DP module is connected to the PLC via the PROFIBUS DP interface.

Starting a new project

Start a new project by choosing **New** from the **File** menu and give name and location to the project. Insert a station to the project by right-clicking the project name and choosing **Insert New Object** -> **SIMATIC 300 Station**.
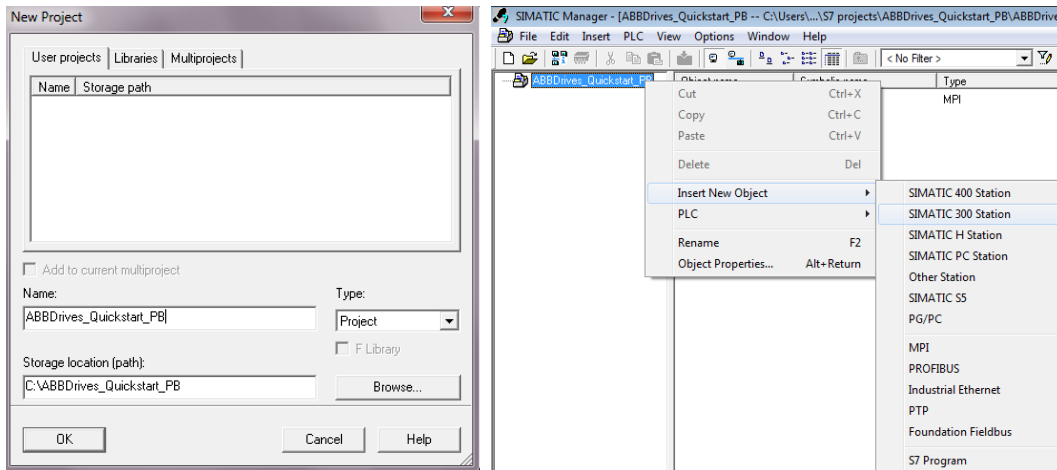
Expand the project tree, click the **SIMATIC** station and double-click **Hardware**. The HW Config window is displayed.
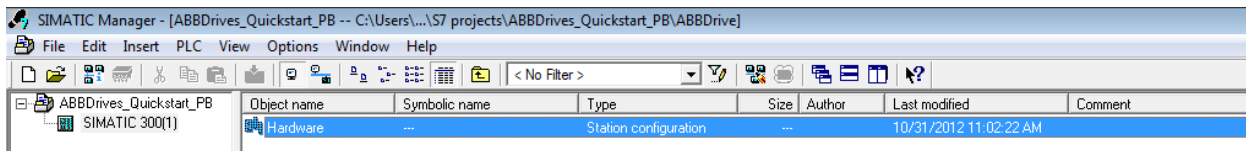
Figure 4 Hardware configuration

Adding Rack and CPU to the hardware

1. Use mouse drag and drop function to add a **Rail** for the SIMATIC station.
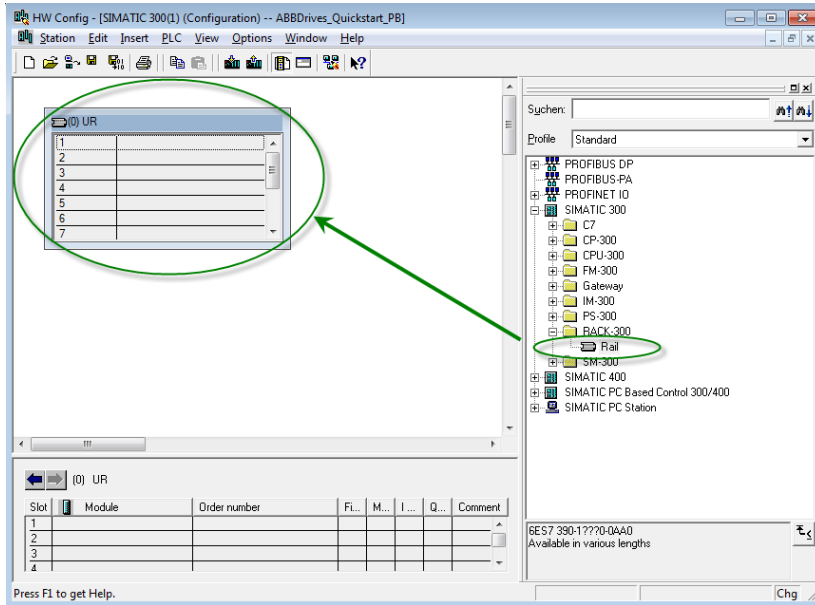
Figure 5 Rail

2. Similarly, drag and drop actual CPU type to slot 2 of the rack.
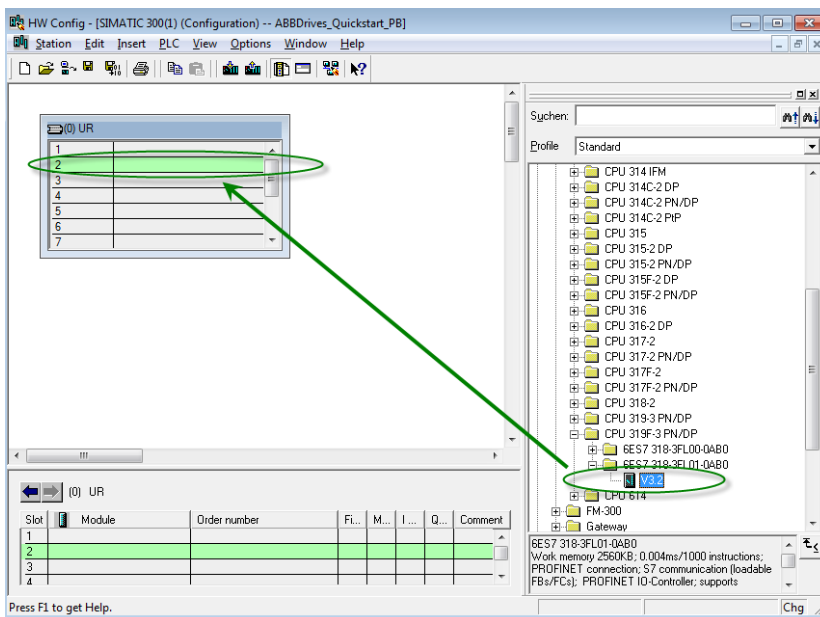
Figure 6 CPU type

3. In the **Ethernet interface** pop-up window, click **New** (1) and then click **OK** (2) -> **OK** (3) to activate the Ethernet connection.
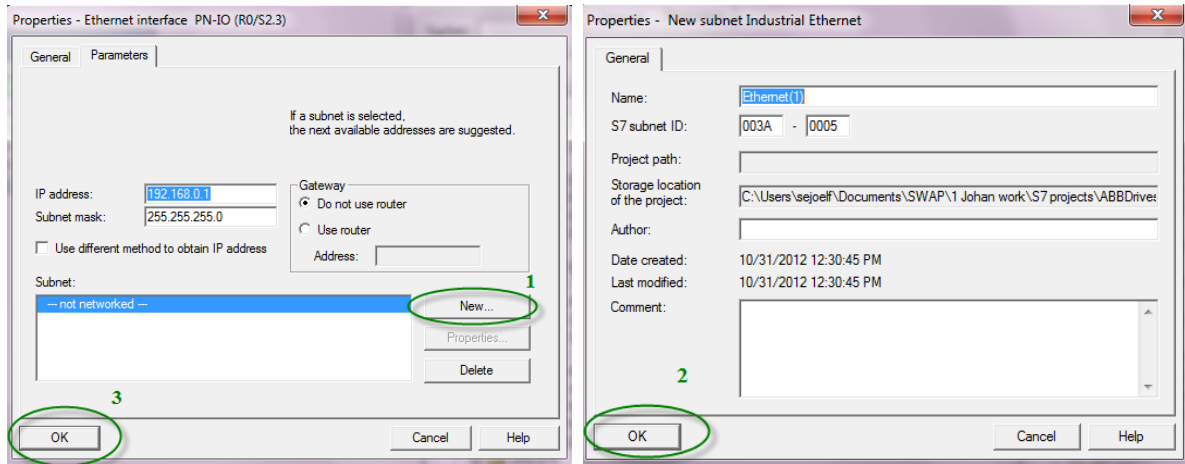
Figure 7 Ethernet interface

If the PROFIBUS interface window pops up, press **New** (1) and then click **OK** (2) -> **OK** (3) to activate the PROFIBUS connection. Otherwise follow instructions below these pictures.
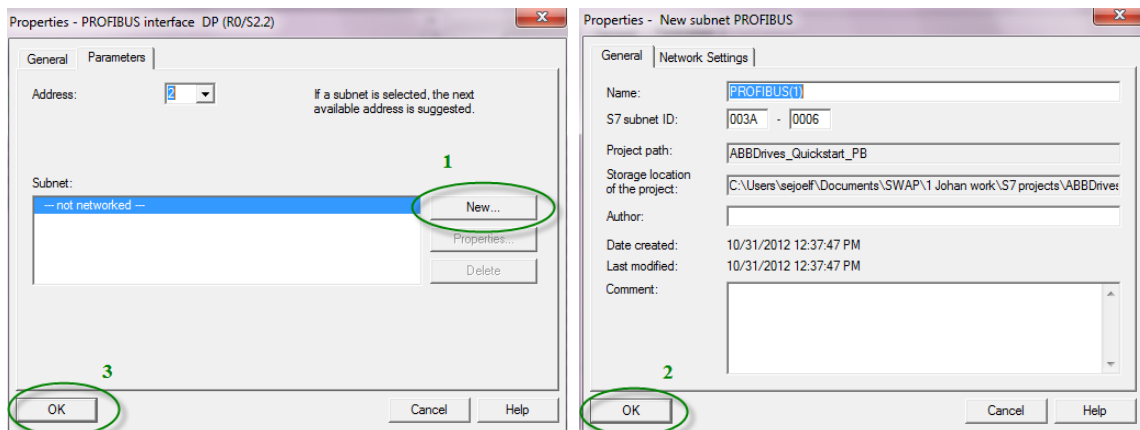
Figure 8 PROFIBUS properties

For PLCs where the PROFIBUS interface window does not appear automatically, double-click **MPI/DP** (or DP). In the pop-up window, change Interface to **PROFIBUS**. A new pop-up window opens. Click **New** to activate the PROFIBUS DP connection and then click **OK** until all pop-up windows are closed.
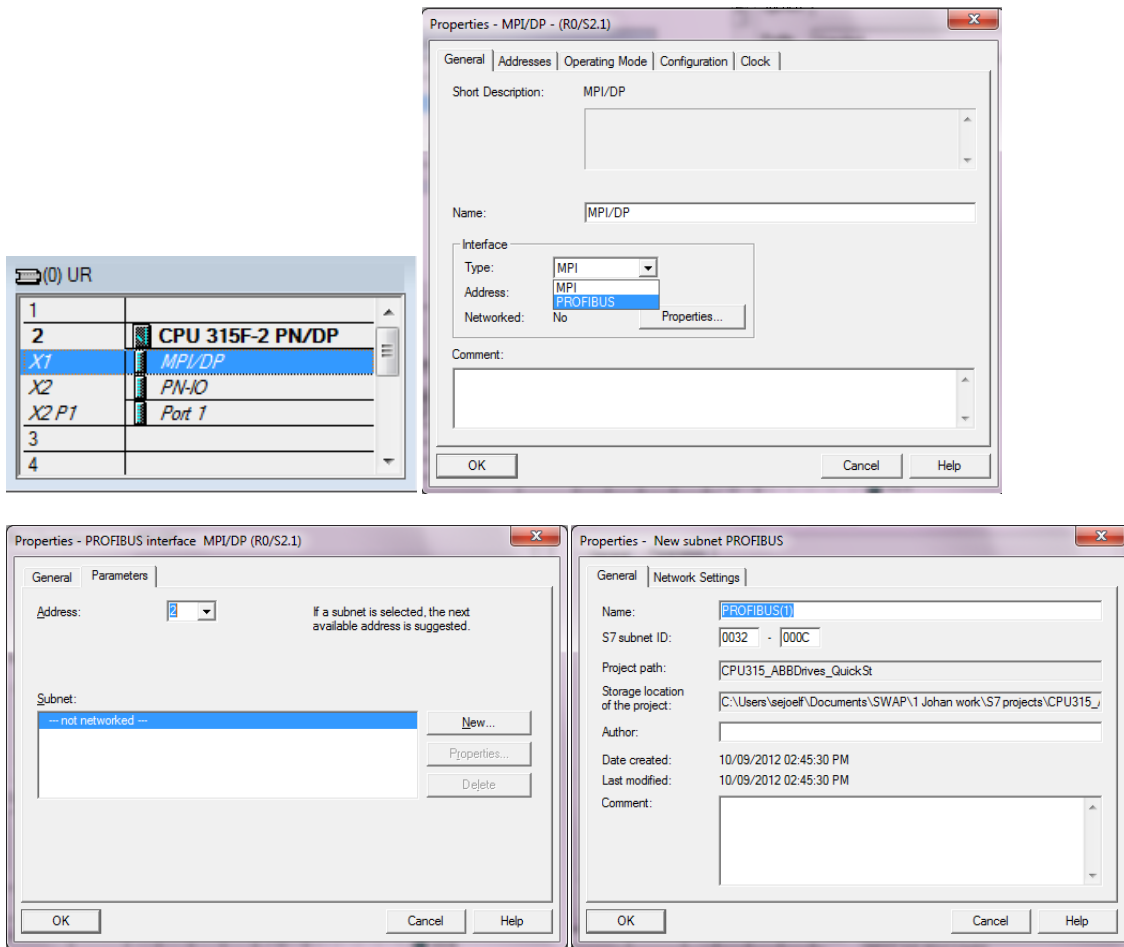
Figure 9 MPI/DP

Installing GSD files for ABB drives

GSD files for ABB drives can be found in *www.abb.com/drives* or from this package. Install the GSD files (Options -> Install GSD File). Find the catalog where the GSD files are placed by clicking the **Browse** button, highlight the GSD file(s) and click **Install**. When the files are installed, click **Close**. The following GSD files are available from the ABB website.

- ABB0959.gsd for FPBA-01 PROFIBUS DP-V0
- ABB10959.gsd for FPBA-01 PROFIBUS DP-V1 (used in this example)
- ABB_0812.gsd for RPBA-01 PROFIBUS DP-V0
- ABB10812.gsd for RPBA-01 PROFIBUS DP-V1
- GSDML-V2 31-ABB-FENA-20150120.xml for PROFINET
- GSDML-V2 31-ABB-FENA-20140901.xml for PROFINET
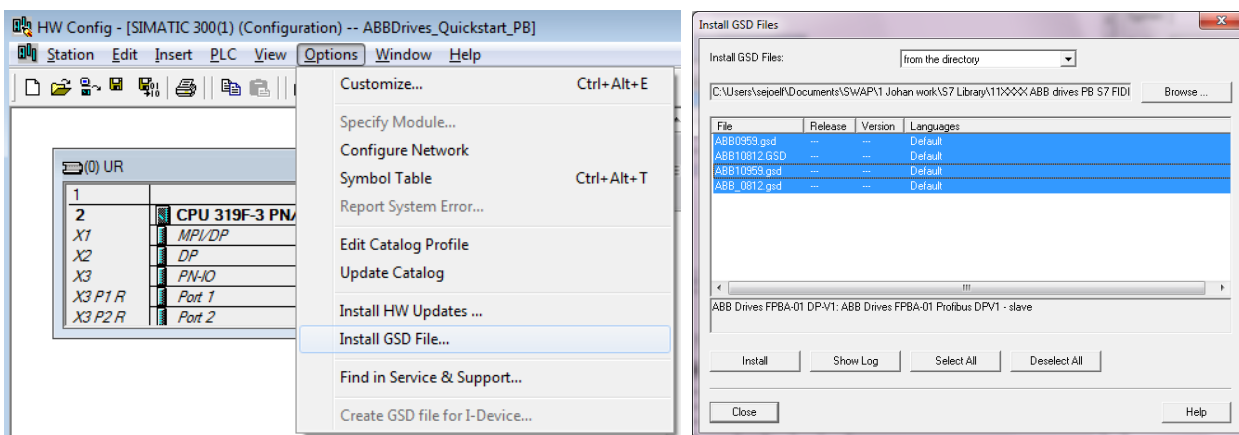
Figure 10 Installing GSD file

Adding drives to PROFIBUS DP line

After the GSD file installation, new drive PROFIBUS DP modules appear in the hardware library. Drag and drop according to your actual fieldbus module type and desired DP mode (V0 or V1) to the PROFIBUS DP line. Set the PROFIBUS DP node address for the fieldbus module and click **OK**.
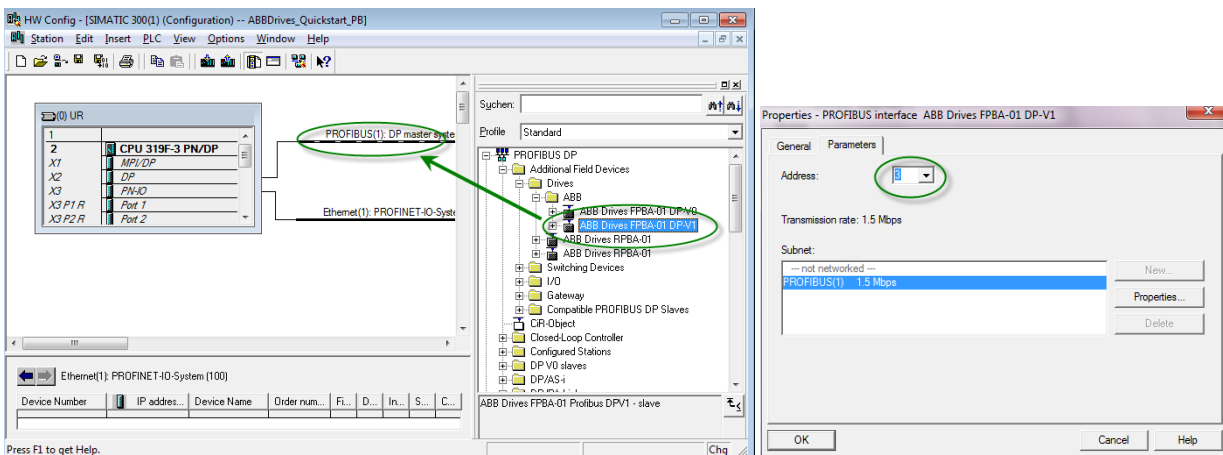
Figure 11 PROFIBUS DP

Select the drive/PROFIBUS DP node (1), drag and drop desired **PPO type** (2) to slot 1 of the module (3).

Note! Only PPO types with consistent data, example, PPO-06, 0 PKW + 10 PZD are supported. PPO types without consistent data, example, PPO-06, 0 PKW + (2+2+2+2+2) PZD or PPO-06, 0 PKW +NoCons. 10 PZD are not supported by *ABB_DRIVE_LIB*.

Note! If you are using DP-V1 protocol, only PPO types without Parameter data area are available (example, PPO-03, PPO-04, PPO-06).
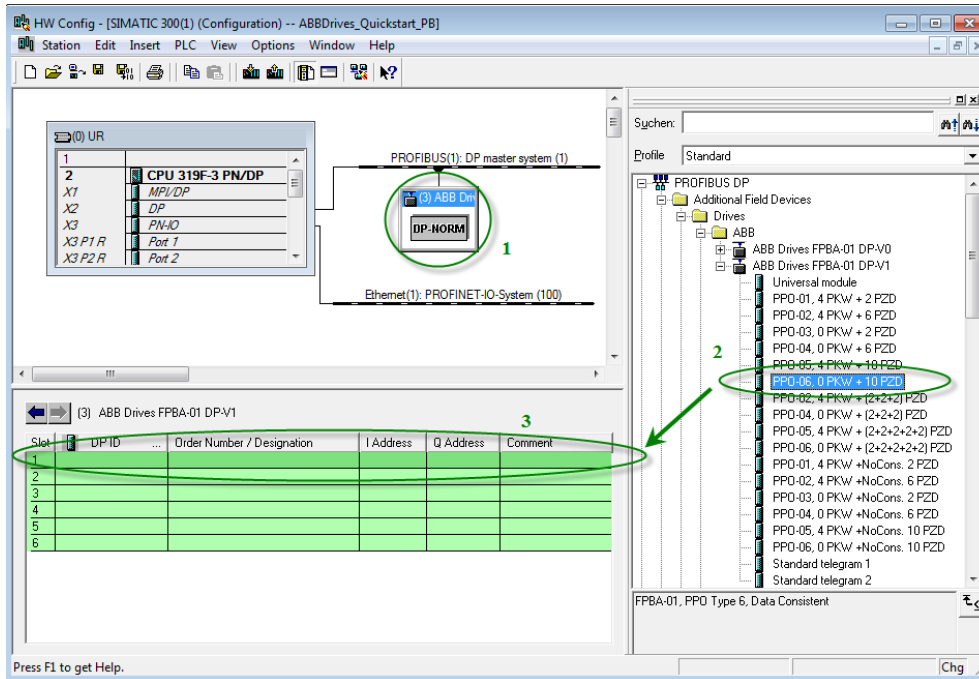
Figure 12 PPO type
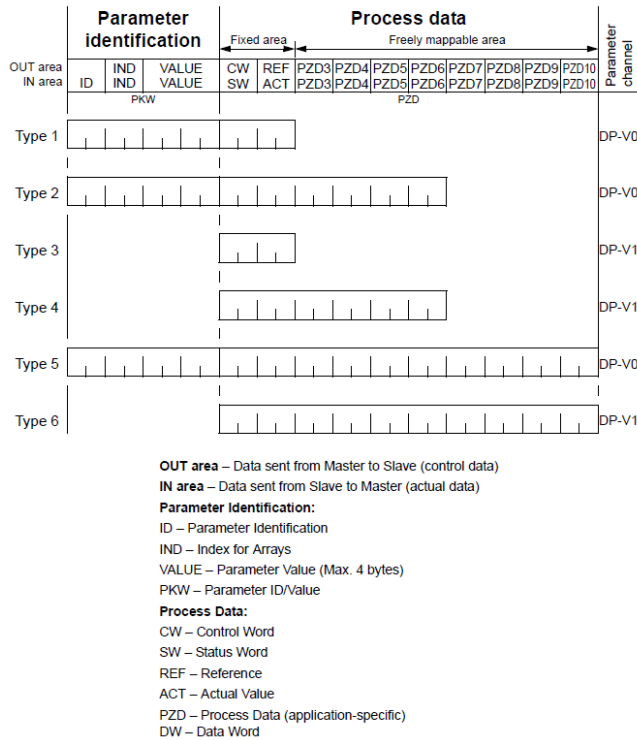
43

OUT area – Data sent from Master to Slave (control data)
IN area – Data sent from Slave to Master (actual data)
Parameter Identification:
ID – Parameter Identification
IND – Index for Arrays
VALUE – Parameter Value (Max. 4 bytes)
PKW – Parameter ID/Value
Process Data:
CW – Control Word
SW – Status Word
REF – Reference
ACT – Actual Value
PZD – Process Data (application-specific)
DW – Data Word

The drive automatically receives peripheral memory areas for data input and output (Process data that will be sent between the PLC and the drive). In this example, the 20 byte long peripheral memory area starts from 256. Change if needed by double-clicking PPO-XX of actual drive.
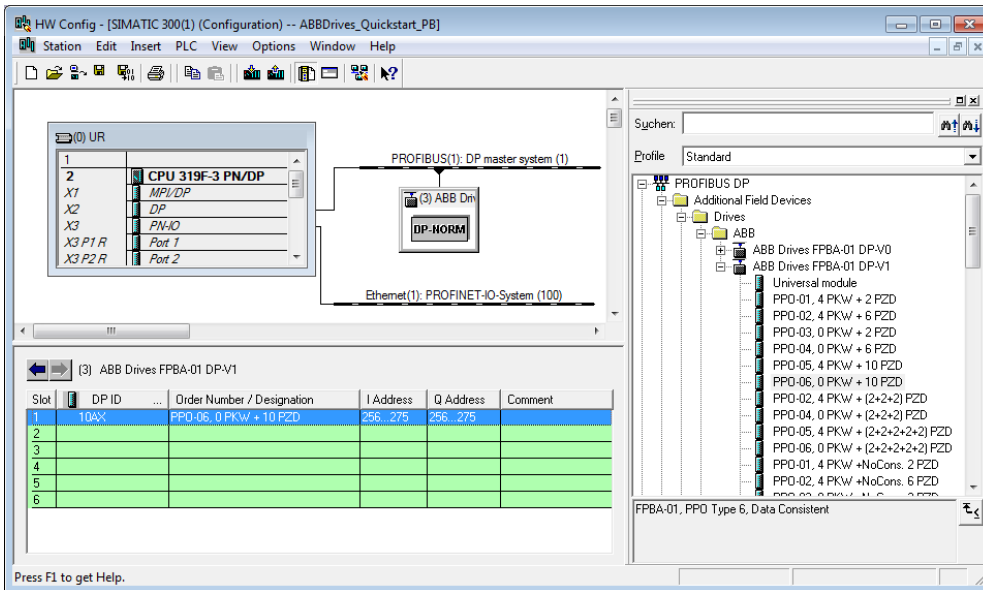
Figure 13 PPO type of actual drive

Right-click the drive (1) and choose **Object Properties** to change name (2) of the node if desired.
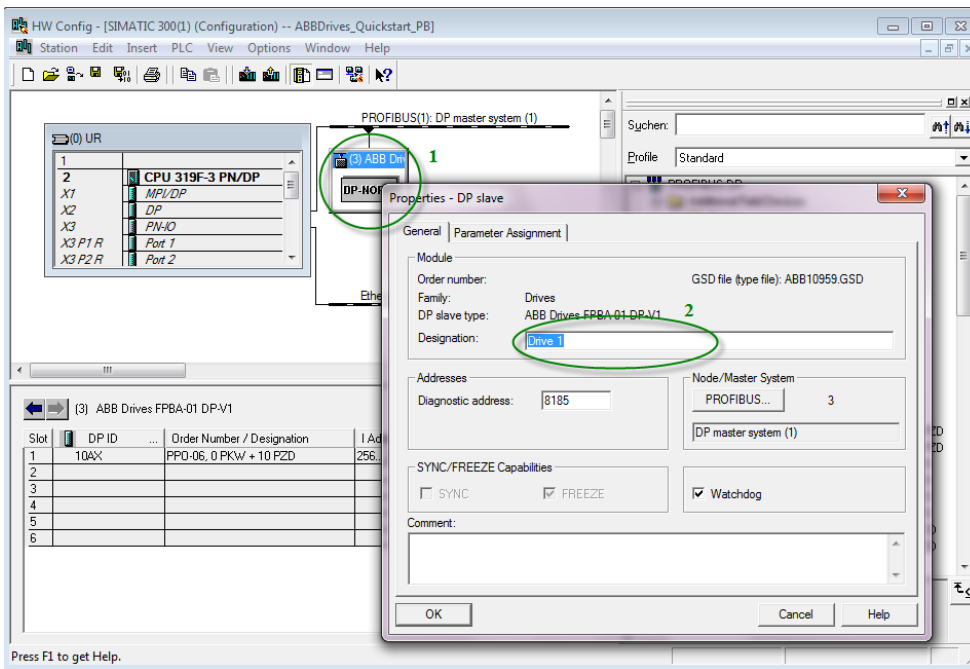
Figure 14 Object properties

Repeat the procedure to add more drives if needed and then click the **Save and Compile** button.
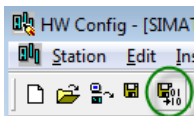
Figure 15 Save and compile

Adding drives to PROFINET

After the GSDML file installation, new drive PROFINET modules appear in the hardware library.

1. Drag and drop FENA-11 according to your actual fieldbus module type to the PROFINET.
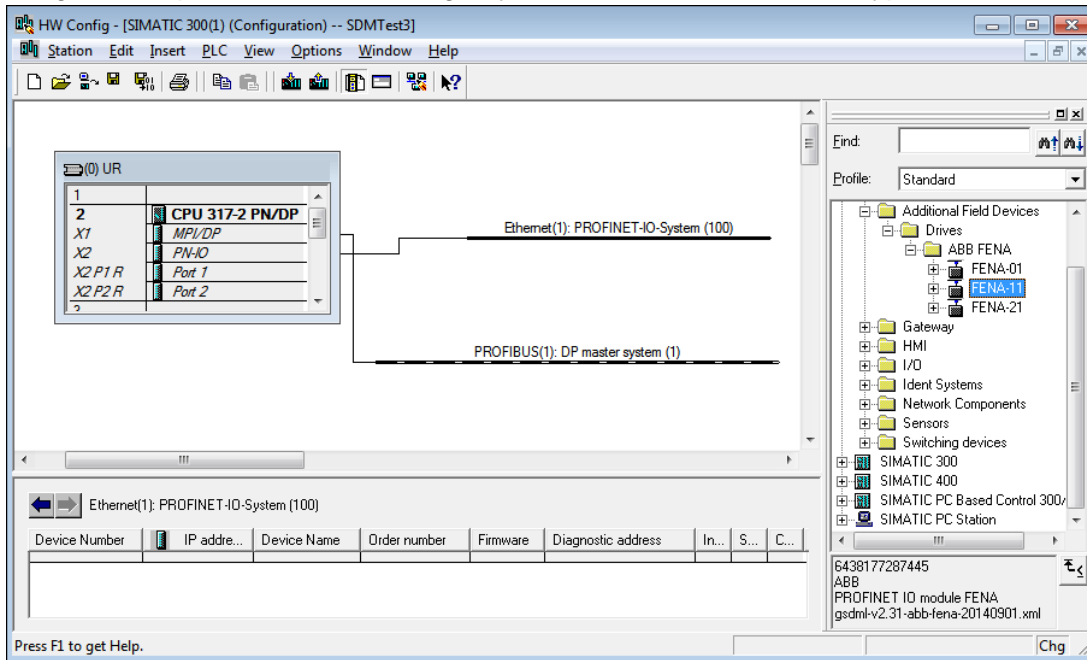
Figure 16 FENA-11

2. Double-click FENA drive to set the IP address.
3. In the Properties window, add Device name (FENA) and click Ethernet to set the PROFINET IP address for the fieldbus module and click **OK**.
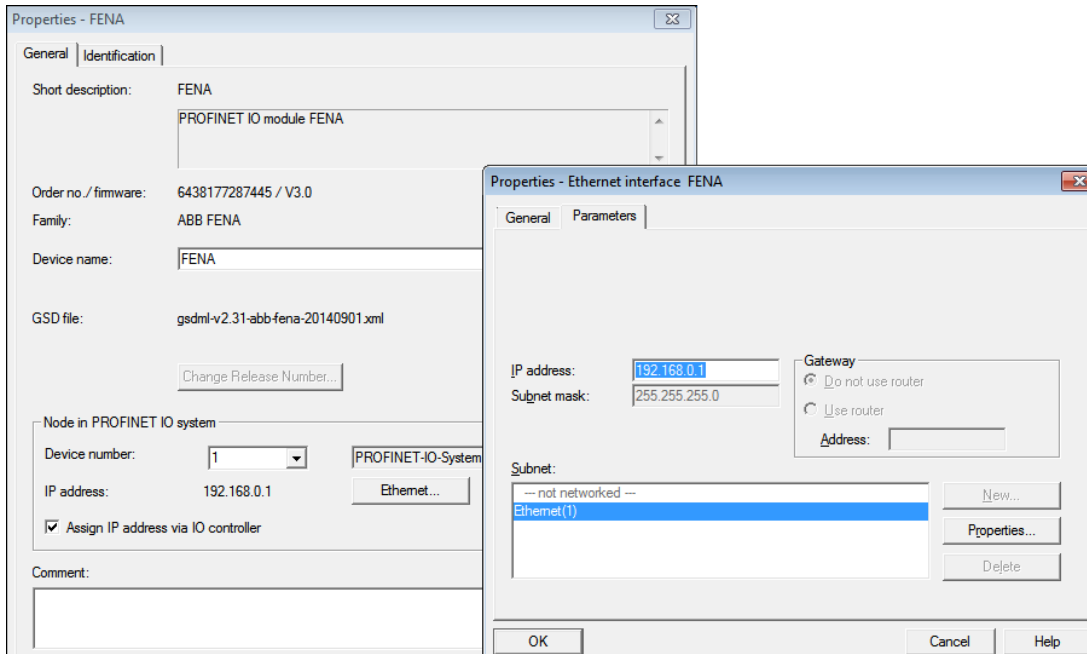
Figure 17 Properties

4. Select the PROFINET drive, and drag and drop the desired **PPO type (6)** to the slot 1 of the module.
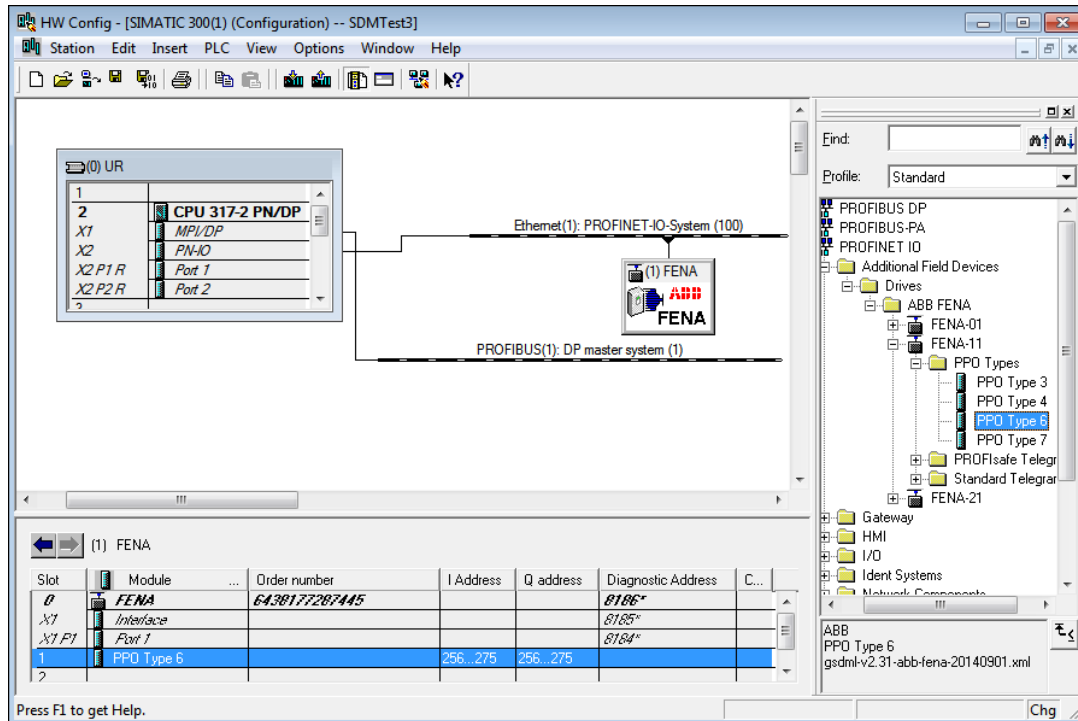
The drive automatically receives peripheral memory areas for data input and output (Process data that will be sent between the PLC and the drive). In this example, the 20 byte long peripheral memory area starts from 256. Change if needed by double-clicking PPO-XX of actual drive.

5. Repeat the procedure to add more drives if needed and then click the **Save and Compile** button.

Downloading Hardware configuration

Choose **Download** from the **PLC** menu. Select your target module (actual PLC) and click **OK**. Select the node address of the PLC (the IP address in case of Ethernet connection) and click **OK**.
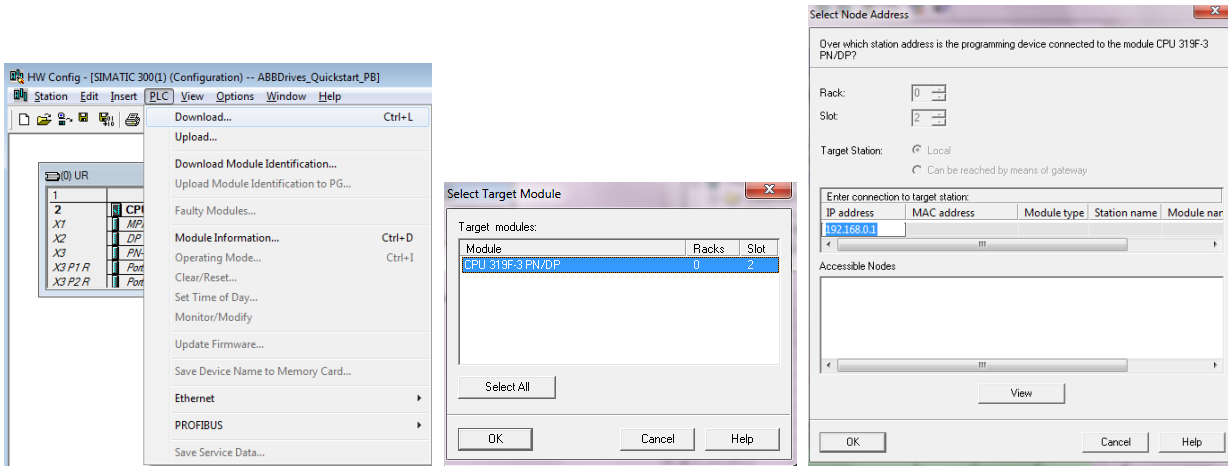
Figure 18 Downloading hardware configuration

If the PLC is in RUN mode, a **Stop Target Modules** message is displayed. Click **OK** and then click **Yes** in the Download pop-up window to set the PLC in RUN mode. Verify that the PLC is in RUN mode, for instance by checking that the CPU RUN led is green.
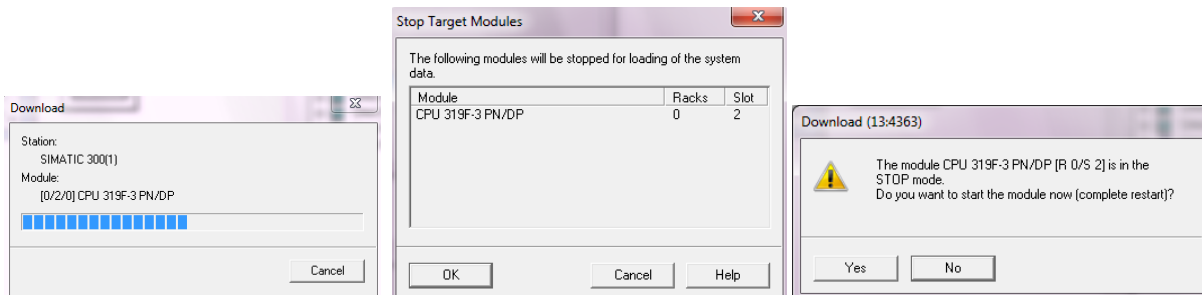
Figure 19 Download

PLC libraries for S7-300

Copying ABB_DRIVES_LIB blocks to the project

Retrieve the ABB drives function block library *ABB_DRIVE_LIB* from SIMATIC Manager. Choose **Retrieve** from the **File** menu and browse your zipped library. Click **Open** and then choose a suitable folder where to place the library, click **OK**.
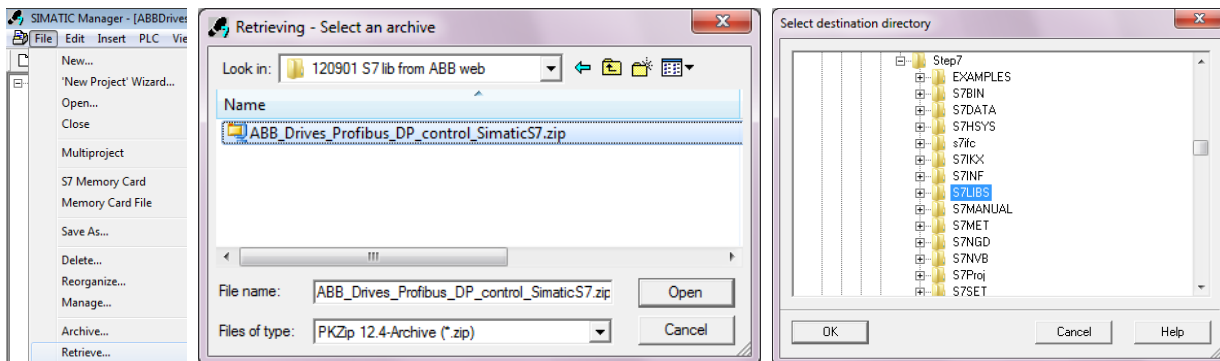
Figure 20 Retrieve library

Select all blocks from the library project, right-click and choose **Copy**. Open your project, right-click in the Blocks view and choose **Paste**.
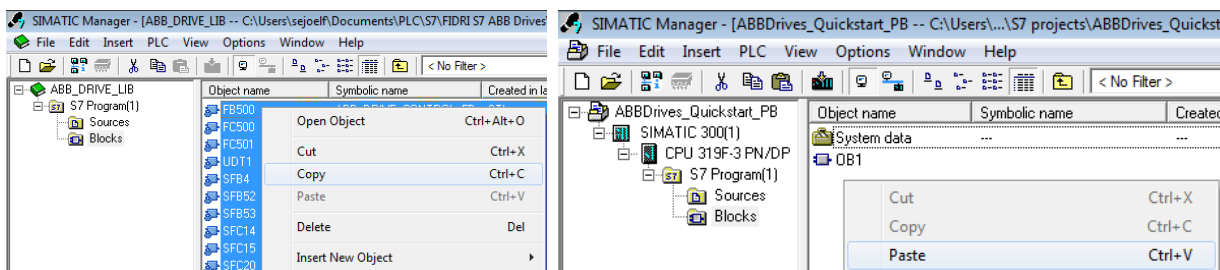
Figure 21 Copying libraries

Symbol Editor

Open the **Symbol Editor** from SIMATIC Manager and create symbols that you will connect to block inputs and outputs. In this example, some of the block inputs are left out since those values are set to fixed values.

Note! The variables for instance also created in a separate Data Block.
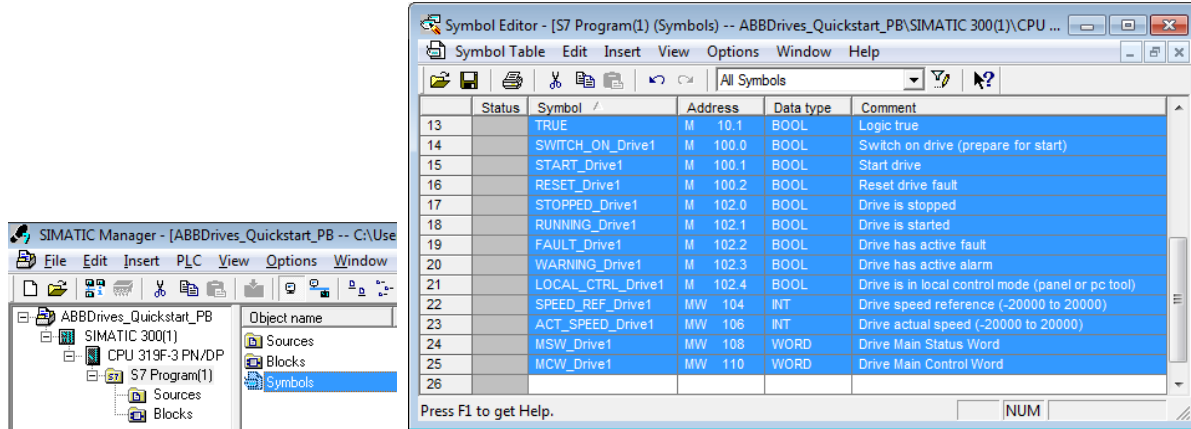
Figure 22 Symbol editor

FB500 ABB_DRIVE_CONTROL_FB

FB500 is used to control the drive (start, stop, reset, emergency stop, speed reference, etc.).

Note! The following descriptions are just an example, there are different ways to use the Control function block.

1. Open **OB1**, the default Start/Main block in Siemens CPUs.
2. Create memory bits for Logic true and Logic false. They will be used later in the program.
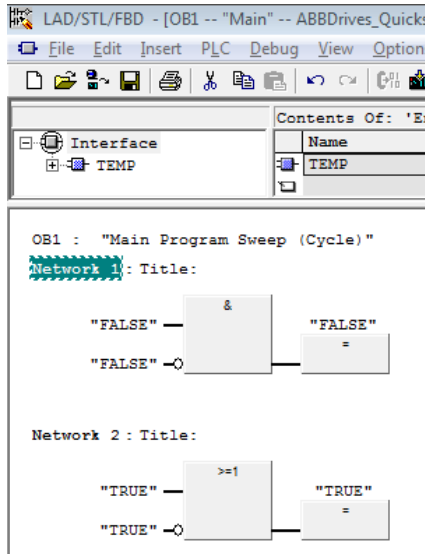
Figure 23 Memory bits

3. Right-click in the next free network and choose **Insert Empty box**. Write **FB500** in the block field to create an instance of FB500.
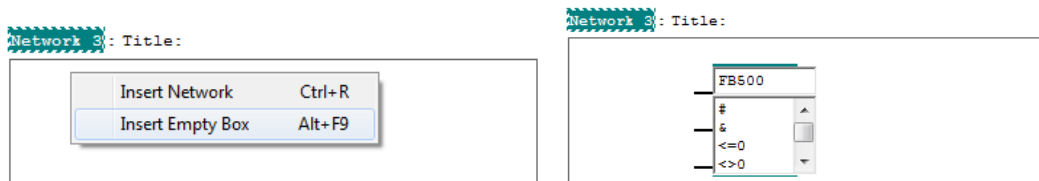
Figure 24 FB500

Create a **unique** Data Block for the control of your drive by writing DBXXX in the field over the block (DB101 has been chosen in this example, the available number of Data blocks vary with actual CPU model). Drive status and more will be stored in this Data Block. Since it is a new Data Block, you have to generate it by clicking **Yes** in the following pop-up window.
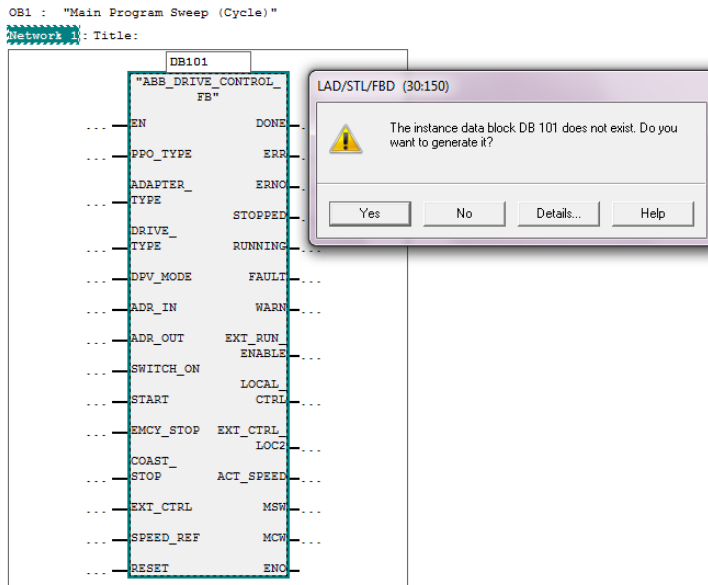
Figure 25 DB101 data blocks

Note! If you later add more drives to the program, make sure to create new unique Data blocks for them.

4. Connect the block inputs and outputs to variables according to your application.

Note! Select a block input or output and see more information in the Info tab (1, 2). See also comments to the block inputs/outputs in the table below.

Figure 26 Connecting block input/output

Block variables and data types

Block variable	Data type	Comment
EN	BOOL	Enabling block. FALSE = block code is not executed. TRUE or unconnected = block code is executed.
PPO_TYPE	INT	The PPO type. 1, 2, 3, 4, 5 or 6; 0 = not allowed.
ADAPTER_TYPE	INT	PROFIBUS module type: FPBA-01 PROFIBUS DP module connected in the drive. 1=FPBA (or FENA), 2=RPBA (or RETA).
DRIVE_TYPE	INT	Drive type: ACS800=1, ACSM1=2, ACS350=3, ACS355=4, ACS550=5, ACS850=6, ACS880=7, ACS580=8, ACS380=9.
DPV_MODE	BOOL	FALSE=DP-V0, TRUE=DP-V1 (or PROFINET).
ADR_IN	INT	The **Process** Data input of the drive, the start of the address range.
ADR_OUT	INT	The **Process** Data output of the drive, the start of the address range.
SWITCH_ON	BOOL	FALSE=Drive control switched off, TRUE=Drive control switched on. After an EMERGENCY STOP a new rising edge of SWITCH_ON is needed before next start. SWITCH_ON also needs to be active (TRUE) for resetting drive faults.
START	BOOL	FALSE=Ramp stop with deceleration time according to drive parameter, TRUE=Start. Drive start via fieldbus requires parameter setting in the drive.
EMCY_STOP	BOOL	FALSE=Emergency stop according to emergency stop deceleration time set in drive parameter, TRUE=Normal operation.
COAST_STOP	BOOL	FALSE=Normal operation, TRUE=Coast stop (drive releases control of the motor).
EXT_CTRL	BOOL	Selection of external control location EXT2. FALSE=EXT1, TRUE=EXT2. Shifting to EXT2 via fieldbus requires parameter setting in the drive.
SPEED_REF	INT	Speed reference value: -20000 to 20000. See chapter "Drive configuration" for scaling. Setting speed reference via fieldbus requires parameter setting in the drive.

RESET	BOOL	FALSE=No operation, TRUE =Reset drive fault.
DONE	BOOL	FALSE=Block execution not finished, TRUE=Block execution finished.
ERR	BOOL	FALSE=No error, TRUE=Error occurred during block execution.
ERNO	INT	Error code when ERR=TRUE, see SIMATIC online help for SFC14 or SFC15.
STOPPED	BOOL	FALSE=Drive is not stopped, TRUE=Drive is stopped.
RUNNING	BOOL	FALSE=Drive is not running, TRUE=Drive is running and following the speed reference value.
FAULT	BOOL	FALSE=No drive fault active, TRUE=Drive fault active.
WARN	BOOL	FALSE=No drive warning active, TRUE=Drive warning active.
EXT_RUN_ENABLE	BOOL	FALSE=No external run enable signal received in the drive, TRUE=External run enable signal received in the drive.
LOCAL_CTRL	BOOL	FALSE=Remote control (normal mode), TRUE=Local control (e.g. drive control panel or pc tool in local mode)
EXT_CTRL_LOC2	BOOL	Actual control place, FALSE=EXT1, TRUE=EXT2.
ACT_SPEED	INT	Drive actual speed: -20000 to 20000. See chapter "Drive configuration" for scaling.
MSW	WORD	Drive main status word. See actual fieldbus adapter manual for detailed description.
MCW	WORD	Drive main control word. See actual fieldbus adapter manual for detailed description.

Save your program.
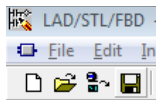
Figure 27 Save program

Setting the drive in standby mode

Set *EMCY_STOP* = *TRUE* (deactivate emergency stop) and after that *SWITCH_ON* = *TRUE* to set the drive in standby mode waiting for START command. To make the drive ready for start, *SWITCH_ON* always needs a positive edge after *EMCY_STOP* has been activated.
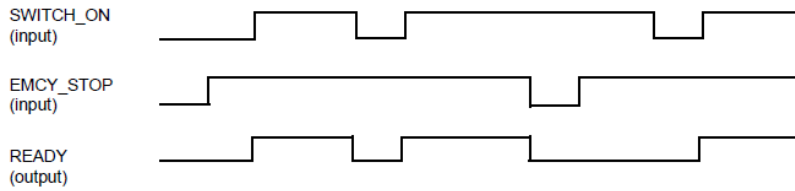
Figure 28 Standby mode

Setting the drive in run mode

Set the drive to the standby mode, *COAST_STOP* = *FALSE* and *START* = *TRUE* to start the drive. Set the desired reference value and the drive accelerates according to used acceleration ramp time to the set reference value. When START is set to FALSE, a restart is possible when the actual speed has reached zero. If flying restart is required, COAST STOP has to be used instead.
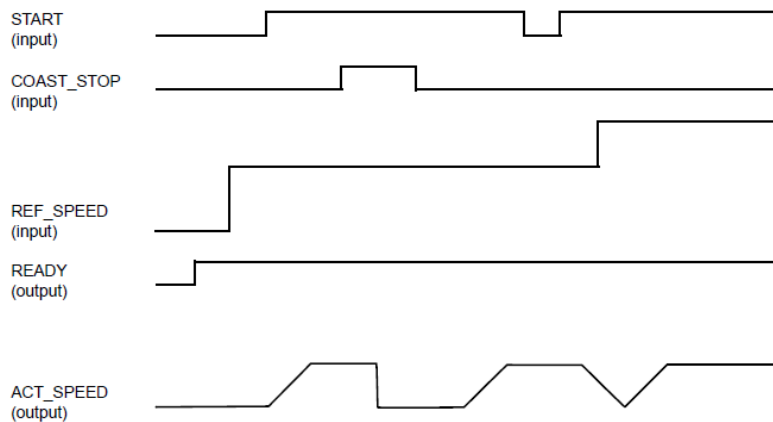
Figure 29 Run mode

Downloading a program and setting PLC in run mode

1. Open the block view in SIMATIC Manager. Select all blocks and choose **Download** from the **PLC** menu.
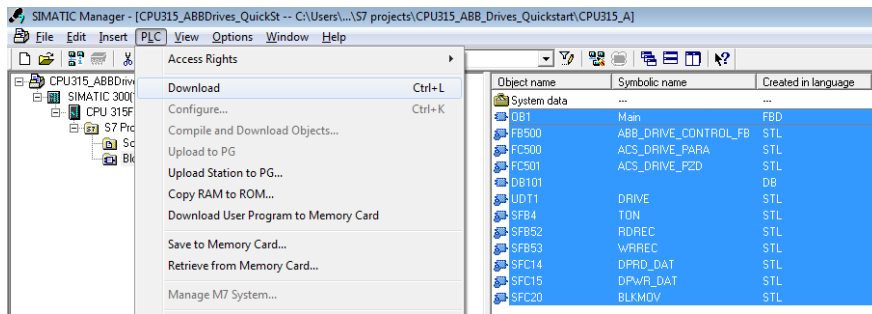
Figure 30 Downloading PLC

2. Set the physical dip switch of the PLC in position **RUN**. Choose **Operating Mode** from the PLC menu and check that Current Operating Mode is RUN, if not then choose and click preferred Restart mode.
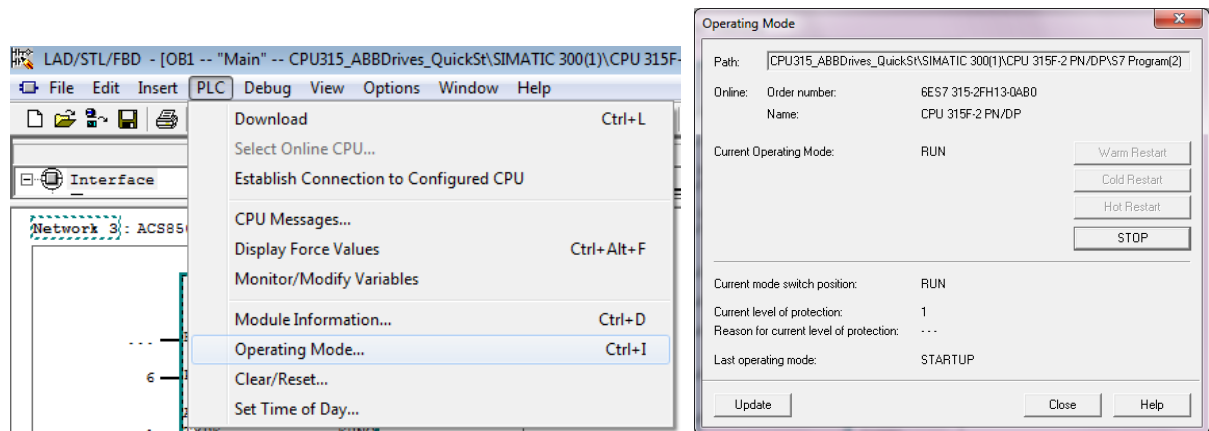
Figure 31 Operating mode

VAT table

To get an overview of your connected parameters, create a Variable Table.

1. In the SIMATIC Manager main window, Blocks view, choose **Insert New Object** and click **Variable Table**.
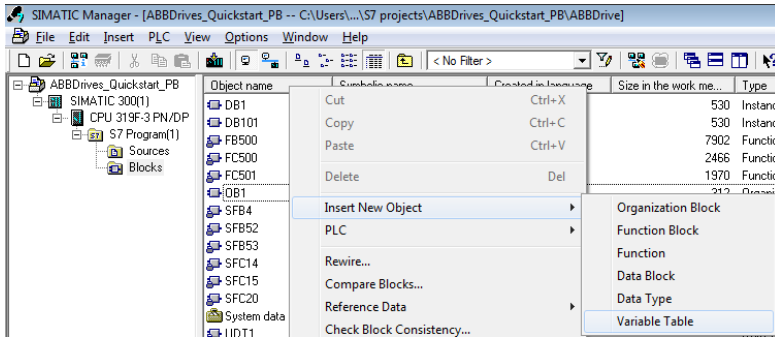
Figure 32 Variable table

2. In the Properties – Variable Table window, give the table a suitable name and click **OK**.
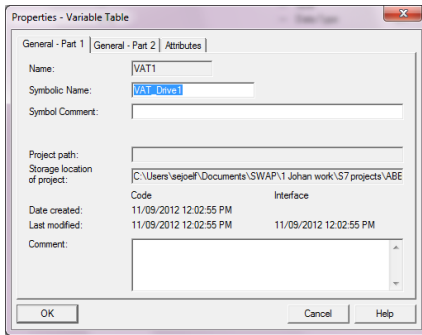
Figure 33 Properties - Variable table

3. Open the Variable table and add your preferred variables (it is possible to copy directly from the Symbol Editor). Click symbol (1) to display online values. Click symbol (3) to update any modified values (2).
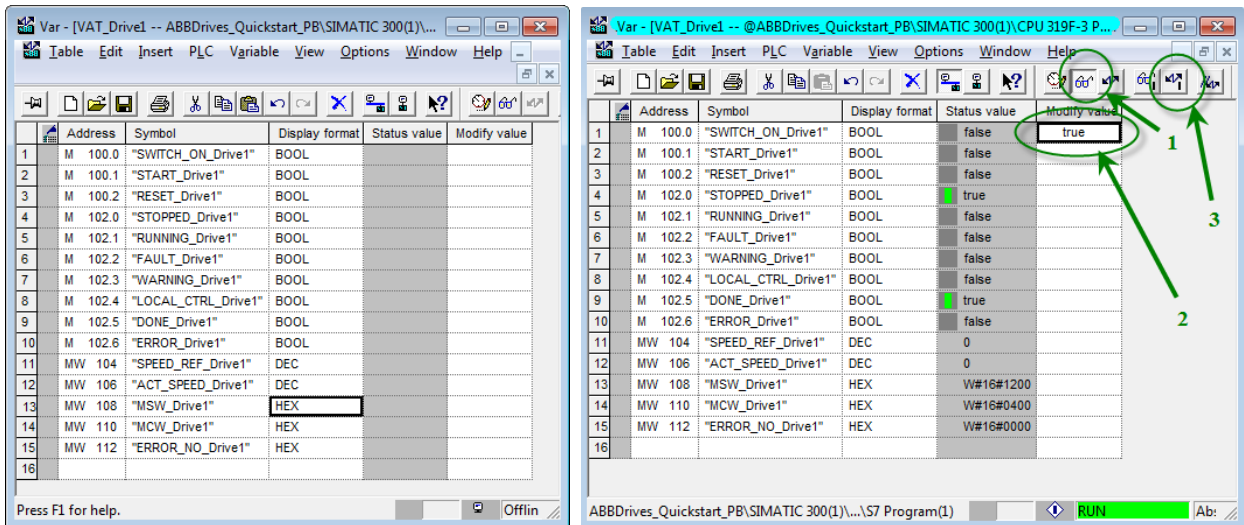
Figure 34 Variable table settings

FC501 ACS_DRIVE_PZD (optional)

FC501 is used to send additional process data between the PLC and the drive. Insert a new **Empty Box** to your program and name **FC501** (FC501 *ACS_DRIVE_PZD* is included in the ABB drives library **ABB_DRIVE_LIB**).
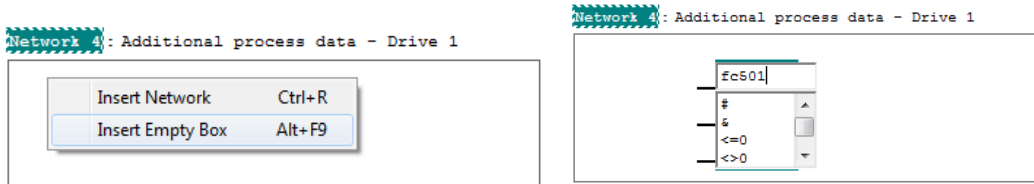
Figure 35 PLC logic - FC501

Depending on actual PPO type, a certain number of data words (PZDs) are exchanged. PPO types 1 and 3 have only 2 PZDs in each direction, so FC501 (ACS_DRIVE_PZD) is not useful for those types. PPO types 2 and 4 have 6 PZDs in each direction, so a part of FC501 (PZD3 to PZD6) is useful here. PPO types 5 and 6 have 10 PZDs in each direction, so the full range of FC501 is useful for those types.

Connect the FC501 block inputs and outputs. All block inputs and outputs need to be connected.

Block variable and data types for FC501

Block variable	Data type	Comment
PZD3_OUT	INT	Write PZD3 value to the drive
PZD4_OUT	INT	Write PZD4 value to the drive
PZD5_OUT	INT	Write PZD5 value to the drive
PZD6_OUT	INT	Write PZD6 value to the drive
PZD7_OUT	INT	Write PZD7 value to the drive
PZD8_OUT	INT	Write PZD8 value to the drive
PZD9_OUT	INT	Write PZD9 value to the drive
PZD10_OUT	INT	Write PZD10 value to the drive
PZD2_SCALED	REAL	Read PZD2 (actual speed / ACT) value from the drive, the default scaling value corresponds to -20 000 to 20 000 -> -100 to 100
PZD3_SCALED	REAL	Read PZD3 value from the drive, the default scaling value is 1 = no scaling
PZD4_SCALED	REAL	Read PZD4 value from the drive, the default scaling value is 1 = no Scaling
PZD5_SCALED	REAL	Read PZD5 value from the drive, the default scaling value is 1 = no Scaling
PZD6_SCALED	REAL	Read PZD6 value from the drive, the default scaling value is 1 = no Scaling
PZD7_SCALED	REAL	Read PZD7 value from the drive, the default scaling value is 1 = no Scaling
PZD8_SCALED	REAL	Read PZD8 value from the drive, the default scaling value is 1 = no Scaling
PZD9_SCALED	REAL	Read PZD9 value from the drive, the default scaling value is 1 = no Scaling
PZD10_SCALED	REAL	Read PZD10 value from the drive, the default scaling value is 1 = no scaling
Drive	BLOCK_DB	Instance Data Block. The drive variable is used for identifying to which drive FC501 *ACS_DRIVE_PZD* belongs. The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* must correspond to the variable FC501 *ACS_DRIVE_PZD* drive of the same drive.

In the example below, the Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* has been named DRIVE1 and an ACS800 with RPBA-01 PROFIBUS module has been used.

```
                    "DRIVE1"
            "ABB_DRIVE_CONTROL_
                     FB"
                                        "DRIVE1_
    ...  — EN                           RPBA_DB".
                            DONE —DONE
     6  — PPO_TYPE
                                        "DRIVE1_
              ADAPTER_                   RPBA_DB".
     2  — TYPE            ERR —ERR
```

```
                 "ACS_DRIVE_PZD"
                                        "DRIVE1_
    ...  — EN                           RPBA_DB".
                         PZD2_    PZD2_
                        SCALED —SCALED
  "DRIVE1_
  RPBA_DB".
  PZD3_OUT — PZD3_OUT                    "DRIVE1_
                                         RPBA_DB".
  "DRIVE1_              PZD3_    PZD3_
  RPBA_DB".            SCALED —SCALED
  PZD4_OUT — PZD4_OUT
                                         "DRIVE1_
  "DRIVE1_                               RPBA_DB".
  RPBA_DB".             PZD4_    PZD4_
  PZD5_OUT — PZD5_OUT  SCALED —SCALED

  "DRIVE1_                               "DRIVE1_
  RPBA_DB".                              RPBA_DB".
  PZD6_OUT — PZD6_OUT   PZD5_    PZD5_
                       SCALED —SCALED
  "DRIVE1_
  RPBA_DB".                              "DRIVE1_
  PZD7_OUT — PZD7_OUT                    RPBA_DB".
                        PZD6_    PZD6_
  "DRIVE1_             SCALED —SCALED
  RPBA_DB".
  PZD8_OUT — PZD8_OUT                    "DRIVE1_
                                         RPBA_DB".
  "DRIVE1_              PZD7_    PZD7_
  RPBA_DB".            SCALED —SCALED
  PZD9_OUT — PZD9_OUT
                                         "DRIVE1_
  "DRIVE1_                               RPBA_DB".
  RPBA_DB".             PZD8_    PZD8_
  PZD10_OUT — PZD10_OUT SCALED —SCALED

  "DRIVE1" — Drive                       "DRIVE1_
                                         RPBA_DB".
                        PZD9_    PZD9_
                       SCALED    
```

Figure 36 Data blocks of FB500

If you need to scale Process Data values, set the scaling values in the program before calling the FC501 *ACS_DRIVE_PZD* block. See example below where PZD3 has been scaled with the value 100.

```
              MOVE        "DRIVE1".
    ...  — EN             INTERNAL_
                          PZD.SCALE.
  1.000000e+     OUT —PZD3_SCALE
       002 — IN
              ENO
```
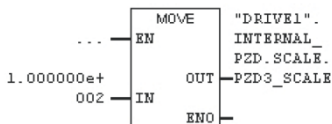
Figure 37 Process data values

Scalable variables and data types

Scalable variable	Data type	Variable to which the scaling value is entered
PZD2	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD2_SCALE
PZD3	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD3_SCALE
PZD4	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD4_SCALE
PZD5	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD5_SCALE
PZD6	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD6_SCALE
PZD7	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD7_SCALE
PZD8	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD8_SCALE
PZD9	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD9_SCALE
PZD10	REAL	"DRIVE1".INTERNAL_PZD.SCALE.PZD10_SCALE

Map the fieldbus process data parameters of the drive according to the application. See chapter *Drive configuration*. See example below from an ACS355 where Current (1.04), Torque (1.05), Power (1.06), DC bus voltage (1.07) are read from drive to PLC through PZD3 to PZD6, and Constant speed 1 to 4 (12.02 to 12.05) are written from PLC to drive through PZD3 to PZD6.

54 FBA DATA IN			
01 FBA DATA IN 1	4	0	9999
02 FBA DATA IN 2	5	0	9999
03 FBA DATA IN 3	104	0	9999
04 FBA DATA IN 4	105	0	9999
05 FBA DATA IN 5	106	0	9999
06 FBA DATA IN 6	107	0	9999
07 FBA DATA IN 7	0	0	9999
08 FBA DATA IN 8	0	0	9999
09 FBA DATA IN 9	0	0	9999
10 FBA DATA IN 10	0	0	9999
55 FBA DATA OUT			
01 FBA DATA OUT 1	1	0	9999
02 FBA DATA OUT 2	2	0	9999
03 FBA DATA OUT 3	1202	0	9999
04 FBA DATA OUT 4	1203	0	9999
05 FBA DATA OUT 5	1204	0	9999
06 FBA DATA OUT 6	1205	0	9999
07 FBA DATA OUT 7	0	0	9999
08 FBA DATA OUT 8	0	0	9999
09 FBA DATA OUT 9	0	0	9999
10 FBA DATA OUT 10	0	0	9999

Figure 38 Example ACS355

FC500 ACS_DRIVE_PARA (optional)

FC500 is used to read/write extra parameters between the PLC and the drive. Insert a new **Empty Box** to your program and name it FC500 (FC500 *ACS_DRIVE_PARA* is included in the ABB drives library *ABB_DRIVE_LIB*).

Figure 39 PLC logic - FC500

Connect the FC500 block inputs and outputs. All block inputs and outputs need to be connected.

Block variables and data types for FC500

Block variable	Data type	Comment
ADR_IN	INT	The beginning of the address range of the drive's: **Parameter Identification** input for DP-V0. **Process Data** input (or 0 "zero") for DP-V1 (or PROFINET).
ADR_OUT	INT	The beginning of the address range of the drive's: **Parameter Identification** output for DP-V0. **Process Data** output (or 0 "zero") for DP-V1 (or PROFINET).
READ	BOOL	Read the parameter value 0 -> 1 (executed on positive edge).
WRITE	BOOL	Write the parameter value 0 -> 1 (executed on positive edge).
PARAM_NUM	DINT	Read/written parameter: 3 numbers = group, 2 numbers = Index. For example, Par 20.06 = 2006.
VALUE_IN	DINT	Parameter value to be written.
DRIVE	BLOCK_DB	Instance Data Block. The drive variable is used for identifying to which drive FC501 *ACS_DRIVE_PZD* belongs. The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* must correspond to the variable FC501 ACS_DRIVE_PZD drive of the same drive.
DONE	BOOL	FALSE=Block execution not finished, TRUE=Block execution finished.
ERR	BOOL	FALSE=No error, TRUE=Error occurred during block execution.
ERNO	WORD	Error code when ERR=TRUE.
BUSY	BOOL	FALSE=No operation active, TRUE=Operation active.
PARAM_NUM_OUT	DINT	Handled parameter number: 3 numbers = group, 2 numbers = index; for example, Par 20.06 = 2006.
VALUE_OUT	DINT	Read parameter value.

In the example below, the Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* is named as DRIVE1 and an ACS800 with RPBA-01 PROFIBUS module has been used.

Note! The FB500 *ABB_DRIVE_CONTROL_FB* DPV_MODE variable defines the used protocol. You must set this variable correctly to make FC500 to work.
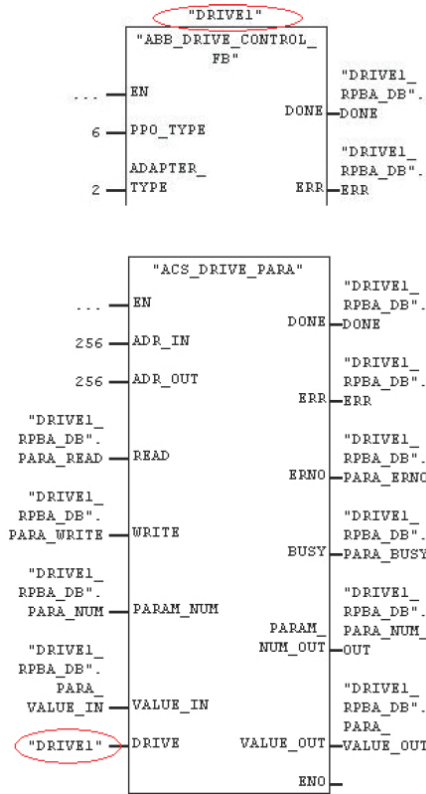
Figure 40 Example

TIA portal

You can configure S7-300, S7-400, S7-1200 and S7-1500 series PLC using TIA portal. This chapter provides the examples for configuring S7-300 and S7-1200 series PLC.

Configuring PC IP address

To configure PC IP address, follow these steps:

1. Open **Control Panel** -> **Network and Sharing Center** -> **Local Area Connection** in the PC.
2. In the Local Area Connection Status window, click **Properties**.
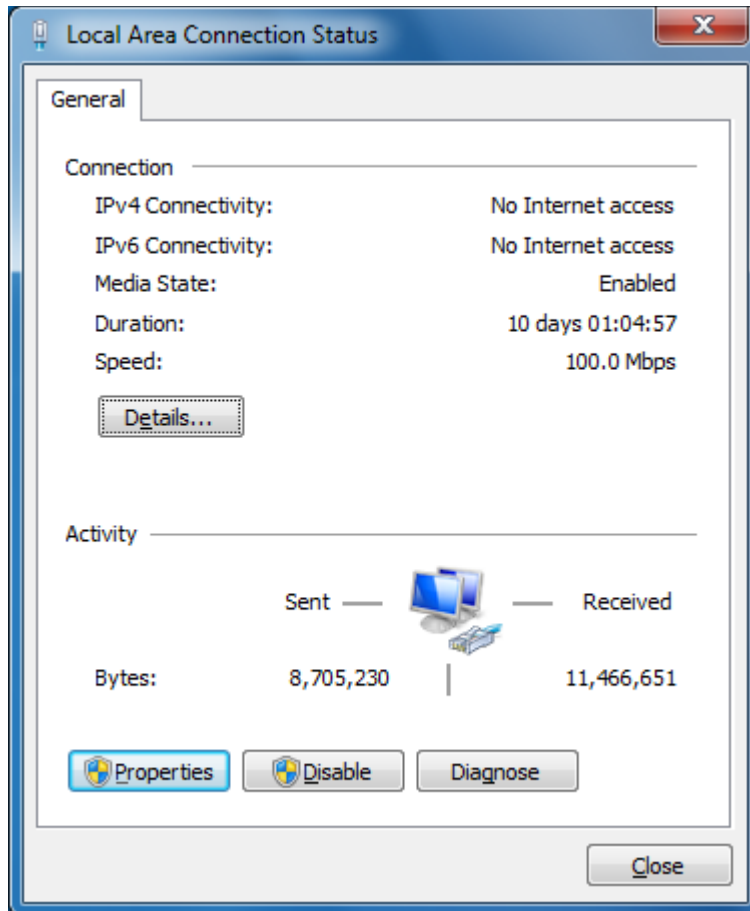
Figure 41 LAC

3. In the Properties window, select the required **Internet Protocol Version** (for example, Internet Protocol Version 4) and click **Properties**.
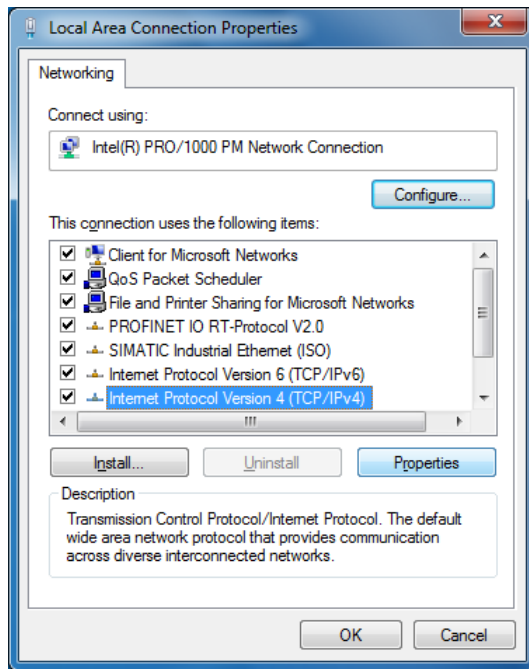
Figure 42 Internet protocol

4. Assign the required IP address and click **OK**.
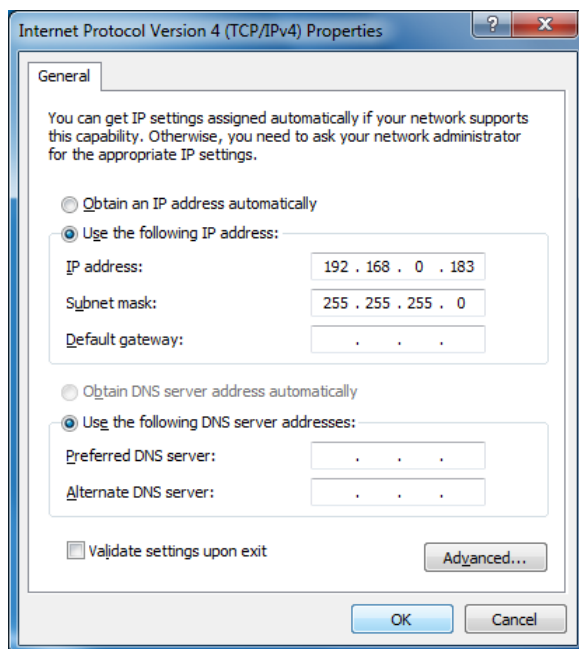
Figure 43 Assign IP address

Configuring S7-300 PLC

To configure TIA Portal with the PLC and ABB Drive libraries, follow these steps:

1. Launch **TIA Portal V11**.
2. Click **Create new project** and enter the project name and path and then click **Create**.
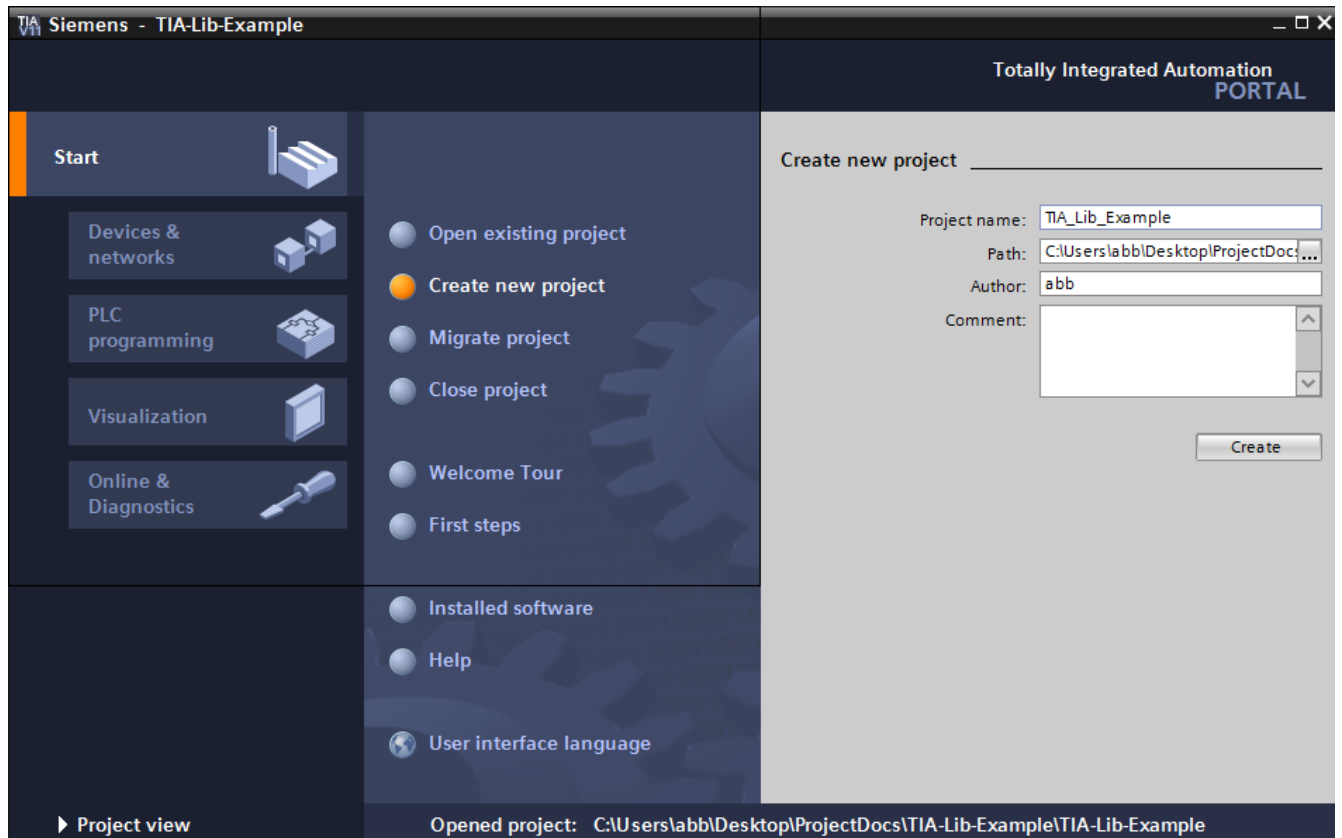
Figure 44 New project

3. In the Start options, click **Configure a device**.

First steps

Project: "TIA_Lib_Example" was opened successfully. Please select the next step:

Start

Devices & networks — Configure a device

PLC programming — Write PLC program

Visualization — Configure an HMI screen

▶ Project view — Open the project view
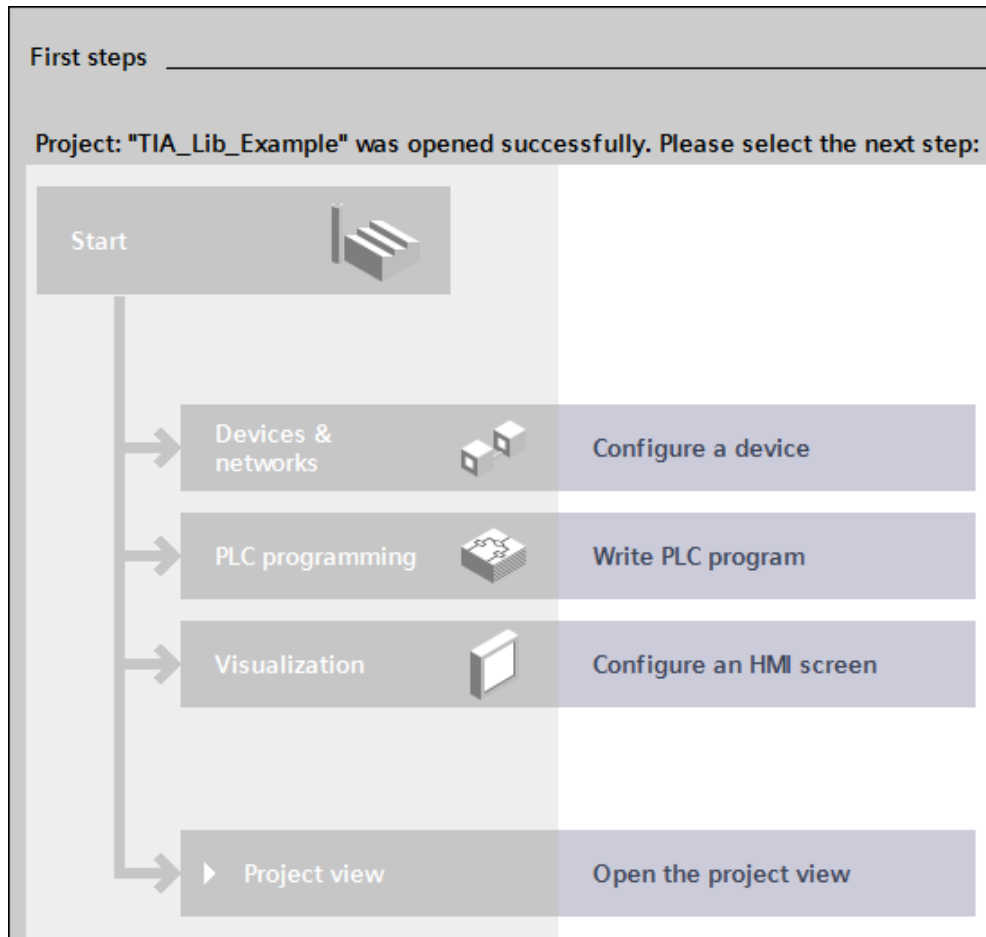
Figure 45 Configure device

4. Select **Add new device** in the left pane and select the required PLC and then click **Add**.
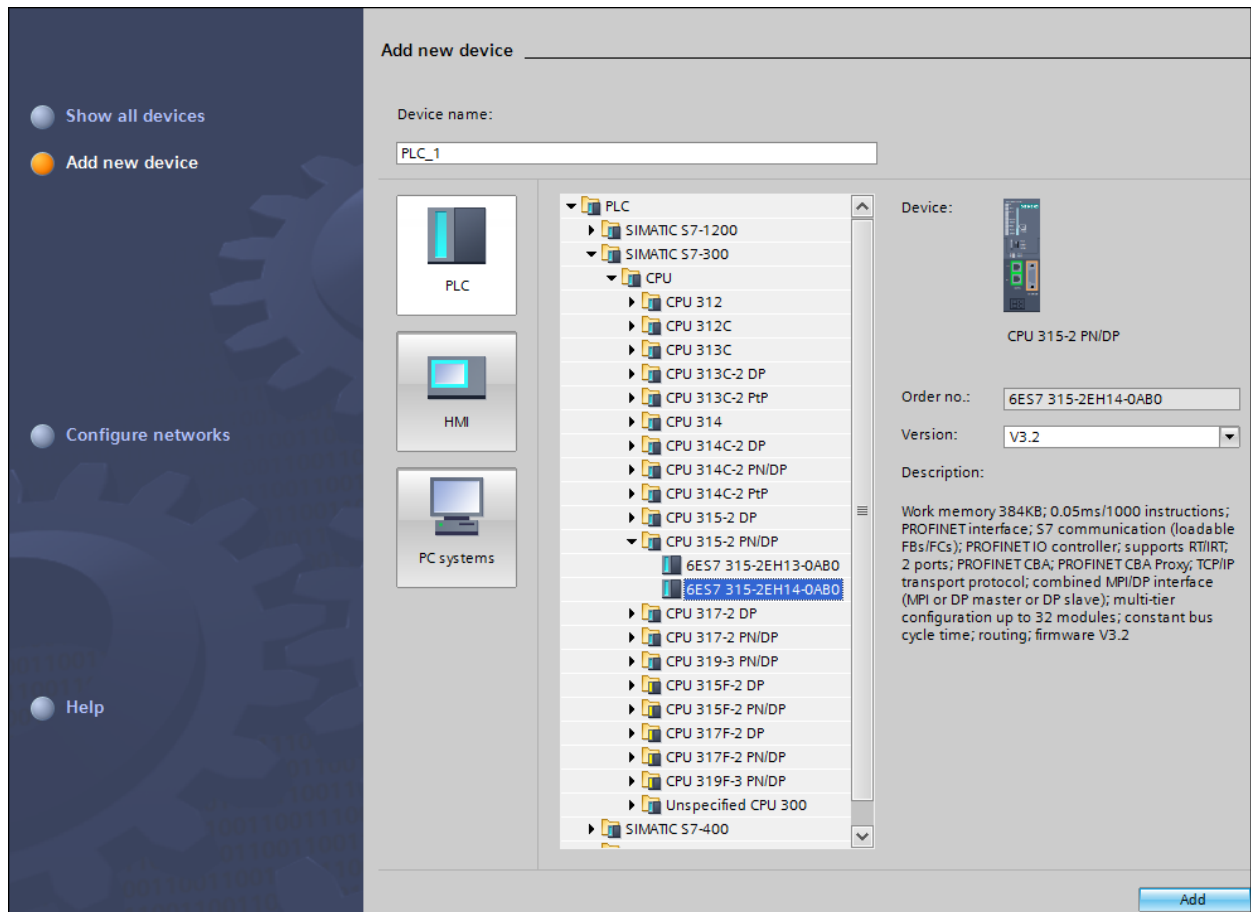
Figure 46 Adding new device

New PLC device is added to hardware configuration.
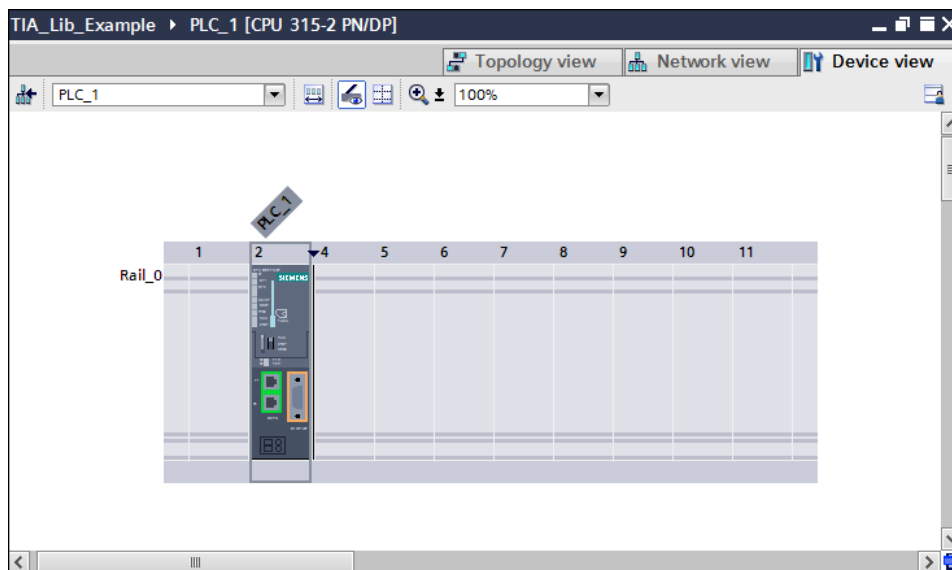
Figure 47 PLC device

5. In the Device view, select **PLC_1** (PROFINET interface) and in the General tab, select **Ethernet addresses** and click **Add new subnet** to add the subnet and then set IP Address in the IP protocol.
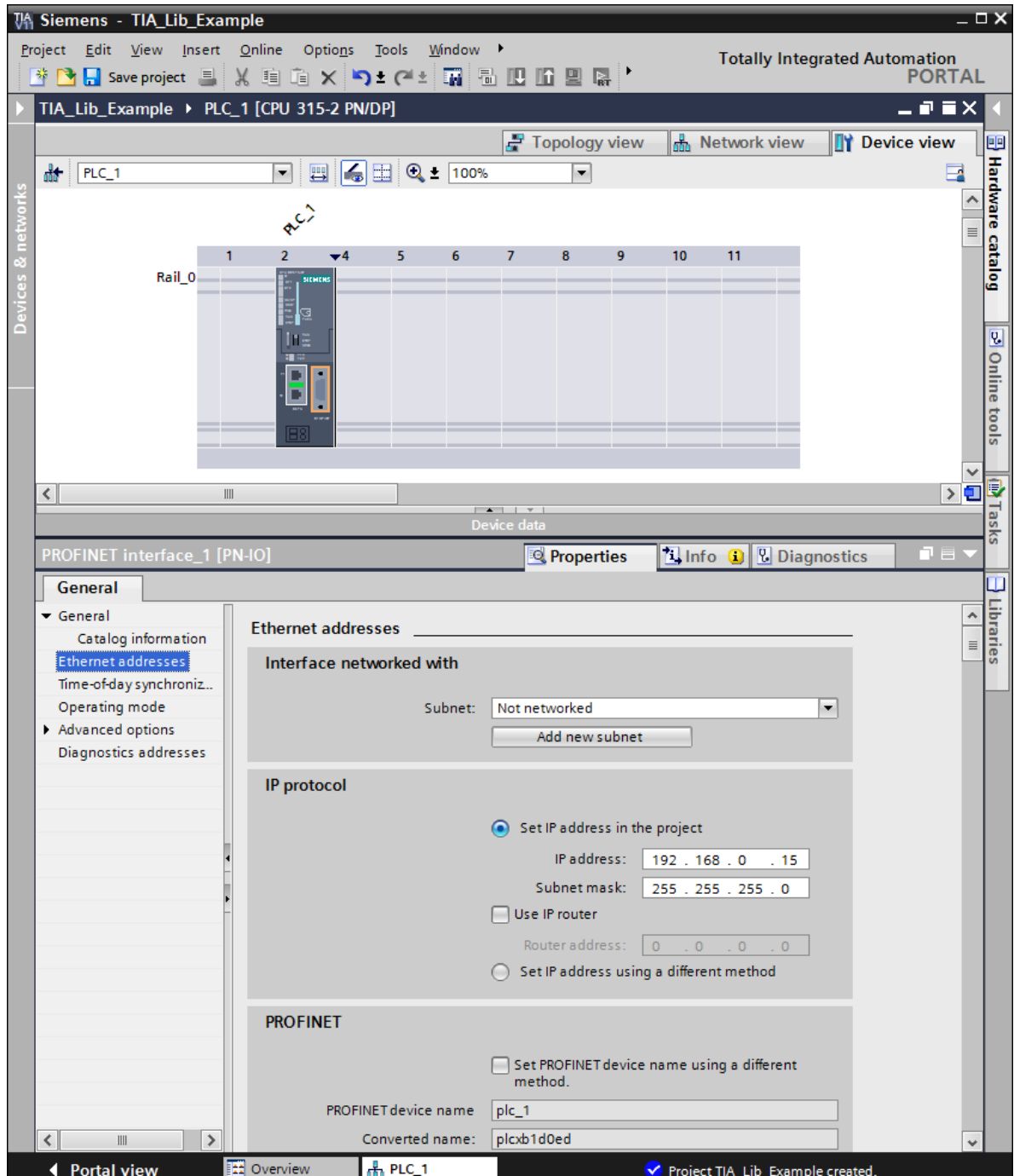
Figure 48 Assign IP

A new subnet and IP protocol is added.
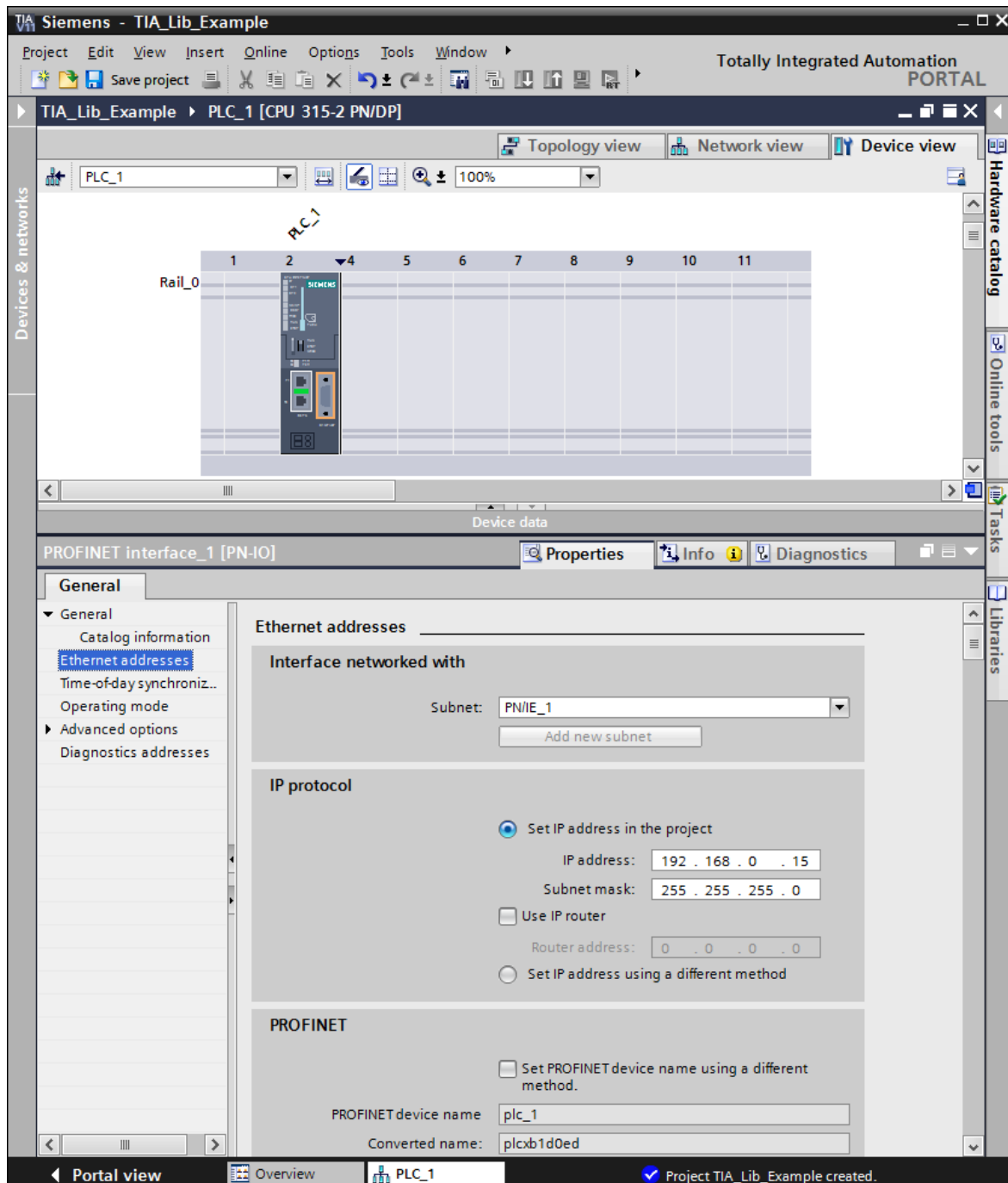
Figure 49 IP added

6. In the Device view, select **PLC_1 (MPI/DP interface)** which is highlighted.
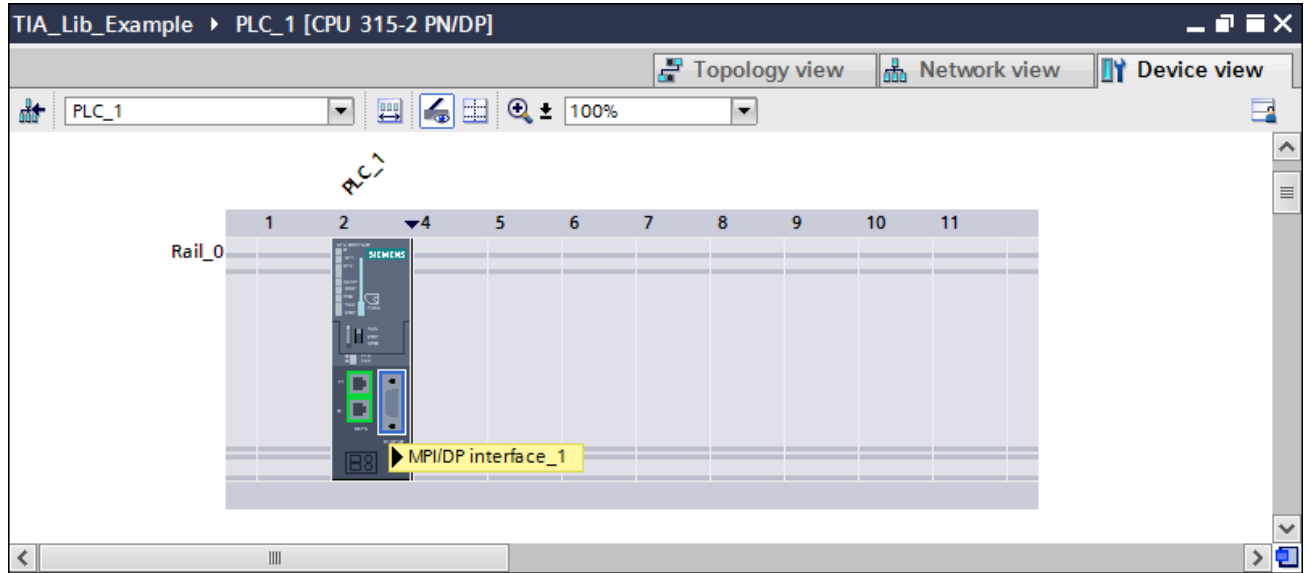
Figure 50 Selecting MPI DP

7. In the General tab, select **MPI address** and in the Parameters Interface type drop-down list select **PROFIBUS** as interface to assign PROFIBUS address.
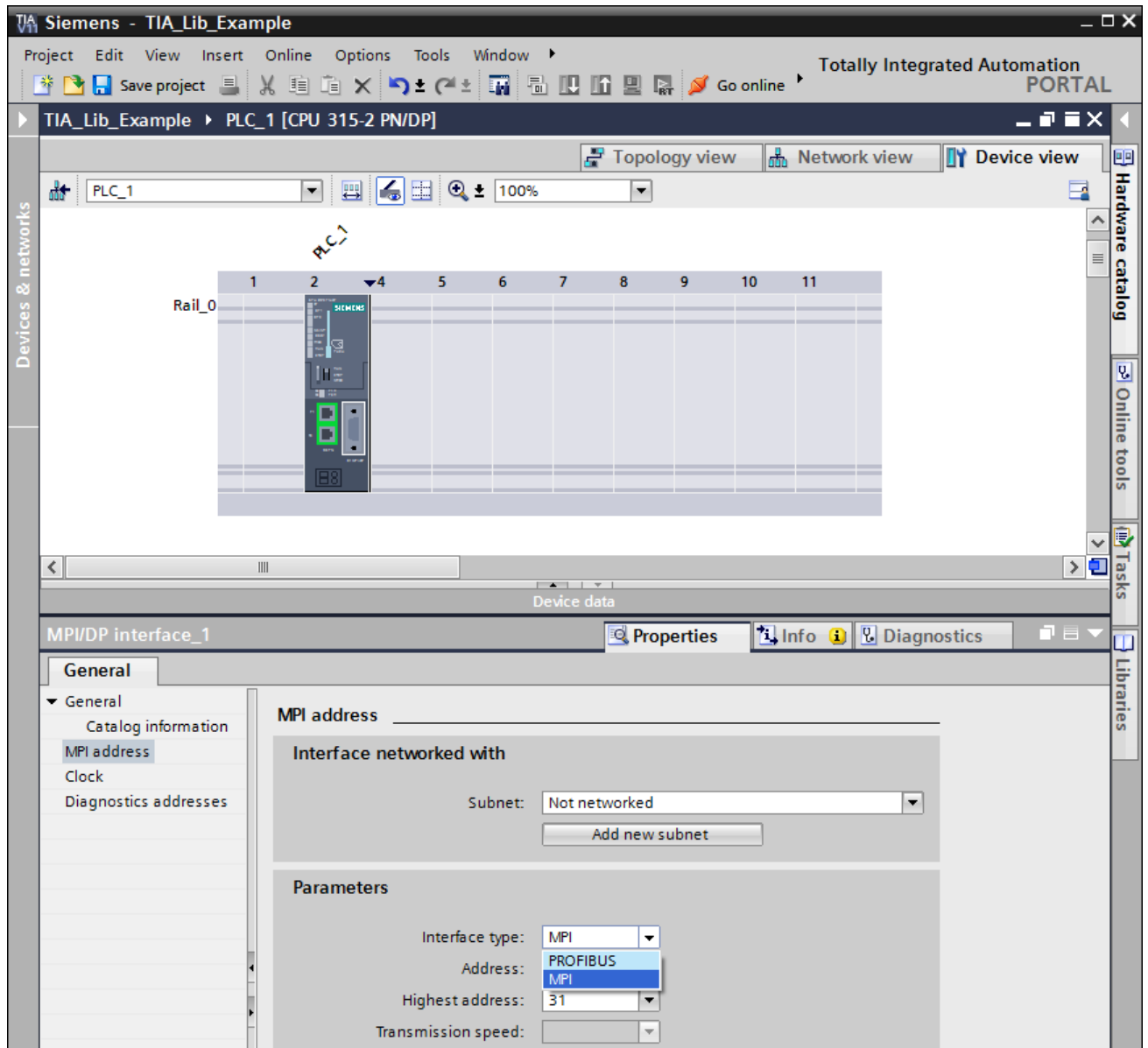
Figure 51 MPI PROFIBUS address change

8. In the PROFIBUS address option, click **Add new subnet** to add the interface network.
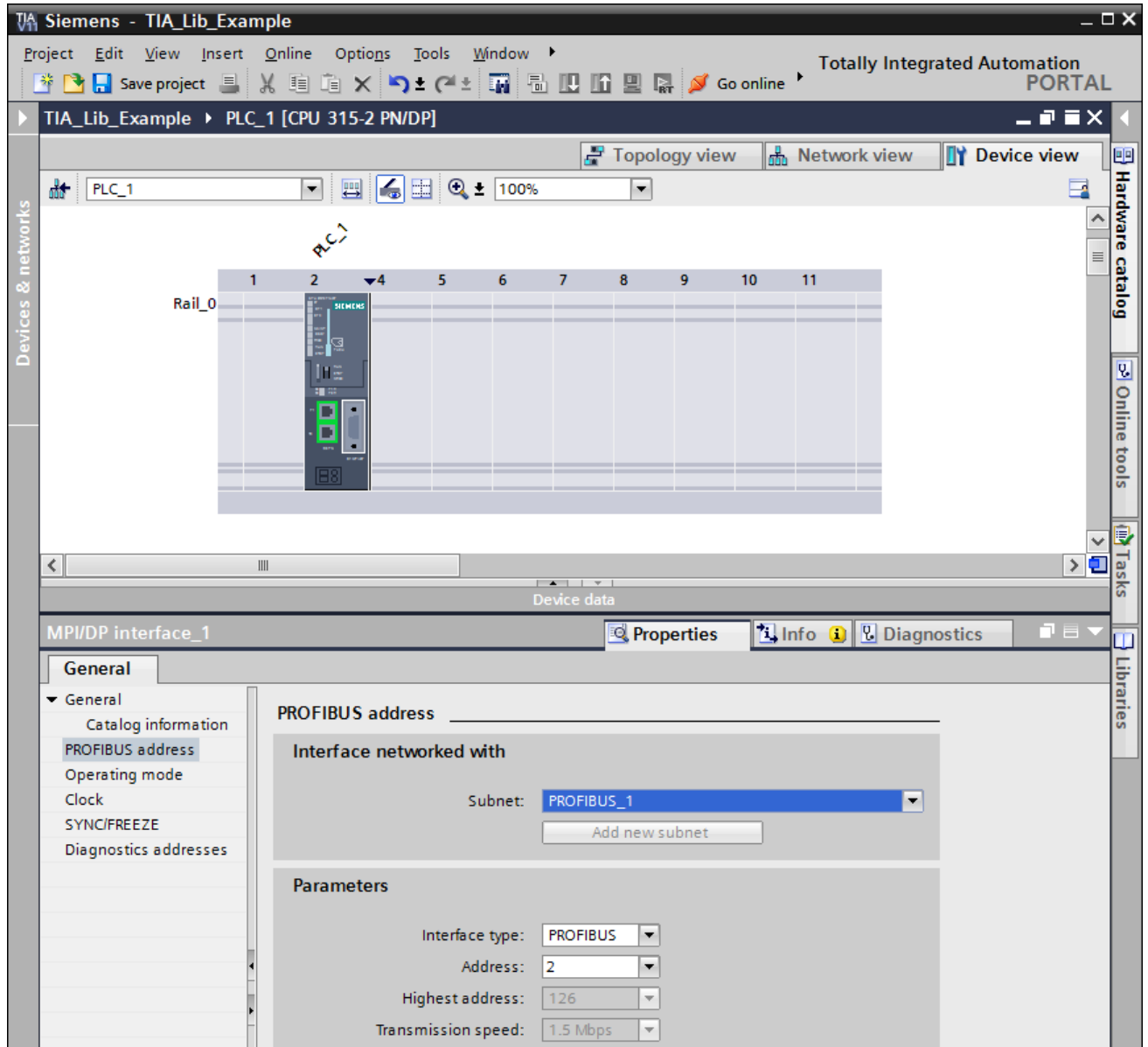
Figure 52 Adding new PROFIBUS subnet

9. After **configuring** PROFINET and PROFIBUS interfaces in TIA portal, click **Network view** to view the PLC connections.
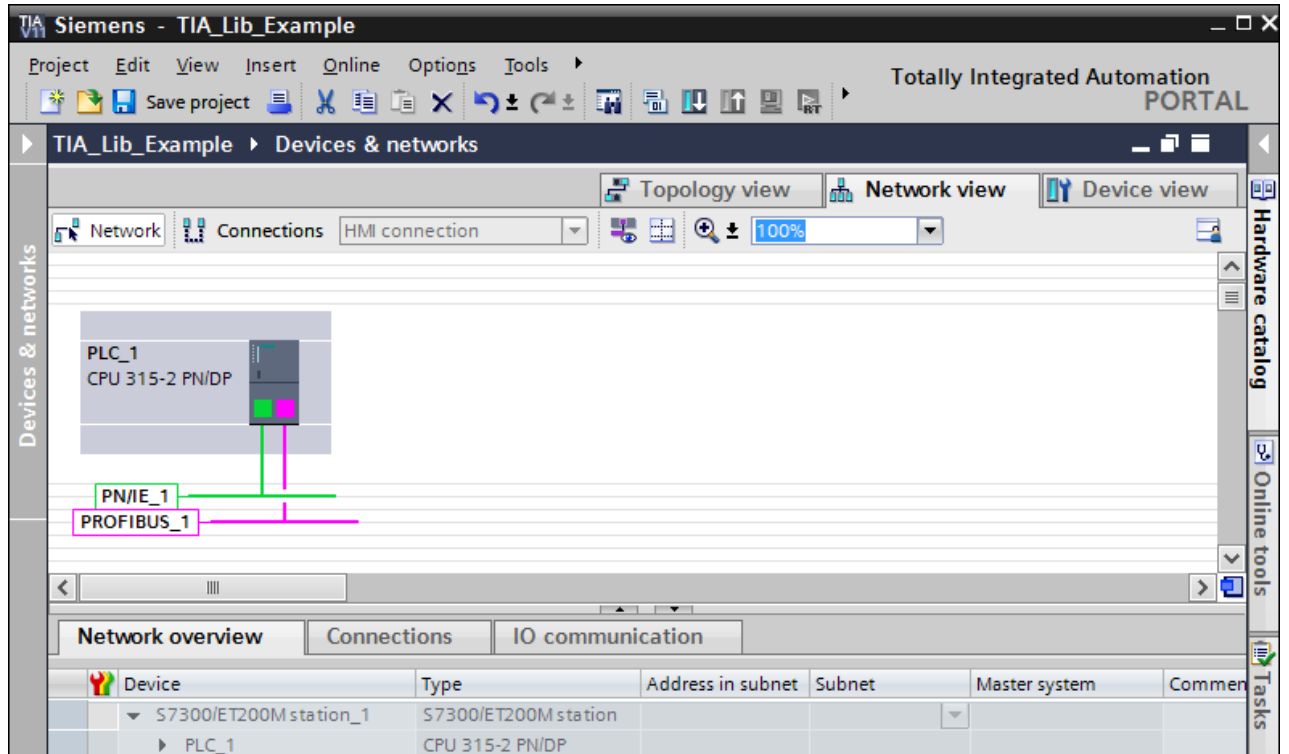
Figure 53 PLC connections

Installing GSD and GSDML file for S7-300 PLC

1. In the TIA portal main menu, click **Options** and then click **Install general station description files (GSD)**.
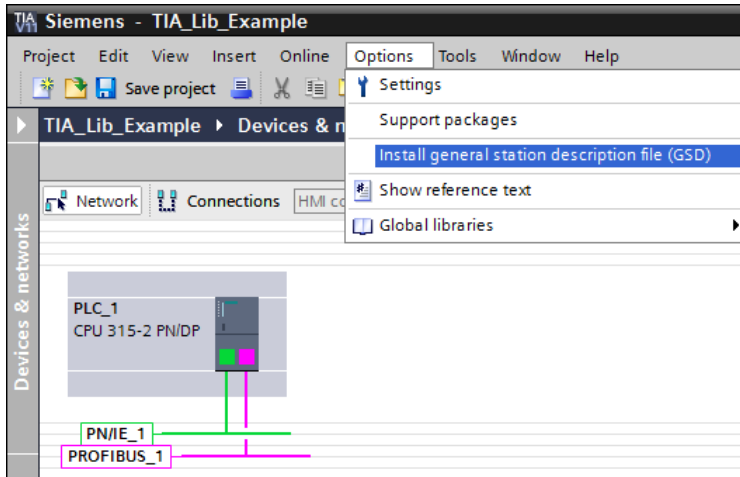
Figure 54 Installing general station description file

2. Browse and select the required GSD files for PROFIBUS or GSDML files for **PROFINET** and click **Install**.
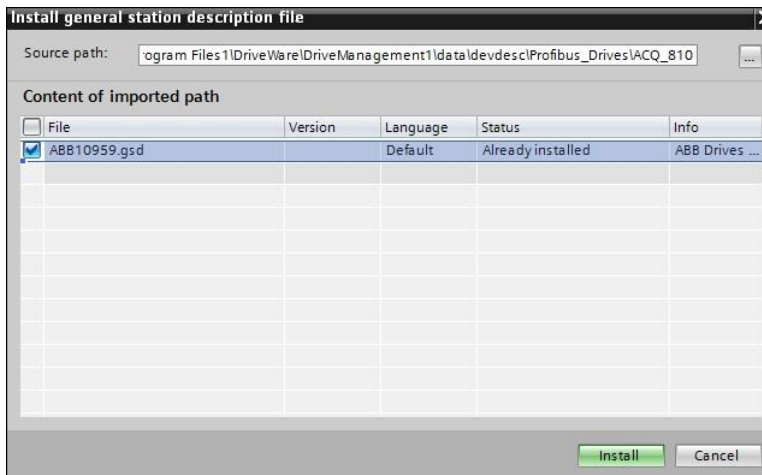
Figure 55 Installing GSD file

3. A popup window is displayed to confirm installing the GSD file. Click **Yes**.
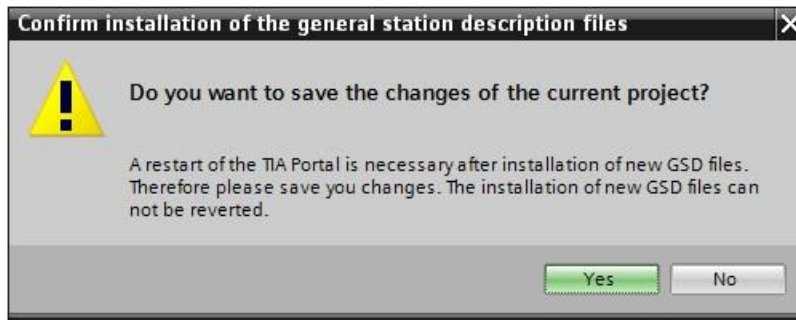
Figure 56 Popup window

After GSD file is installed, system prompts to close TIA portal, click **Close TIA Portal**.
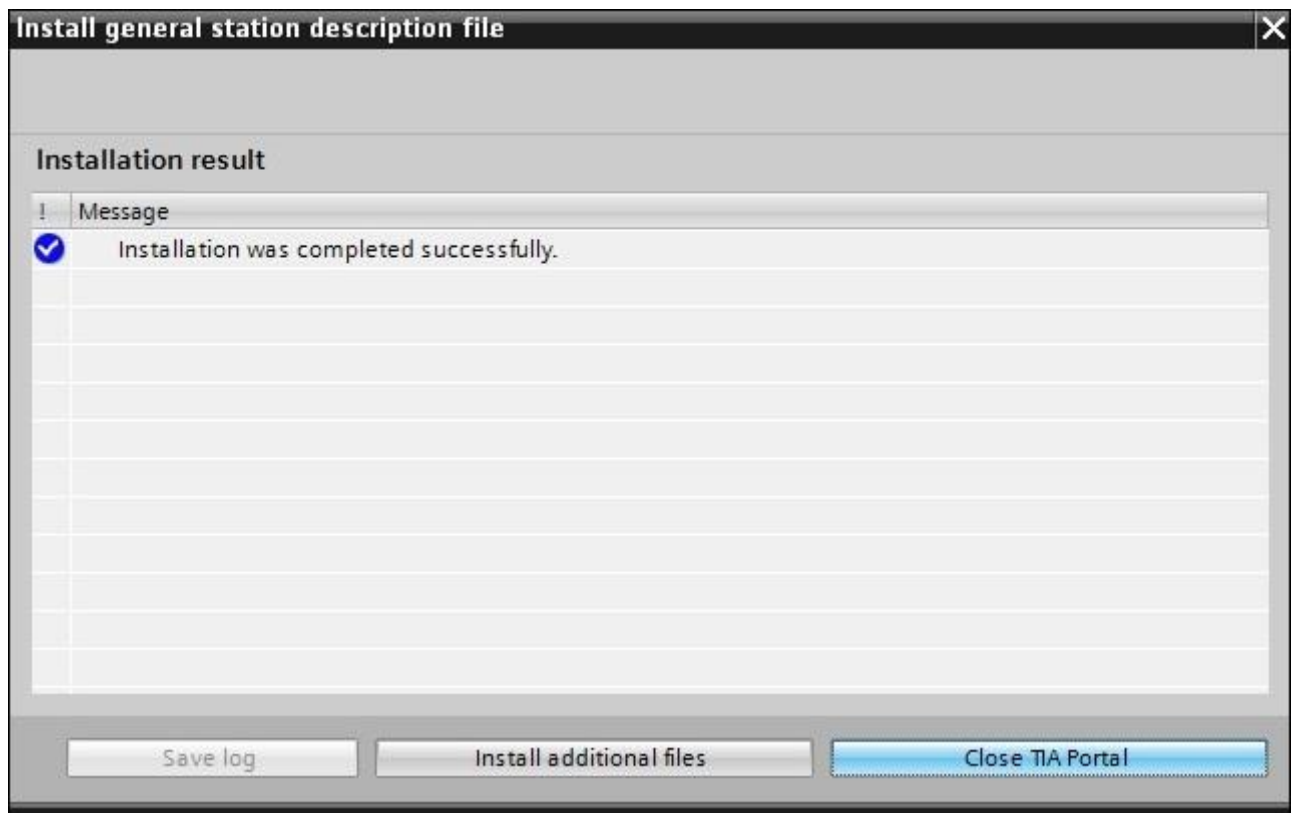
Figure 57 Restart TIA portal

PLC libraries for S7-300

Adding ABB drives to PROFIBUS DP line

1. Launch TIA portal and click **Open the project view** in the **Start** options.

   Note: After installing GSD files, ABB drives are added to the hardware catalog of TIA portal.

2. In the Project tree pane, double-click **Device & networks**.
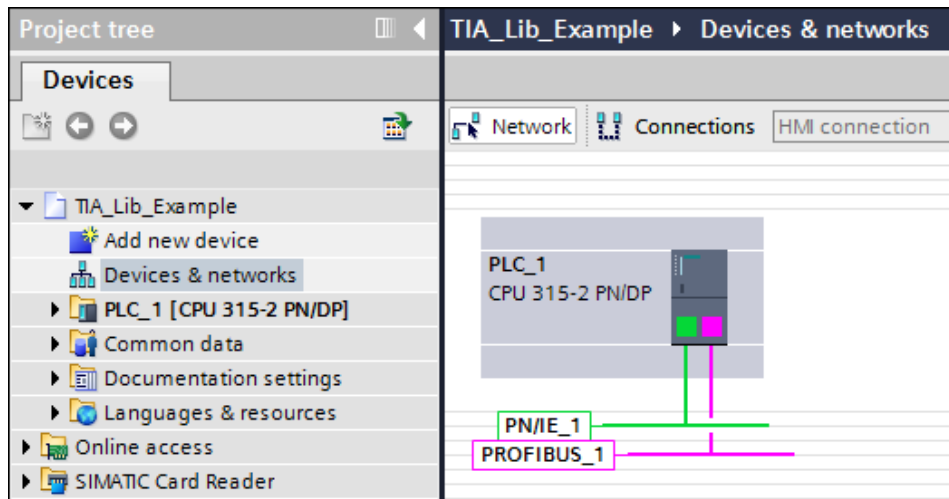
Figure 58 Devices and network

3. From the hardware catalog pane, drag and drop the required ABB Drive to the PLC.
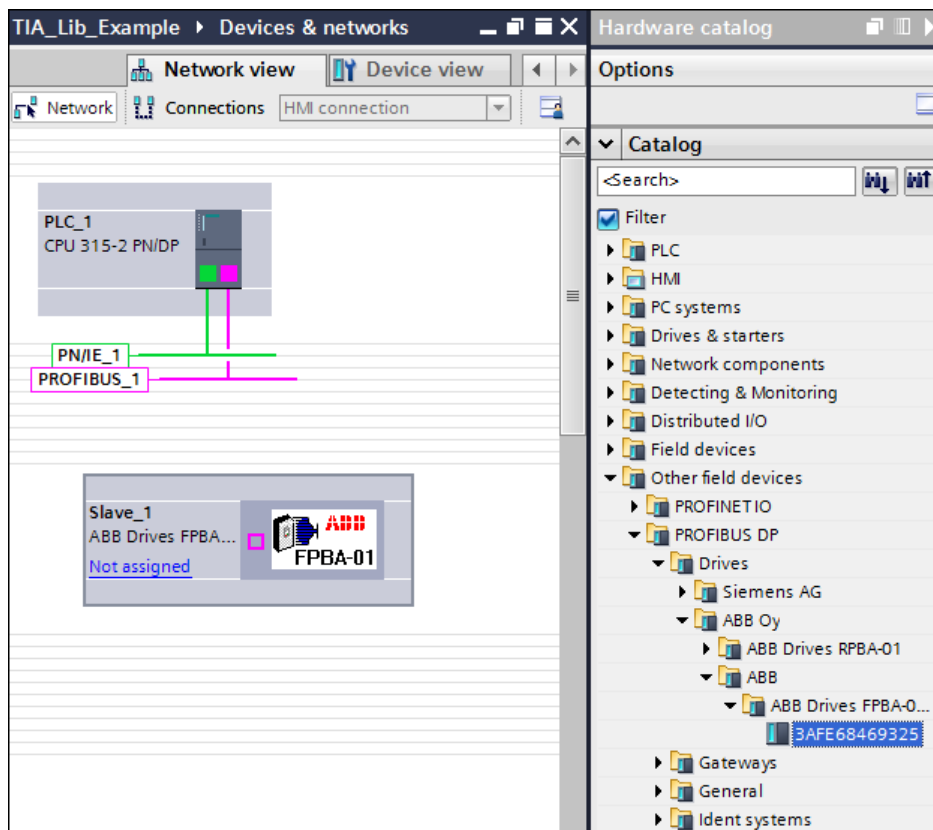
4. In the ABB drive, connect PLC to the PROFIBUS interface.
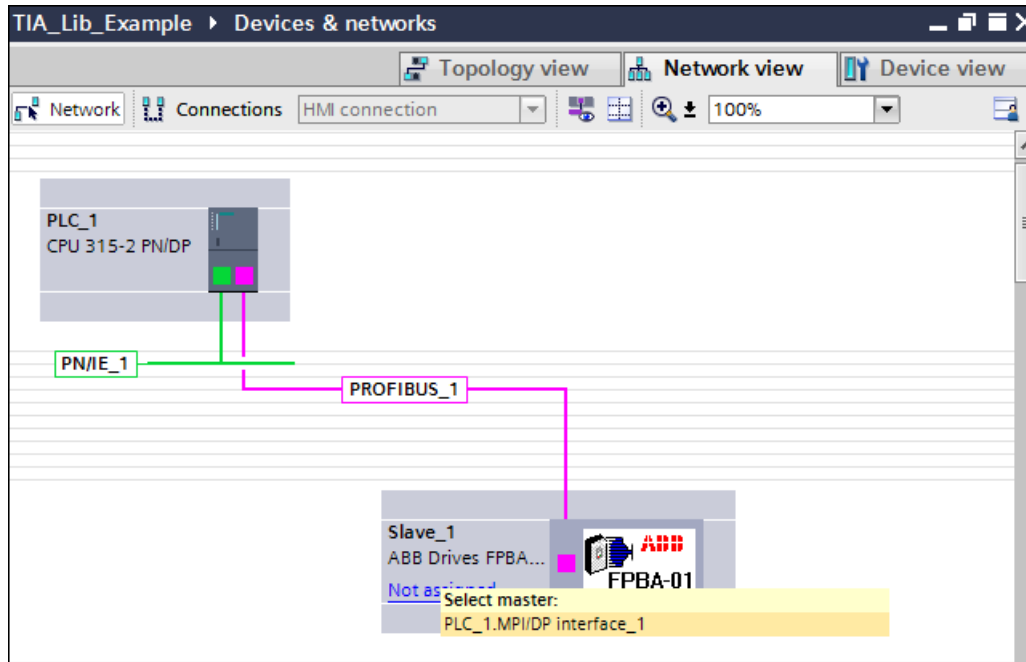
Figure 60 Assign to PLC

5. Select the ABB drive (Slave_1) and click Device view to configure PROFIBUS address.
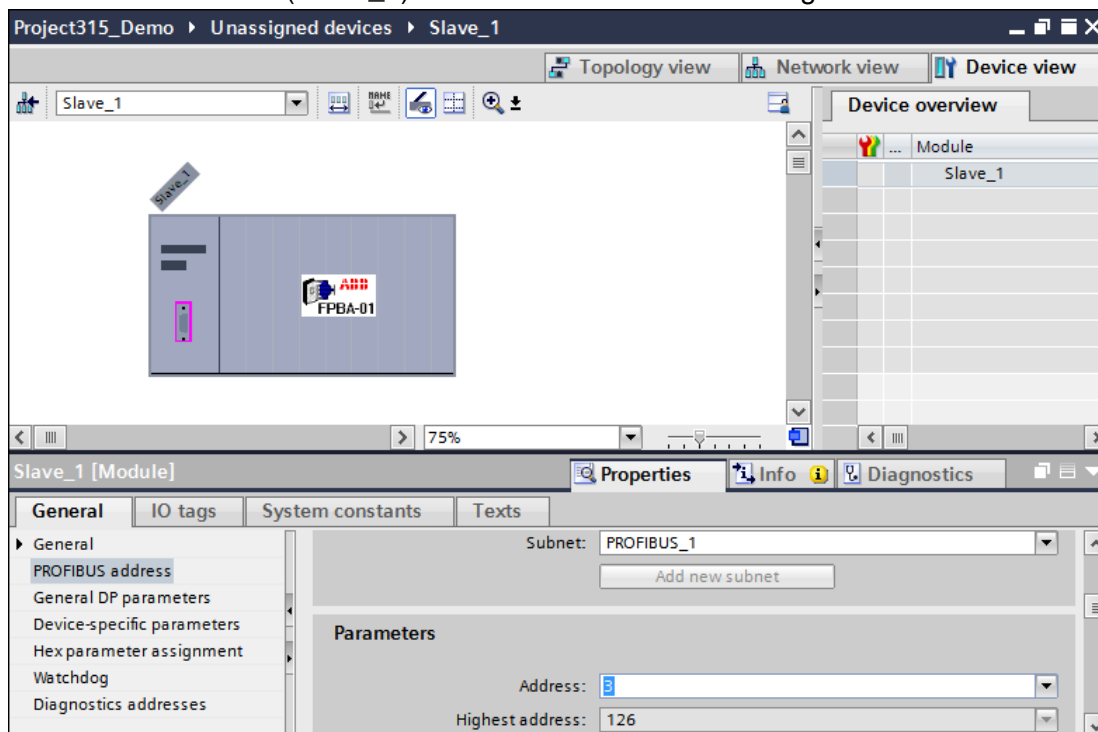
Figure 61 Configure address

6. In the Hardware catalog pane, double-click the required **PPO type**. The selected PPO type is displayed in the Device overview.
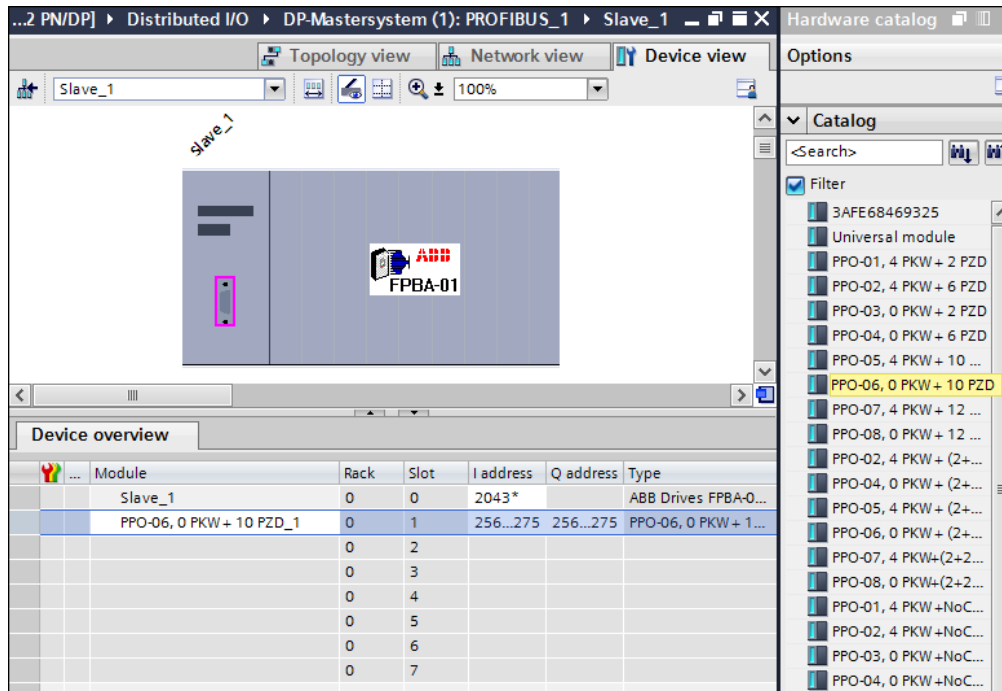
Figure 62 PPO type added

7. In the Network view, click **Not assigned** and select **PLC_1.MP/DP interface_1** to assign PLC to the ABB drive.
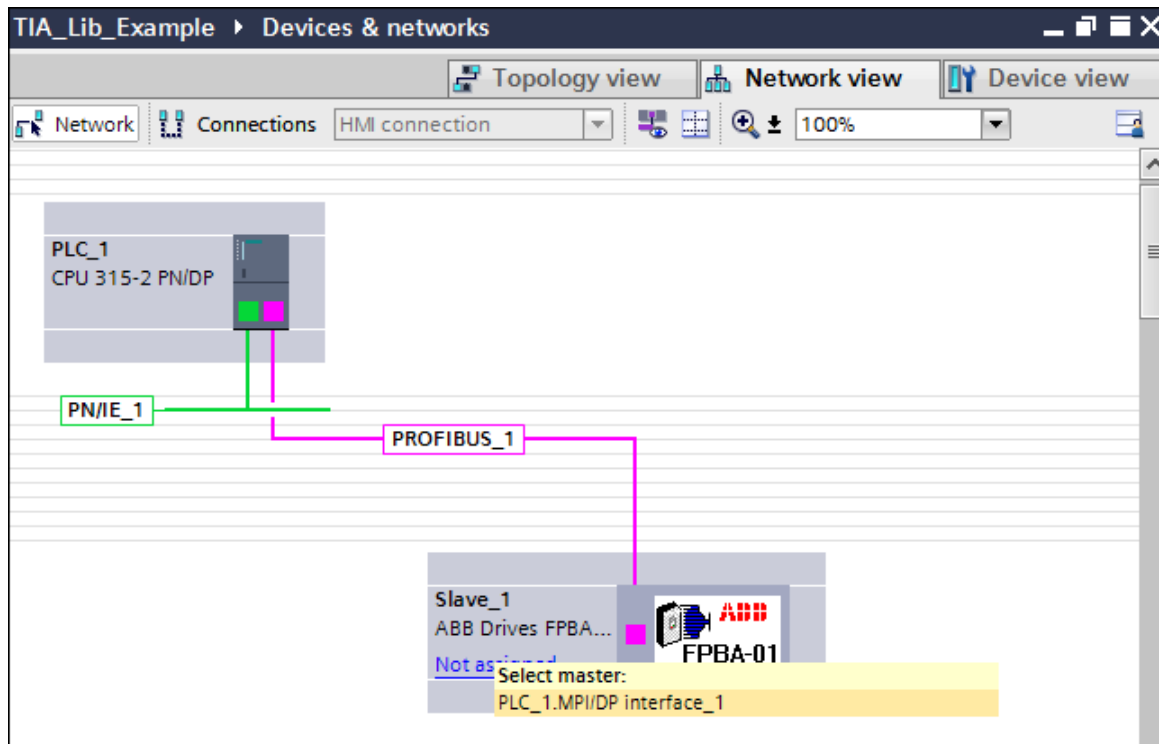
Figure 63 Assign to PLC

The selected PLC is assigned to the ABB drive.
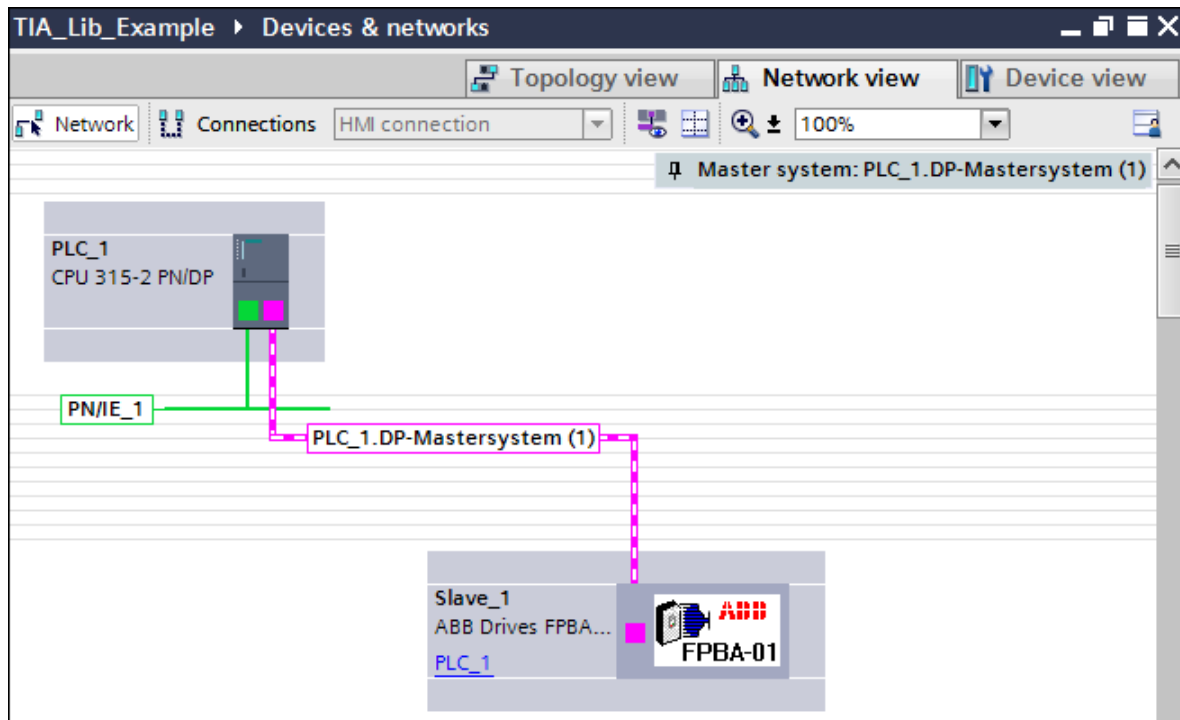
Figure 64 PLC assigned

8. In the Project tree pane, select **PLC_1** and then click ⊞ to compile.
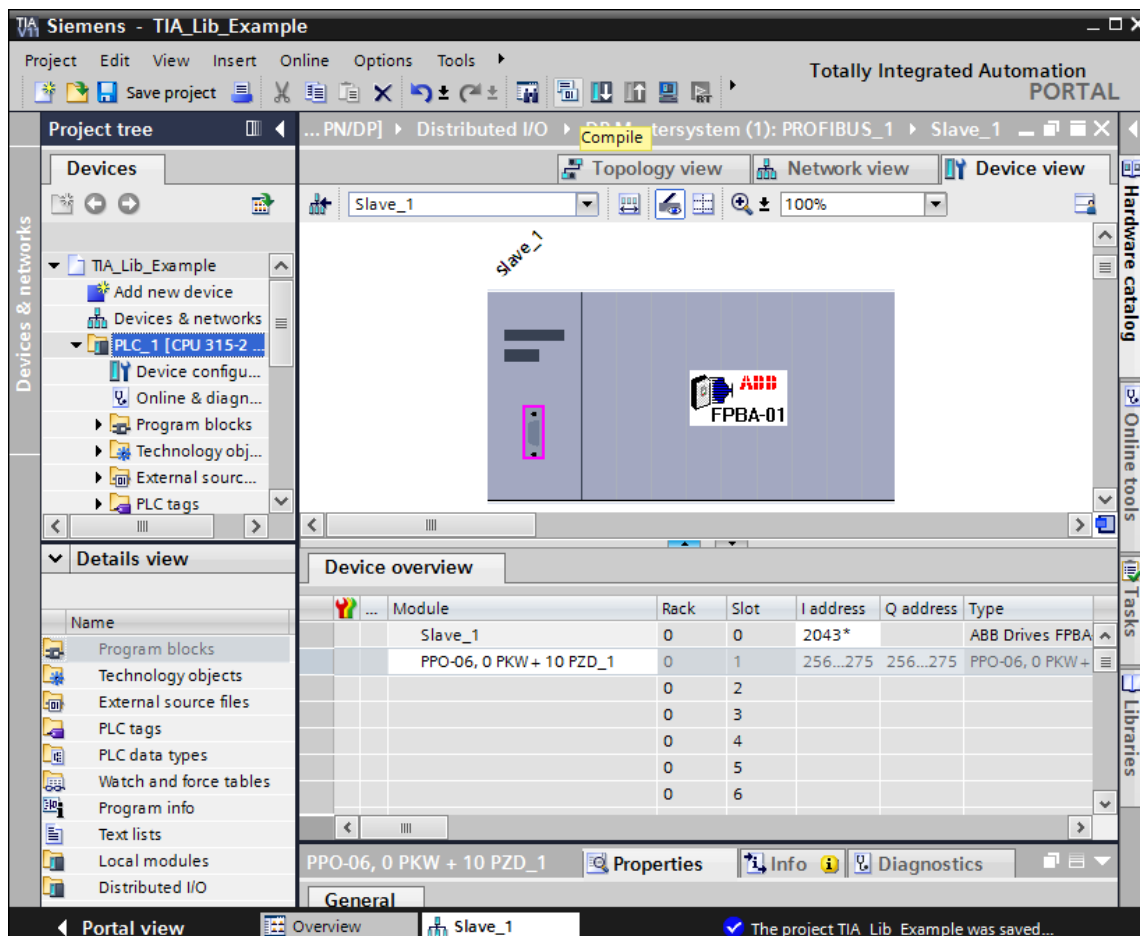
Figure 65 Compile

Note: Before downloading the configuration to PLC, check the PC IP address. For more information, see section *Configuring PC IP address* (page *64*).

9. Click ⬇ to download the configuration to PLC.
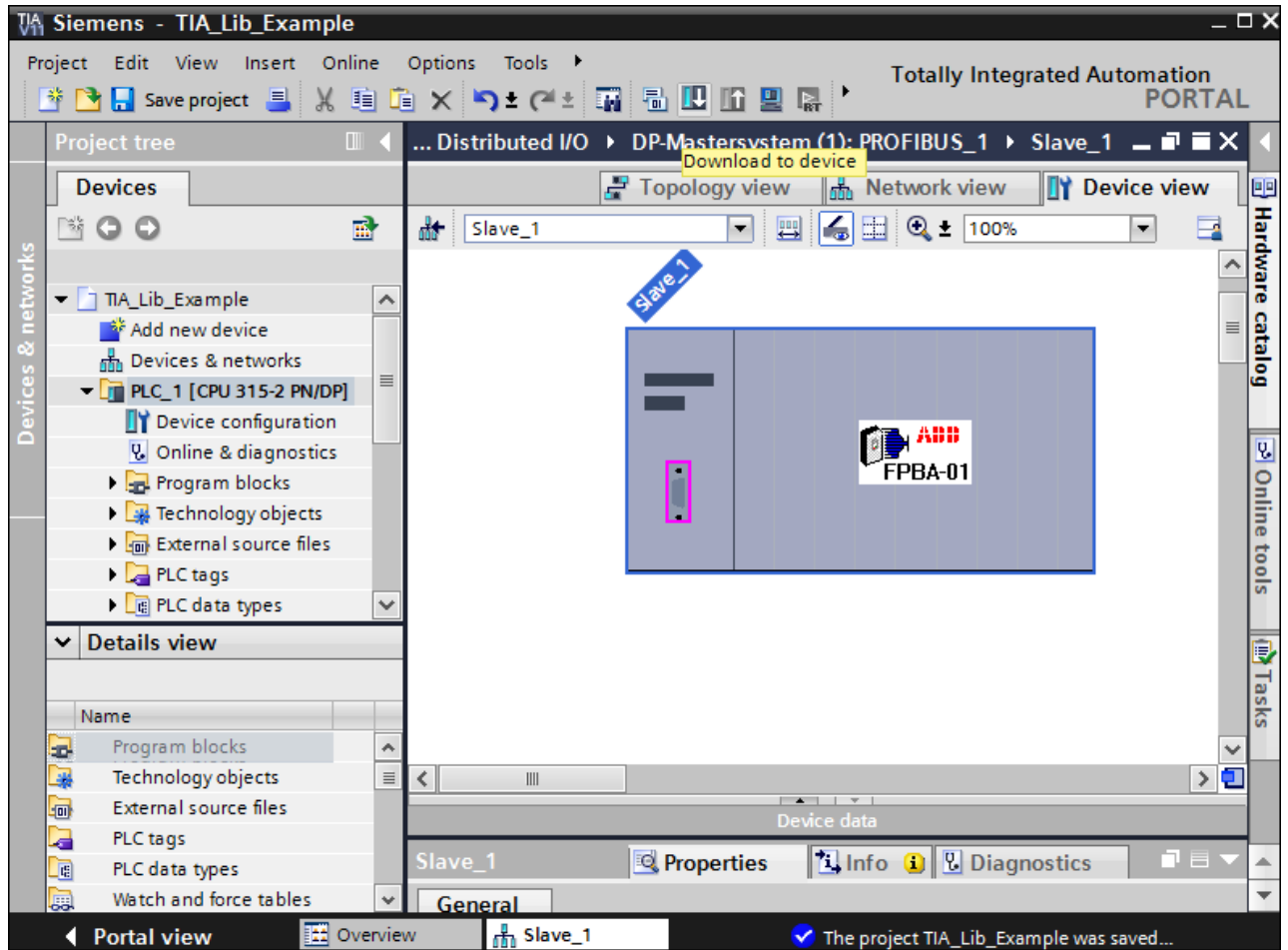
Figure 66 Download to device

Adding ABB drives to PROFINET

Note: After installing GSDML files, ABB drives are added to the hardware catalog of TIA portal.

1. Select Network view, and then from the hardware catalog pane, drag and drop the required ABB Drive to the PLC PROFINET network.
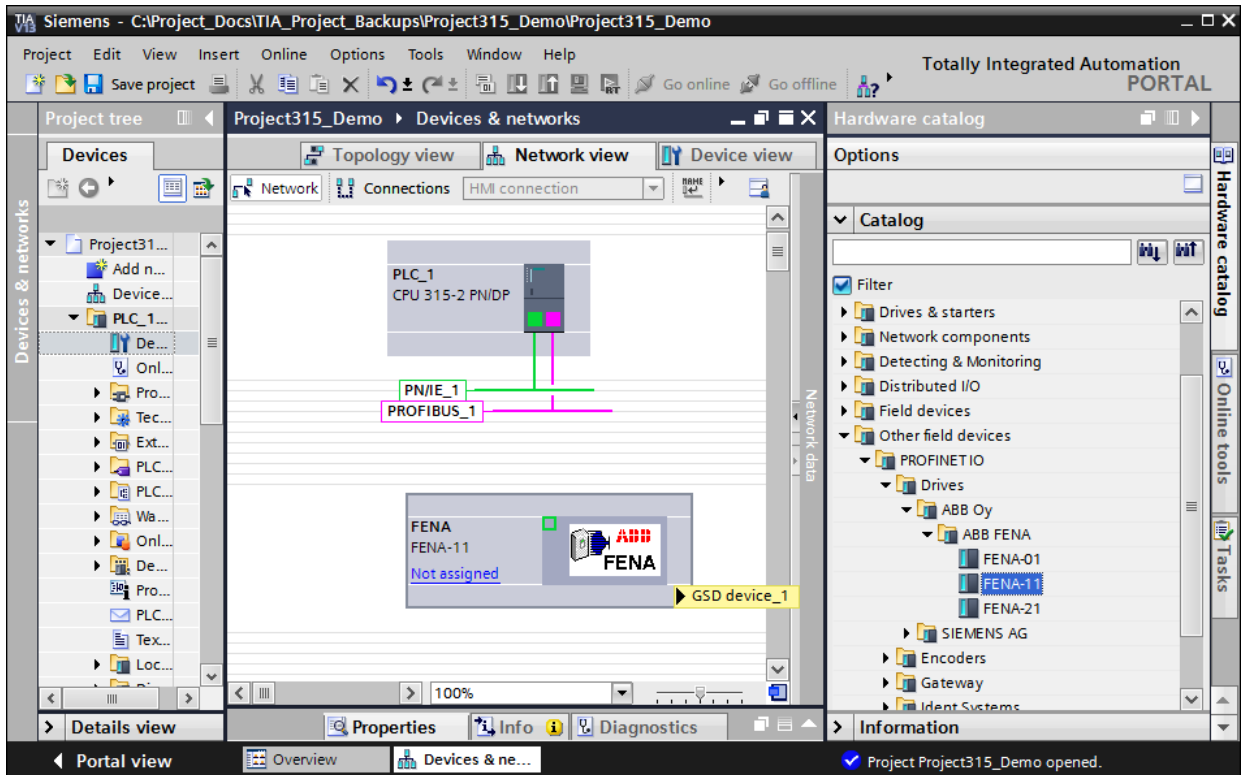
Figure 67 PLC PROFINET

2. In the ABB drive, connect PLC to the PROFINET interface.
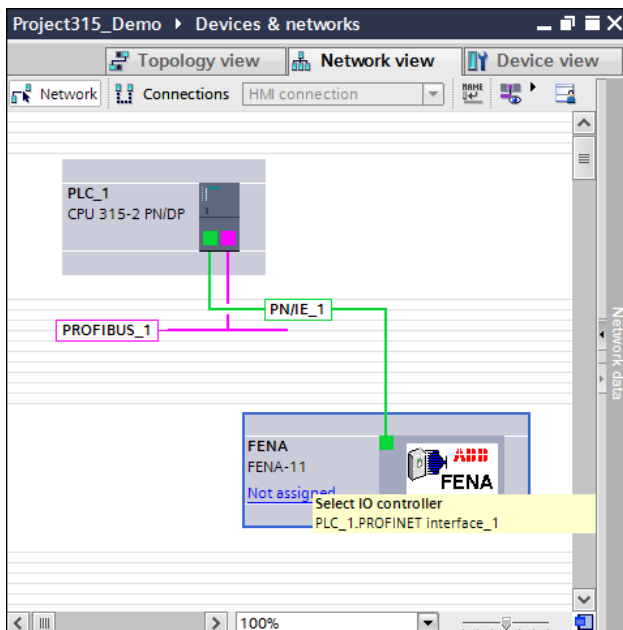
Figure 68 PROFINET interface

3. Select the ABB drive (FENA) and click Device view to configure PROFNET IP address and device name.

- In the General tab, click **General** to add the device name.
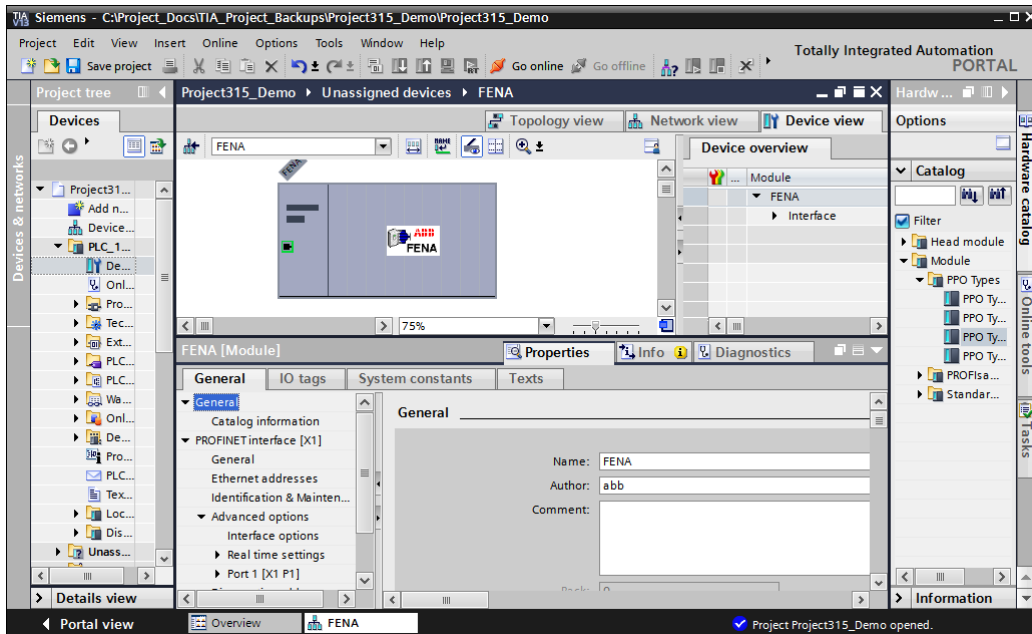
Figure 69 Configuring device name

- In the PROFINET interface [X1], click **Ethernet address** to add **Subnet** and **IP address**.

Figure 70 Ethernet address

4. In the Hardware catalog pane, double-click the required PPO type. The selected PPO type is displayed in the Device overview.
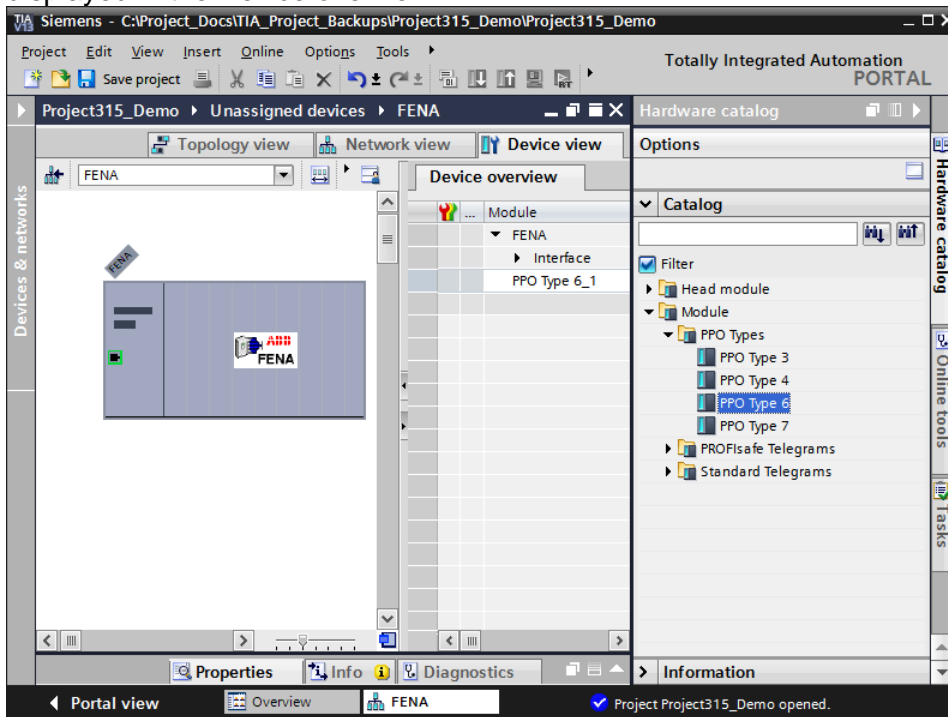
Figure 71 PPO type

5. In the Network view, click **Not assigned** and select **PLC_1.PROFINET interface_1** to assign PLC to the ABB drive.
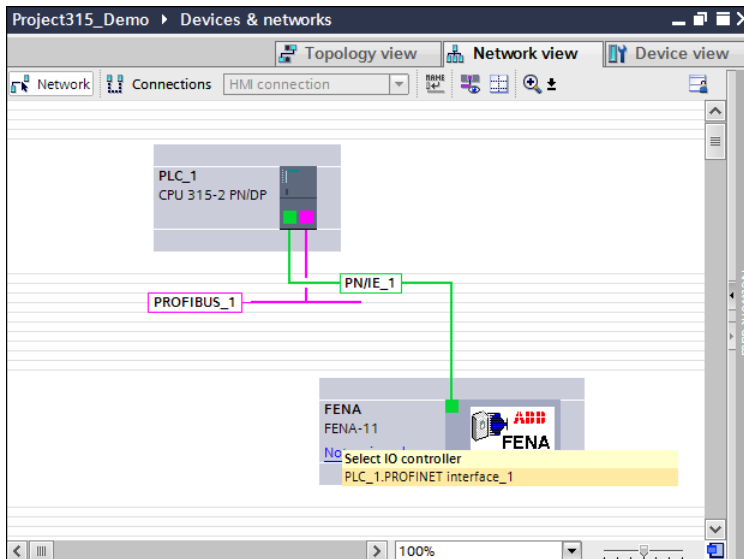
Figure 72 Assigning PLC PROFINET

The selected PLC is assigned to the ABB drive.
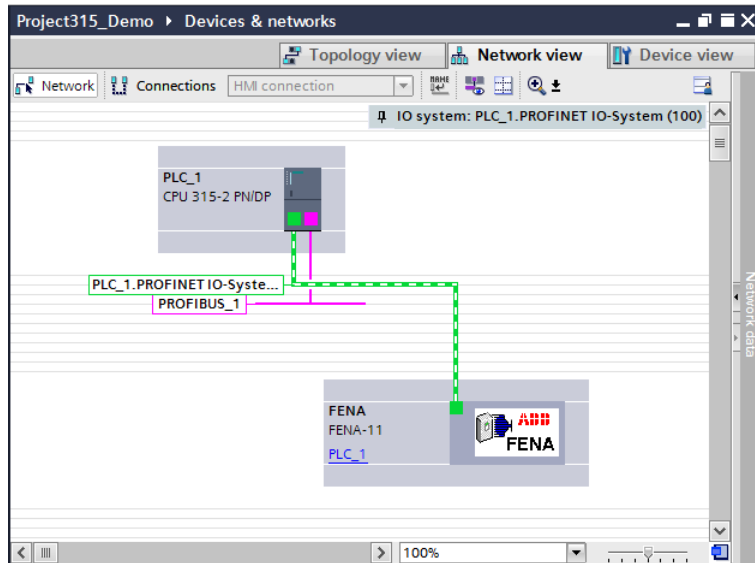
Figure 73 PLC

6. In the Project tree pane, select **PLC_1** and then click **Save** project and click 🔲 to compile.

Downloading configuration to PLC

1. Click ⏬ to download the configuration to PLC.
2. In the Extended download to device window, select **PG/PC interface** and **Connection to subnet** from the drop-down list and click **Load**.
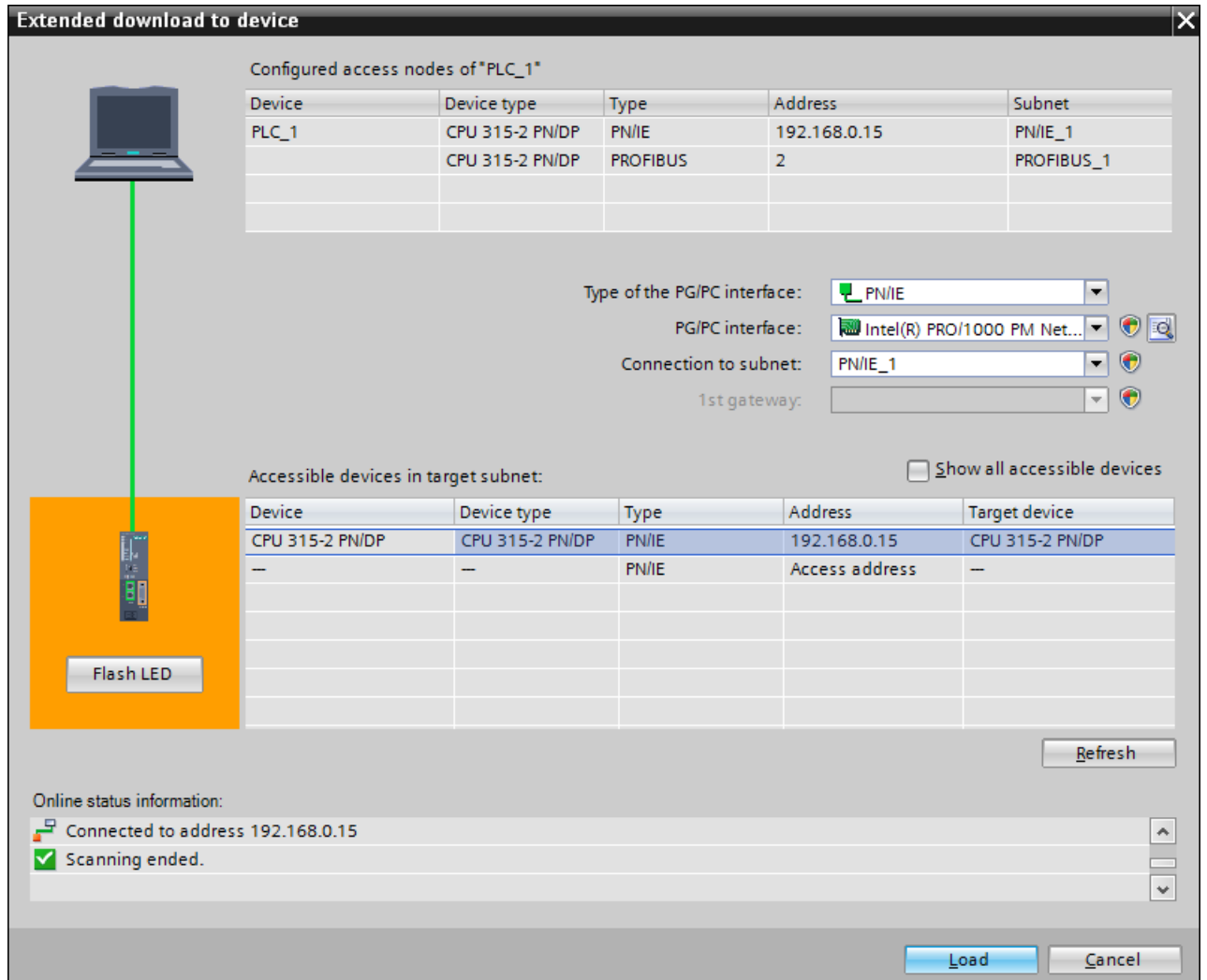
Figure 74 Extended download to device

The **Load preview** window is displayed with the PLC ready for loading.

3. Select the required PLC device and click **Load**.
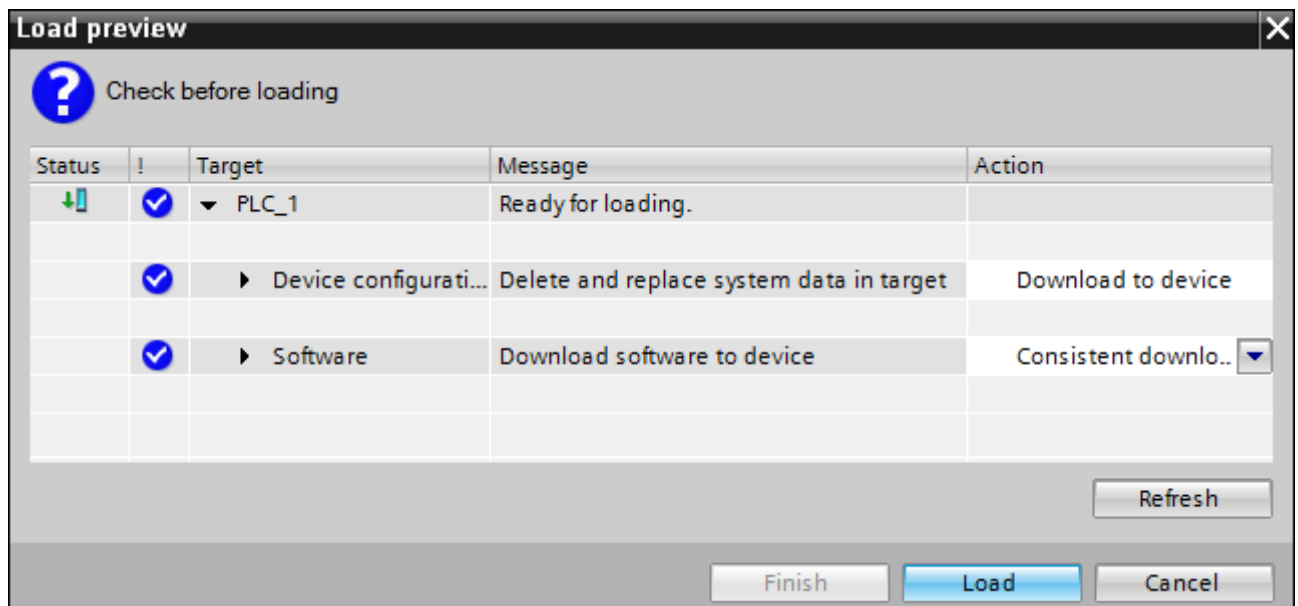
Figure 75 Load preview

4. In the Load results window, select **Start all** and click **Finish**.
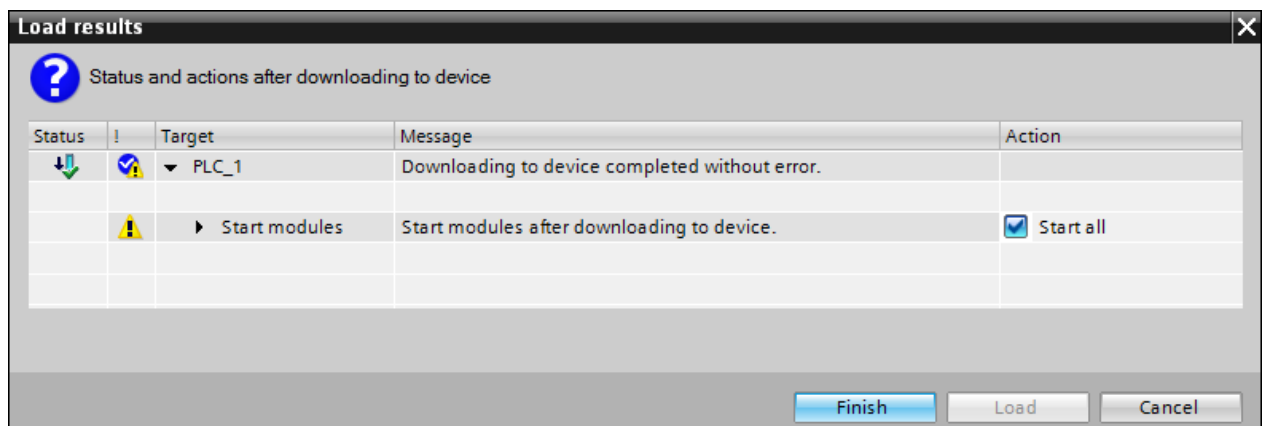
Figure 76 Load results

Copying ABB drive libraries to S7-300 project

Prerequisites: Check for ABB Drive libraries available in the user PC.

1. In the main menu of TIA portal, click **Options** and click **Global libraries** and then click **Open library**.
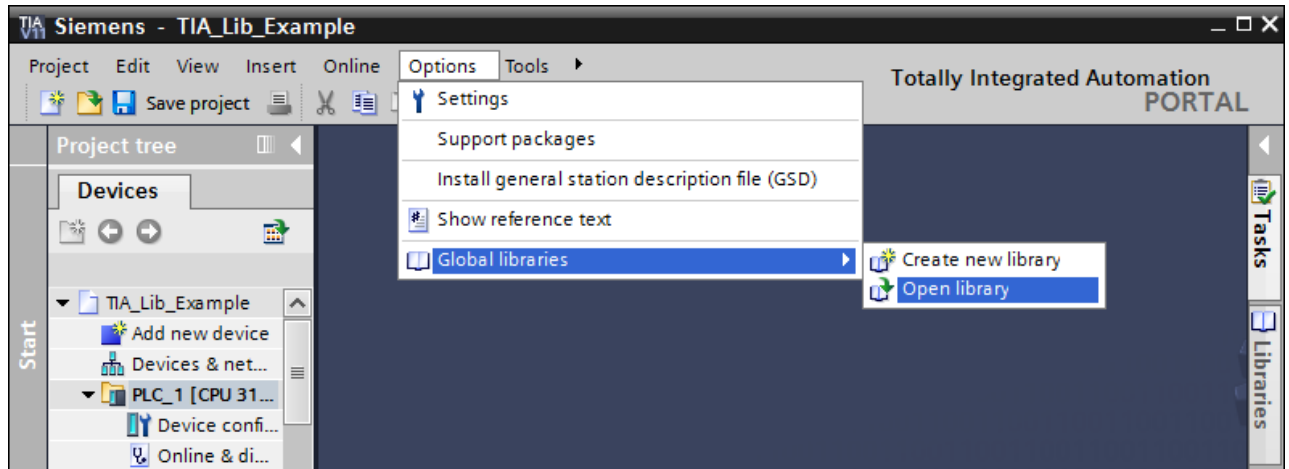
Figure 77 Global library

2. Select the required ABB drive library in the file system and click **OK**.
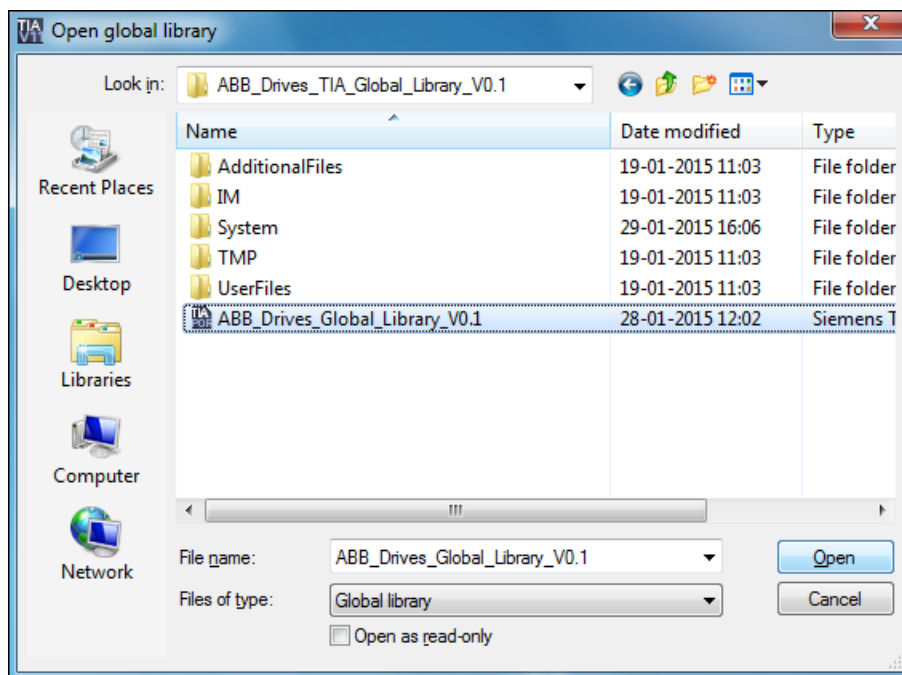
Figure 78 File system

After selecting ABB drive library, a Libraries pane is displayed with the selected library at the right side of the TIA portal window.

3. Expand **ABB_Drives_Global_Library** -> **Master copies** and right-click
    ABB_DRIVE_CONTROL_FB and then click **Copy**.
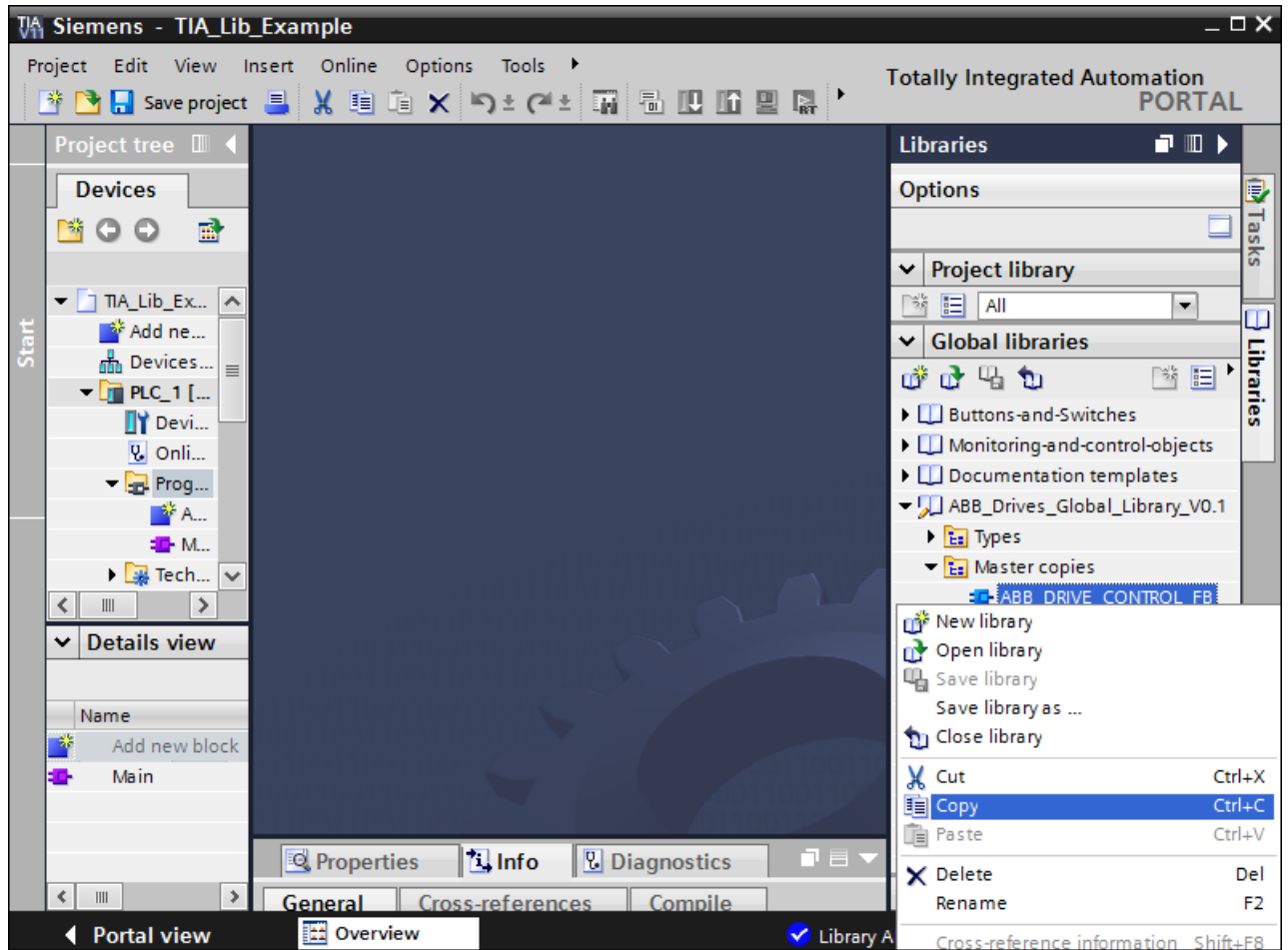
4. In the Project tree pane, right-click **Program blocks** and click **Paste**.
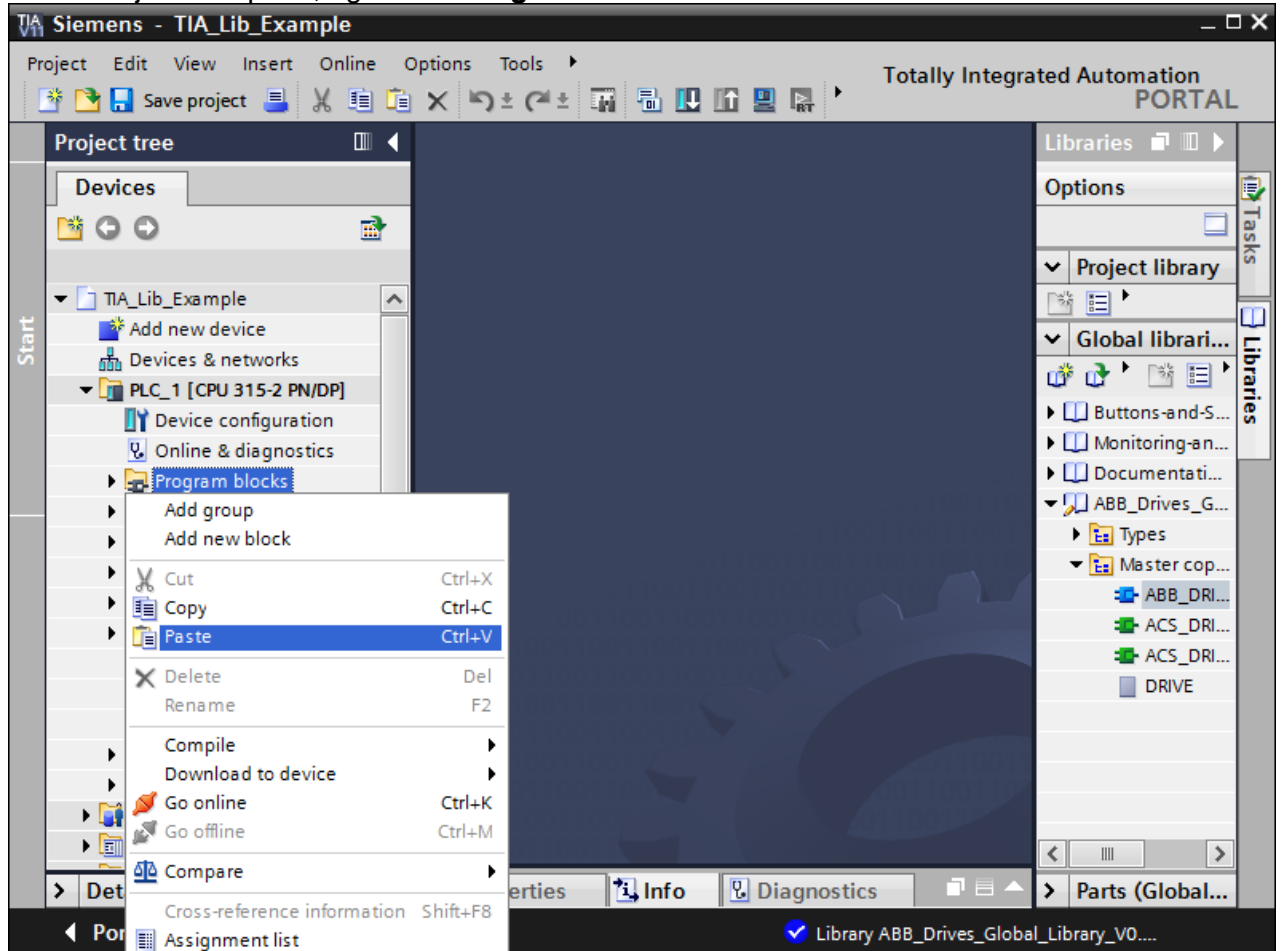
Figure 80 Program blocks

ABB drive libraries are copied to PLC.
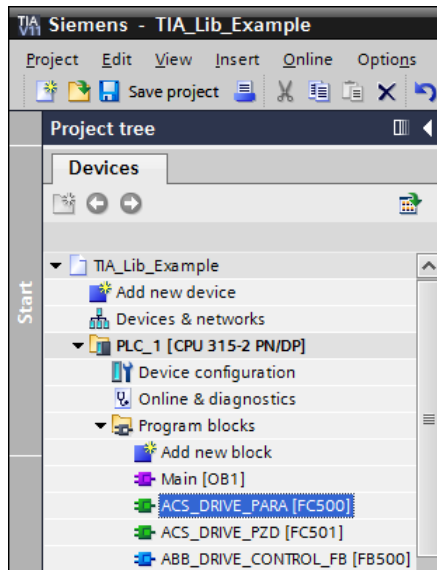
Figure 81 Copied libraries to PLC

Note: Similarly, copy other ABB drive libraries and paste it in the **Program blocks**.

5. In the Libraries pane, expand **ABB_Drives_Global_Library** -> **Master copies** and right-click **DRIVE** (data types) and click **Copy**.
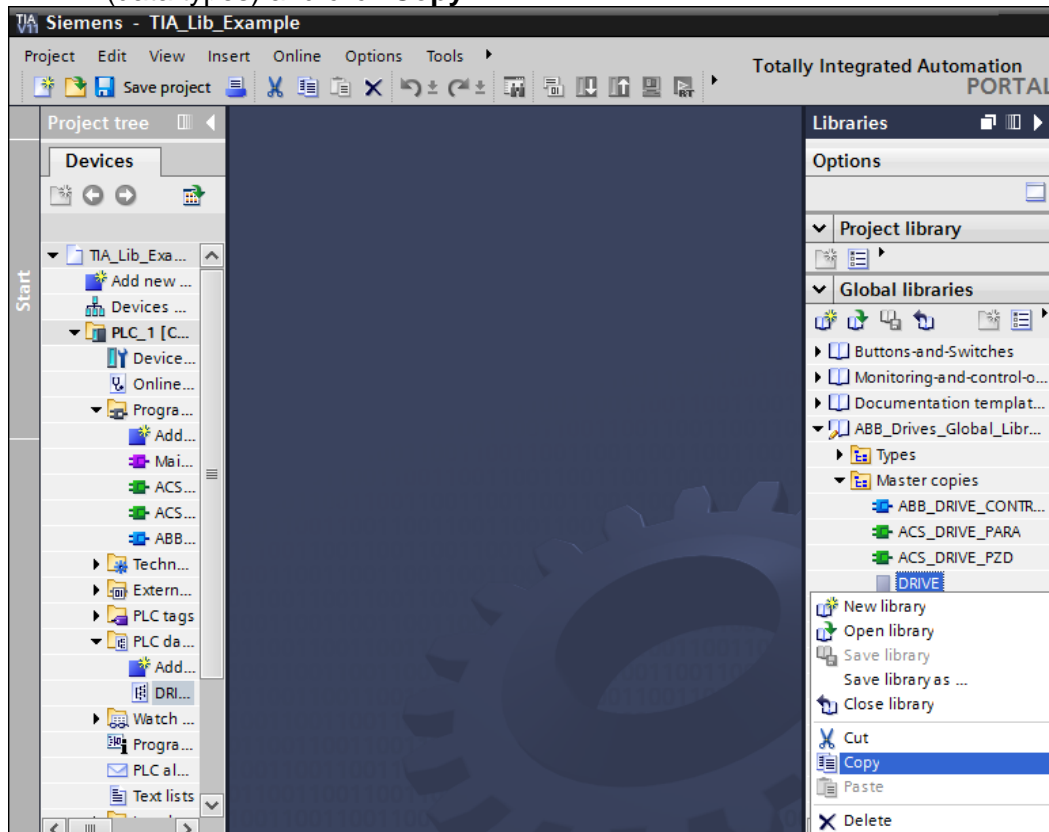
Figure 82 Drive

6. In the Project tree pane, right-click **PLC data types** and click **Paste**.
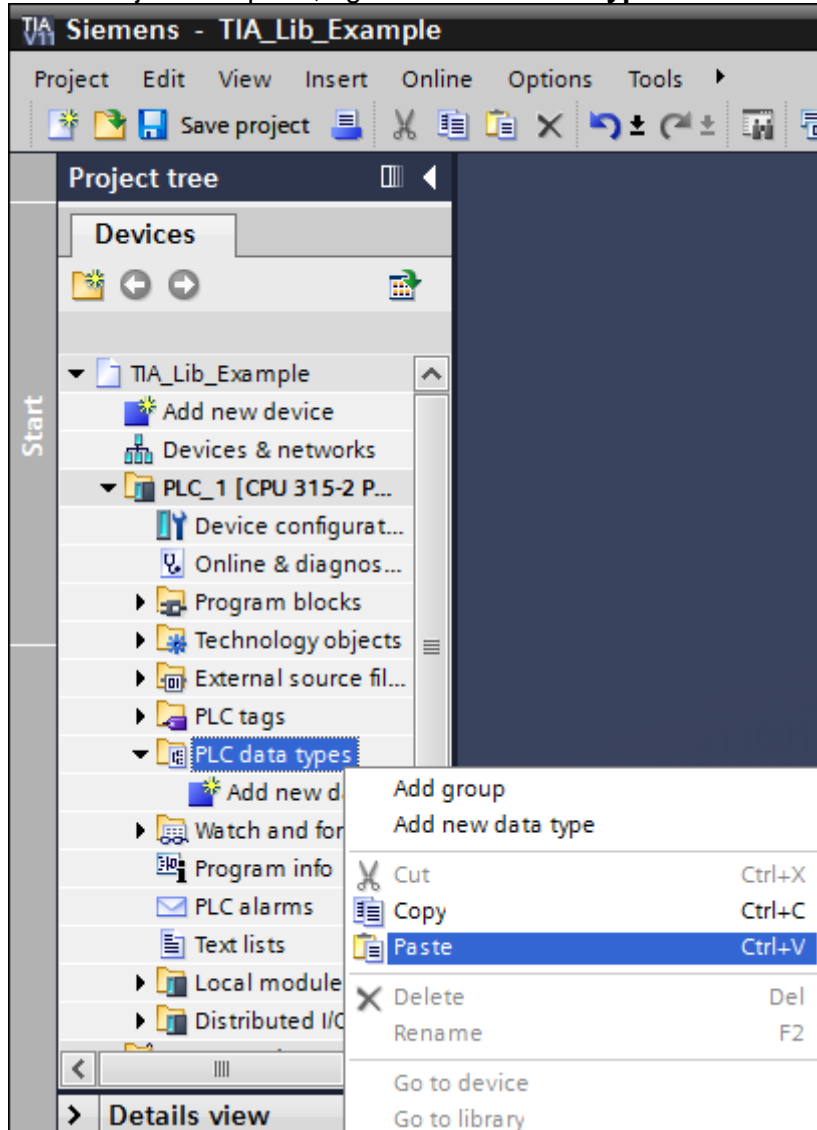
Figure 83 Data types

Adding PLC tags to S7-300

1. In the Project tree pane, expand PLC tags and double-click **Add new tag table** to create symbols.
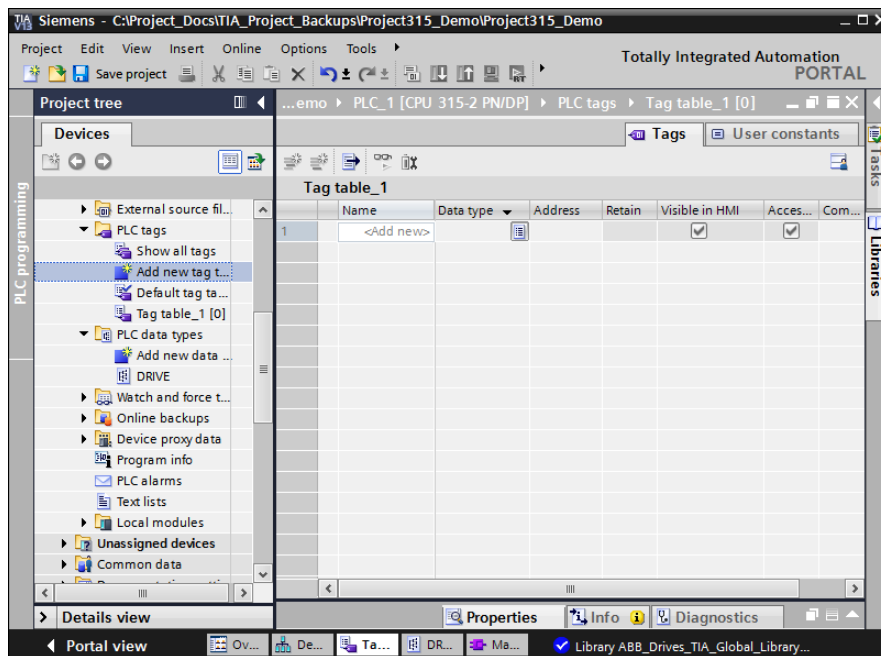
Figure 84 New tag table

2. Add tag Name, Data type and Address to connect block inputs and outputs.
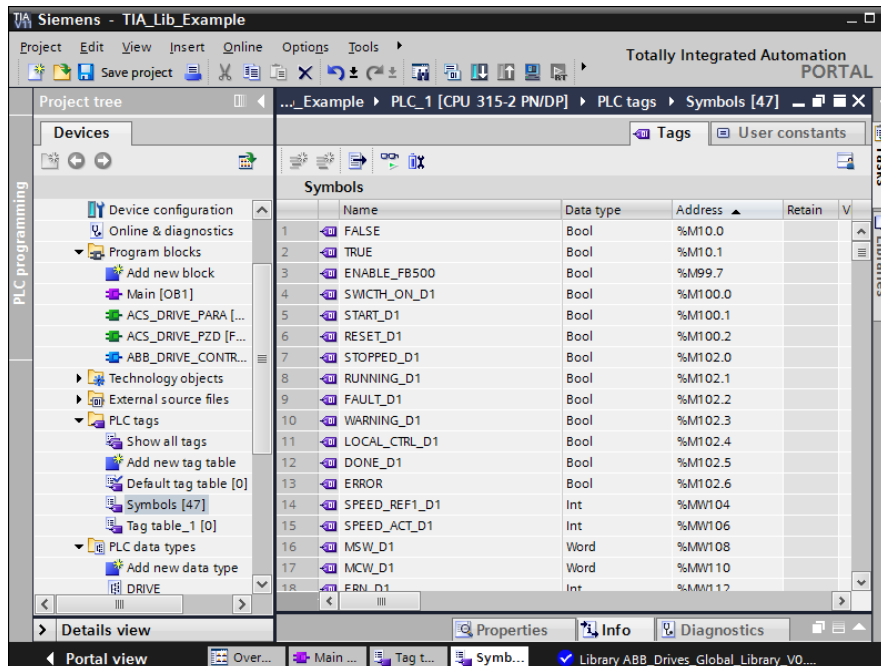
Figure 85 PLC tags

For information on Data types of tags, see sections *Block variables and data types* (page *53*) and *Block variable and data types for FC501*(page *59*) *Block variables and data types for FC500* (page *62*).

FB500 ABB_DRIVE_CONTROL_FB

FB500 is used to control the drive (start, stop, reset, emergency stop, speed reference, etc.).

Note! The following description is an example, there are different ways to use the Control function block.

1. In the Program blocks, double-click **Main [OB1]** and create memory bits for Logic true and Logic false.
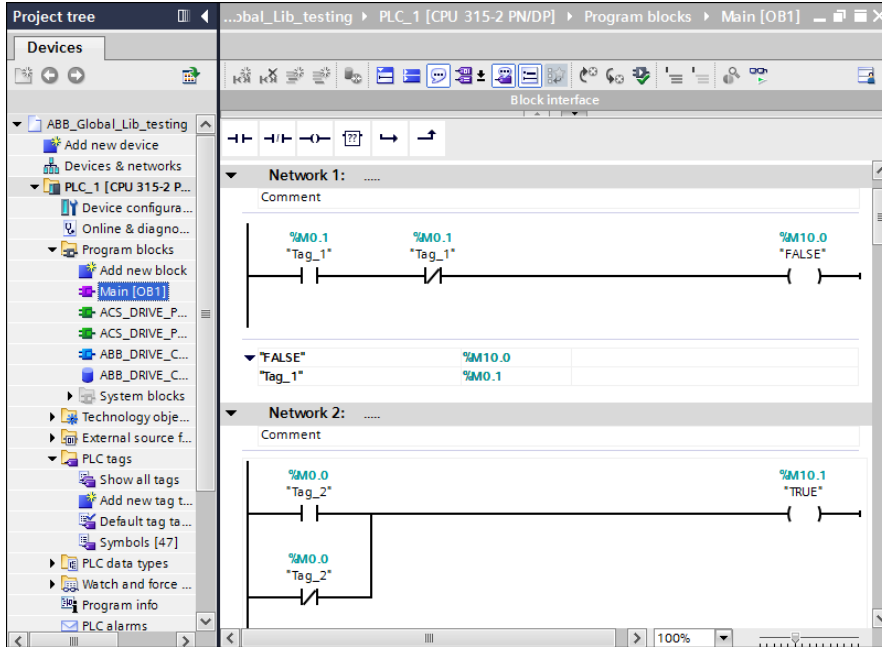
Figure 86 Logic TRUE/FALSE

2. Right-click in the next free network and choose **Insert empty box**.
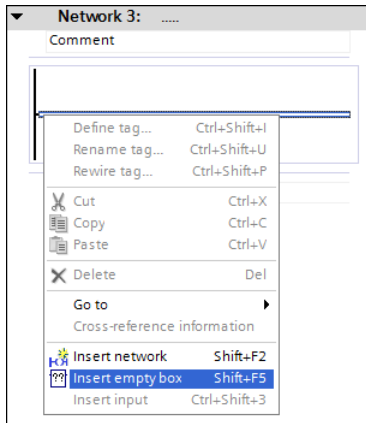
Figure 87 Inserting empty box

3. Write FB500 in the block field to create an instance of FB500.
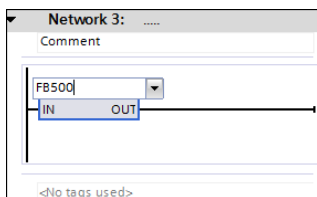
Figure 88 Instance of FB500

4. Create a **unique** Data block for the control of the drive by enabling **Manual** option and by selecting the Number.
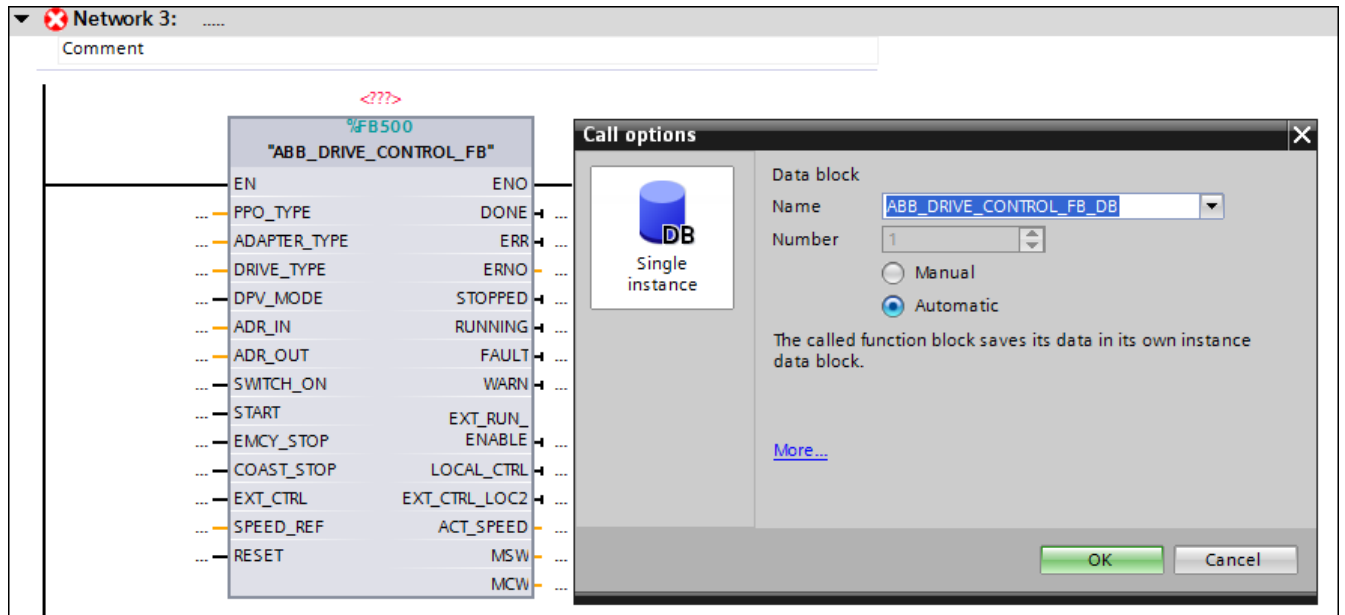
Figure 89 FB500 DB1

Drive status and more will be stored in this Data block. Since it is a new Data block, you have to generate it by clicking **OK** in the Call options window.

Note! If you add more drives to the program, make sure to create new unique Data blocks.

5. Connect the block inputs and outputs to variables according to your application.
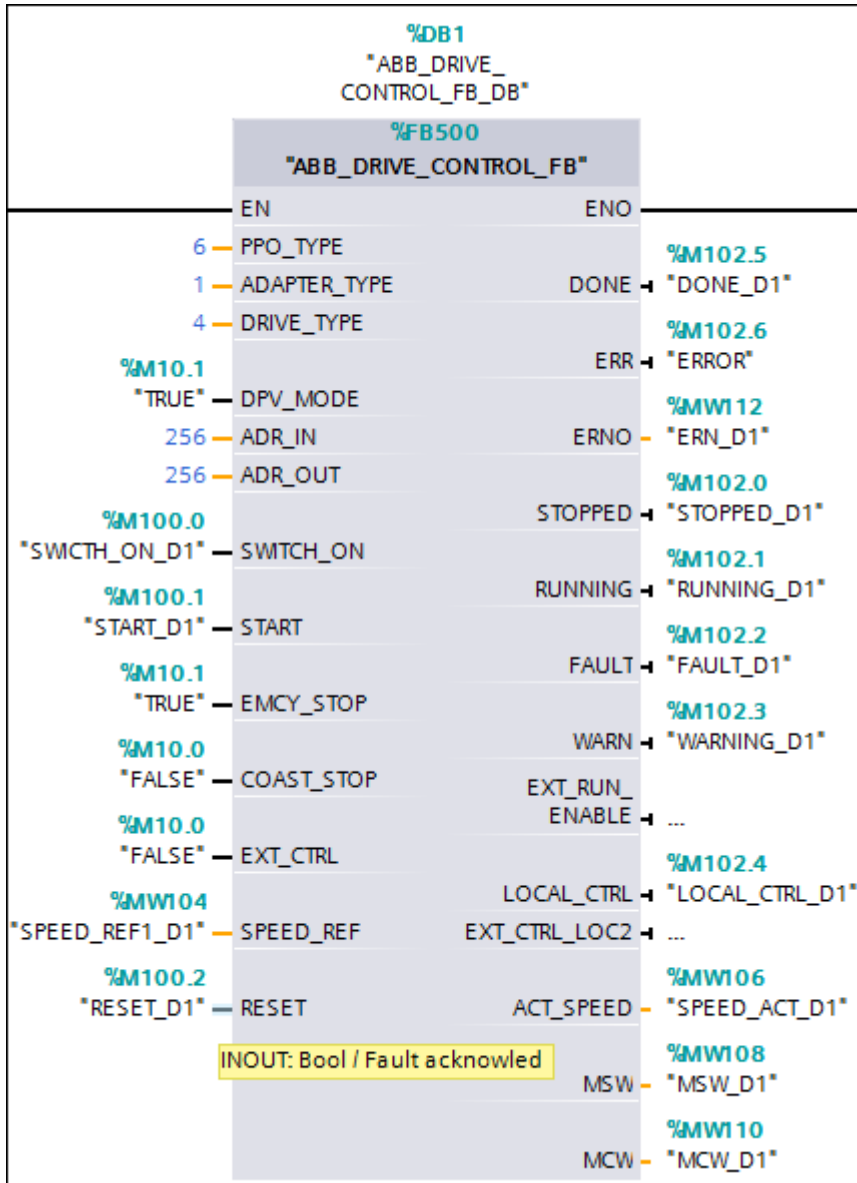
Figure 90 Assigned variables

Note! Select a block input or output and see more information in the Info.

For more information on block variables and data types, see *Block variables and data types* (page).

6. Click **Save** to save to the project.
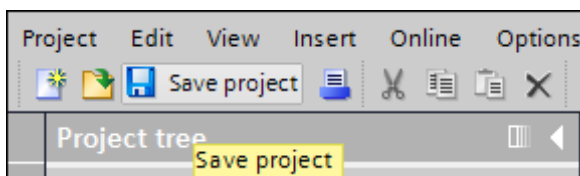
Figure 91 Save project

Adding watch and force tables

1. In the Project tree, expand **Watch and force tables** and double-click **Add new watch table** and provide a suitable name and click **OK**.
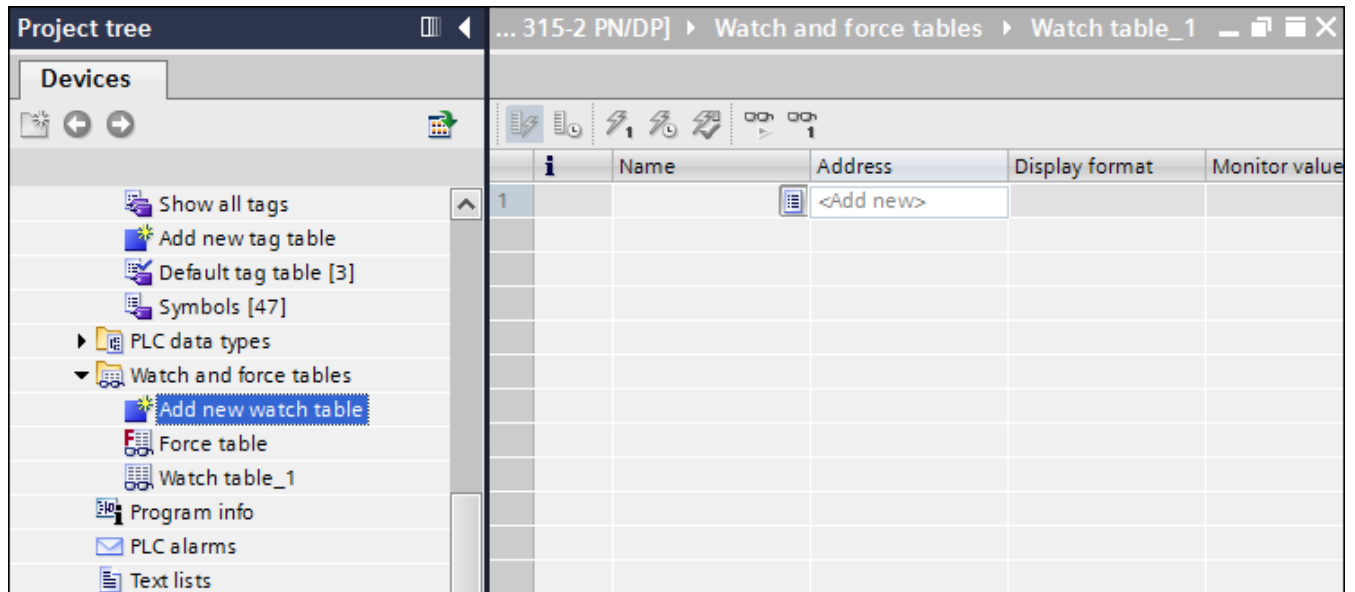
Figure 92 Watch and force tables

2. Open Watch table and add your required variables (for example, FB500 variables). It is possible to copy directly from the PLC tags.
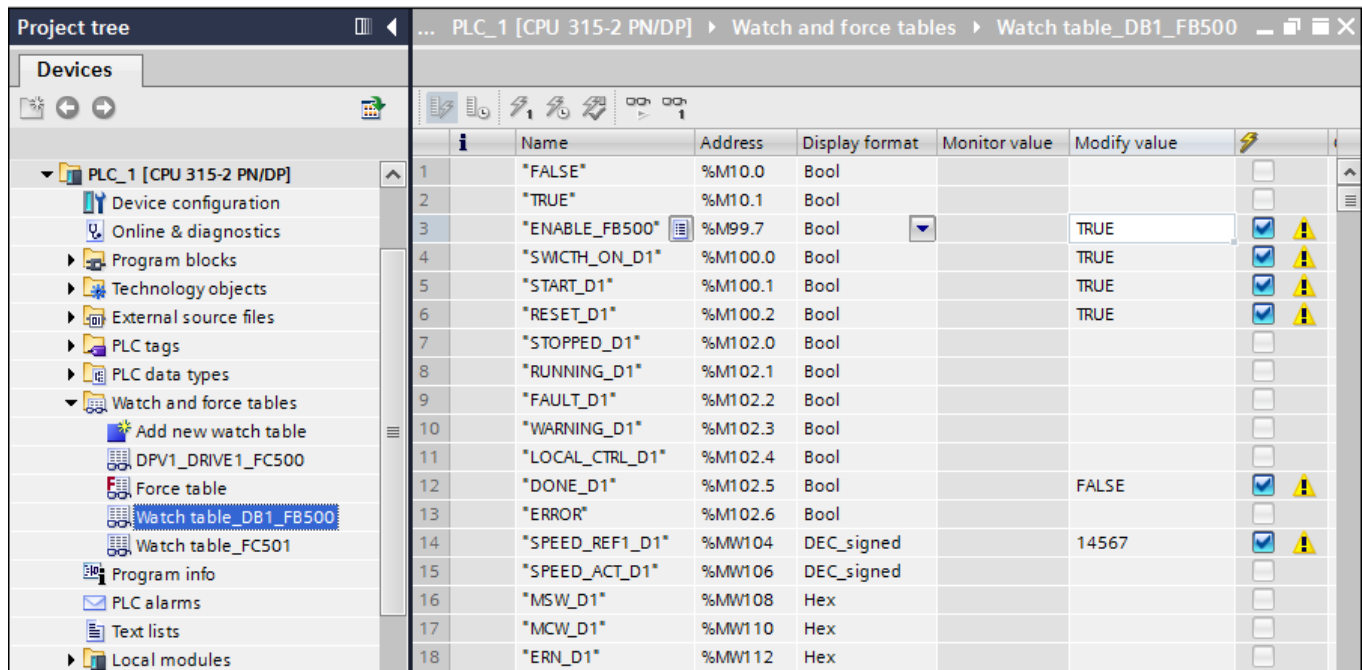
Figure 93 Watch table FB500

3. Click to display the online values.
4. Click to update modified values.

FC501 ACS_DRIVE_PZD

FC501 is used to send additional process data between the PLC and the drive.

Depending on actual PPO type, a certain number of data words (PZDs) are exchanged. PPO types 1 and 3 have only 2 PZDs in each direction, so FC501 (*ACS_DRIVE_PZD*) is not useful for those types. PPO types 2 and 4 have 6 PZDs in each direction, so a part of FC501 (PZD3 to PZD6) is useful here. PPO types 5 and 6 have 10 PZDs in each direction, so the full range of FC501 is useful for those types.

1. Right-click in the next free network and choose **Insert empty box**.
2. Write **FC501** in the block field to create an instance of FC501.
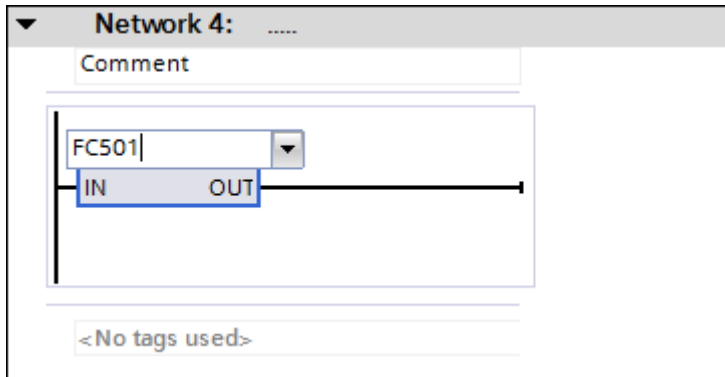
Figure 94 FC 501

3. Connect the FC501 block inputs and outputs. All block inputs and outputs need to be connected.

   In the example below, the Instance Data block of FC501 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB* and an ACS355 with FPBA-01 PROFIBUS module is been used.
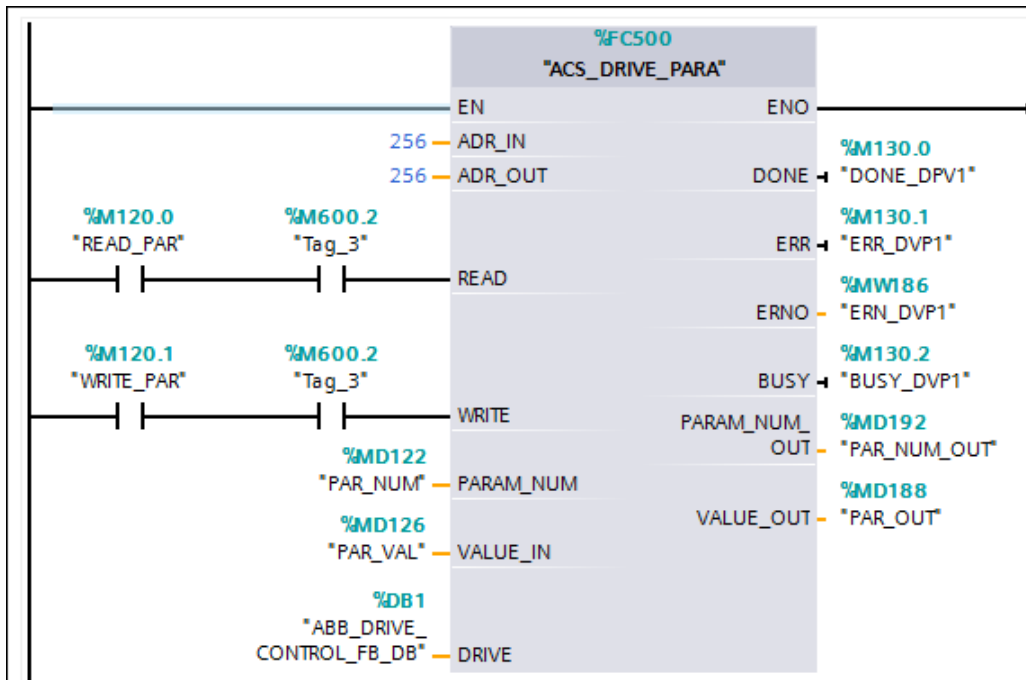

Figure 95 Example FC501

   For more information on block variables and data types, see section *Block variable and data types for FC501* (page *59*).

If you want to scale Process Data values, set the scaling values in the program before calling FC501 *ACS_DRIVE_PZD* block. See example below where PZD3 is scaled with the value 100.
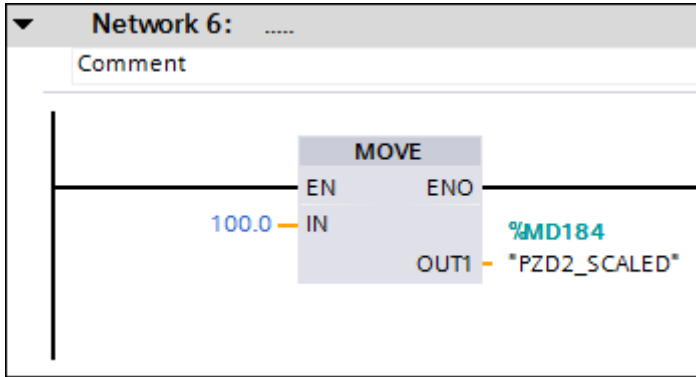
Figure 96 Scaling process data values

For more information on mapping fieldbus process data parameters of the drive according to the application, see section *Scalable variables and data types* (page *61*).

FC500 ACS_DRIVE_PARA

FC500 is used to read/write extra parameters between the PLC and the drive.

1. Right-click in the next free network and choose **Insert empty box**.
2. Write **FC500** in the block field to create an instance of FC500 (*FC500 ACS_DRIVE_PARA* is included in the ABB library ABB_DRIVE_LIB).
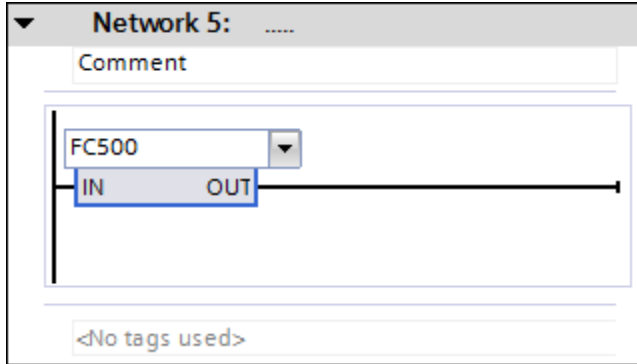
Figure 97 FC500

3. Connect the FC500 block inputs and outputs. All block inputs and outputs need to be connected.

For more information on block variables and data types, see section *Block variables and data types for FC500* (page *62*).

In the example below, the Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB* and an ACS355 with FPBA-01 PROFIBUS module has been used.

Figure 98 Example FC500

4. In the Project tree, right-click PLC_1 (CPU) and select **Properties** for executing read/write parameter value (0 -> 1 (executed on positive edge).
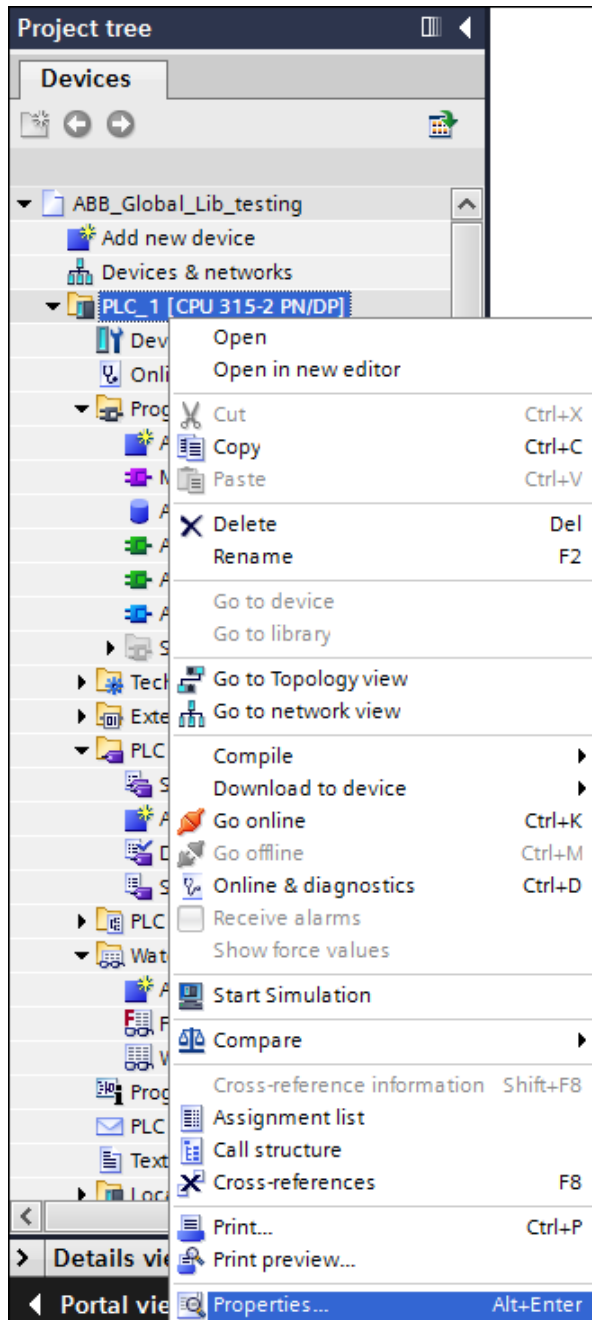
Figure 99 Clock memory properties

5. In the PLC_1 (CPU) window, click **Clock memory** and enable **Clock memory**, assign memory address (example, 600) and click **OK**.
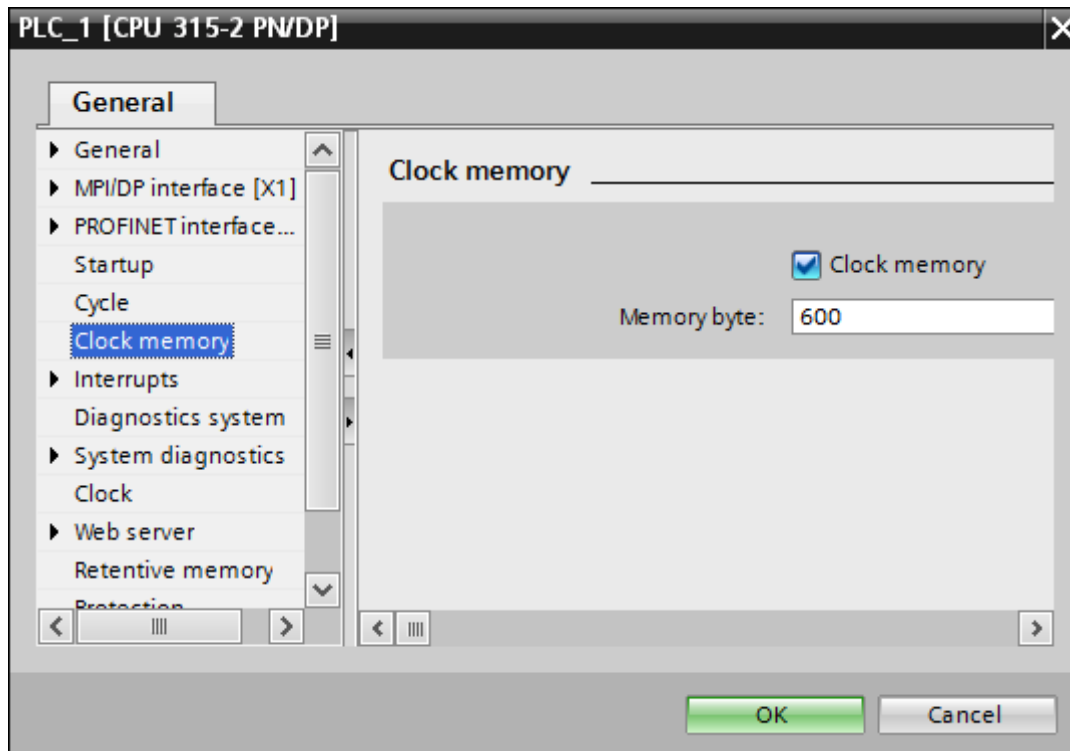
Figure 100 Clock memory

Error codes for FB500 and FC500

■ The DP-V0 ERROR table

```
Task cannot be executed, follow by error number
0   =  Illegal parameter number
1   =  Parameter value cannot be changed
2   =  Lower or upper limit violated
3   =  Erroneous subindex
4   =  No array
5   =  Incorrect data type
6   =  Setting not allowed (can only be reset)
7   =  Descriptive element cannot be changed
9   =  Descriptive data not available
11  =  No parameter changes rights
15  =  Text array not available
17  =  Task cannot be executed due to operating status
       (e.g. parameter is currently read-only)
18  =  Other error
101 =  Vendor specific error
102 =  Request not supported
103 =  Request cannot be completed due to communication error
110 =  Failure during write to non-volatile memory
111 =  Request aborted due to time-out
120 =  Parameter cannot be mapped to PZD
       (size mismatch or non-existent
121 =  Parameter cannot be mapped to PZD (end of memory)
122 =  Parameter cannot be mapped to PZD (multiple PZD write)
130 =  Cannot map Control Word bit
       (parameter 933-937, e.g. double mapping of bits)
140 =  Cannot change mode to TORQUE (frequency is used)
150 =  Internal buffer overflow
160 =  Internal communication error
```

■ The DP-V1 ERROR table

Error #	Meaning	Used at
00h	Impermissible parameter number	Access to unavailable parameter
01h	Parameter value cannot be changed	Change access to a parameter value that cannot be changed
02h	Low or high limit exceeded	Change access with value outside the limits
03h	Invalid subindex	Access to unavailable subindex
04h	No array	Access with subindex to non-indexed parameter
05h	Incorrect data type	Change access with value that does not match the data type of the parameter
06h	Setting not permitted (can only be reset)	Change access with value unequal to 0 when this is not permitted
07h	Description element cannot be changed	Change access to a description element that cannot be changed
09h	No description data available	Access to unavailable description (parameter value is available)
0Bh	No operation priority	Change access rights without rights to change parameters
0Fh	No text array available	Access to text array that is not available (parameter value is available)

Error #	Meaning	Used at
11h	Request cannot be executed because of operating mode	Access is temporarily not possible for reasons that are not specified in detail
14h	Value impermissible	Change access with a value that is within limits but is not permissible for other long-term reasons (parameter with defined single values)
15h	Response too long	The length of the current response exceeds the maximum transmittable length
16h	Parameter address impermissible	Illegal value or value that is not supported for the attribute, number of elements, parameter number or sub-index, or a combination
17h	Illegal format	Write request: Illegal format or format of parameter data that is not supported
18h	Number of values inconsistent	Write request: Number of values of parameter data does not match number of elements at the parameter address
65h ... FF	Manufacturer-specific error area	–
65h	Vendor-specific error	Vendor-specific error
66h	Request not supported	Request not supported
67h	Communication error	Request cannot be completed because of communication error
6Eh	Non-volatile error	Failure during write to non-volatile memory
6Fh	Time-out error	Request aborted because of timeout
78h	PZD map failure	Parameter cannot be mapped to PZD (size mismatch or non-existent)
79h	PZD memory failure	Parameter cannot be mapped to PZD (out of memory)
7Ah	Multiple PZD map	Parameter cannot be mapped to PZD (multiple PZD write)
82h	Control word bit map	Cannot map Control word bit (parameter 933...937, e.g. double mapping of bits)
8Ch	Set torque mode error	Cannot change mode to TORQUE (frequency is used)
90h	Illegal Request ID	The request ID of the response is illegal
96h	Internal buffer	Buffer overflow
A0h	Internal communication	Communication error between module and drive

Configuring S7-1200 and S7-1500 PLC

In the following example, S7-1200 PLC is configured. Similarly, you can configure S7-1500 PLC.

To configure TIA Portal with the PLC and ABB Drive libraries, follow these steps:

1. To configure S7-1200 PLC, follow steps 1-3 in *Configuring S7-300 PLC*.

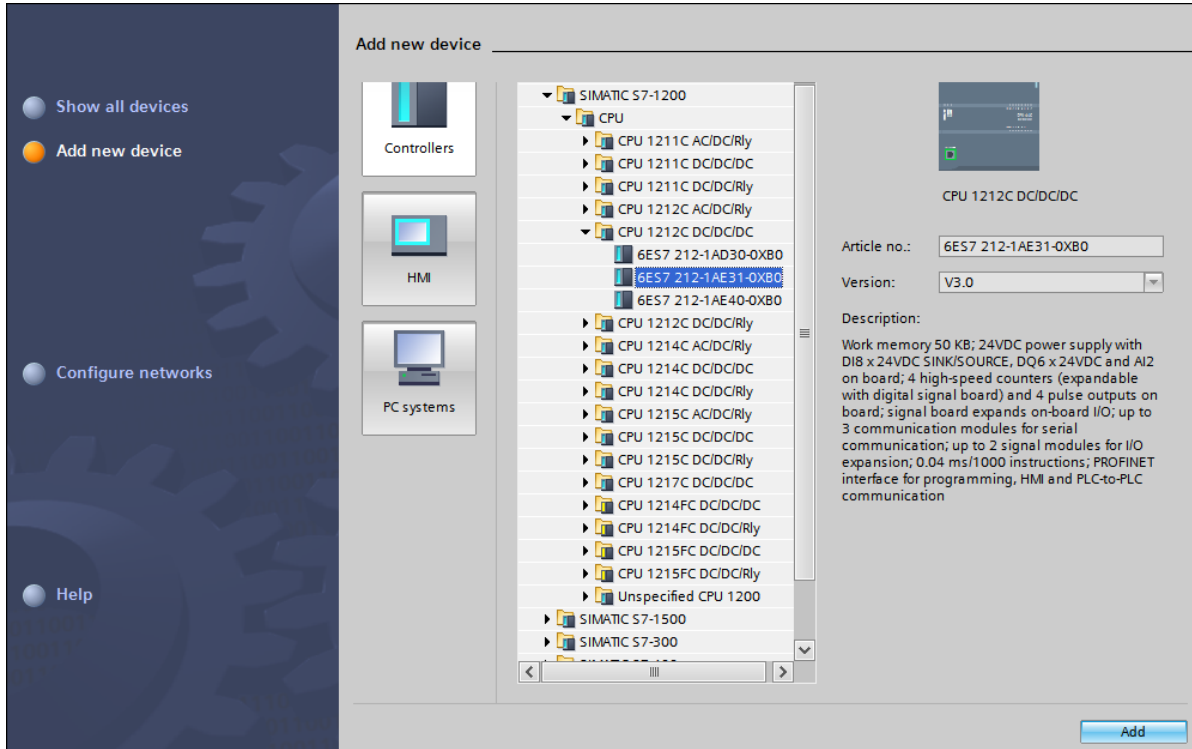2. Select **Add new device** in the left pane and select the required PLC and then click **Add**.

Figure 101 Adding new device S7-1200
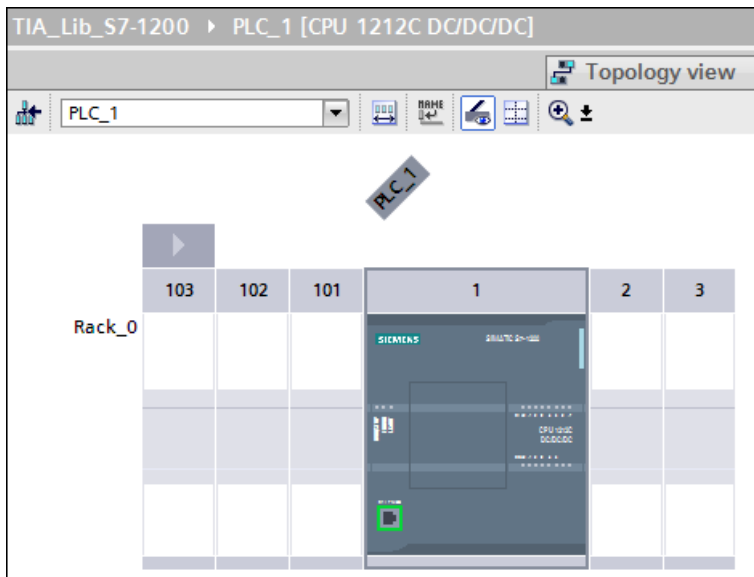
New PLC device is added to hardware configuration.

Figure 102 PLC device S7-1200

3. Select the required slot in the rack and drag-and-drop the PROFIBUS module from the Hardware catalog to the slot.
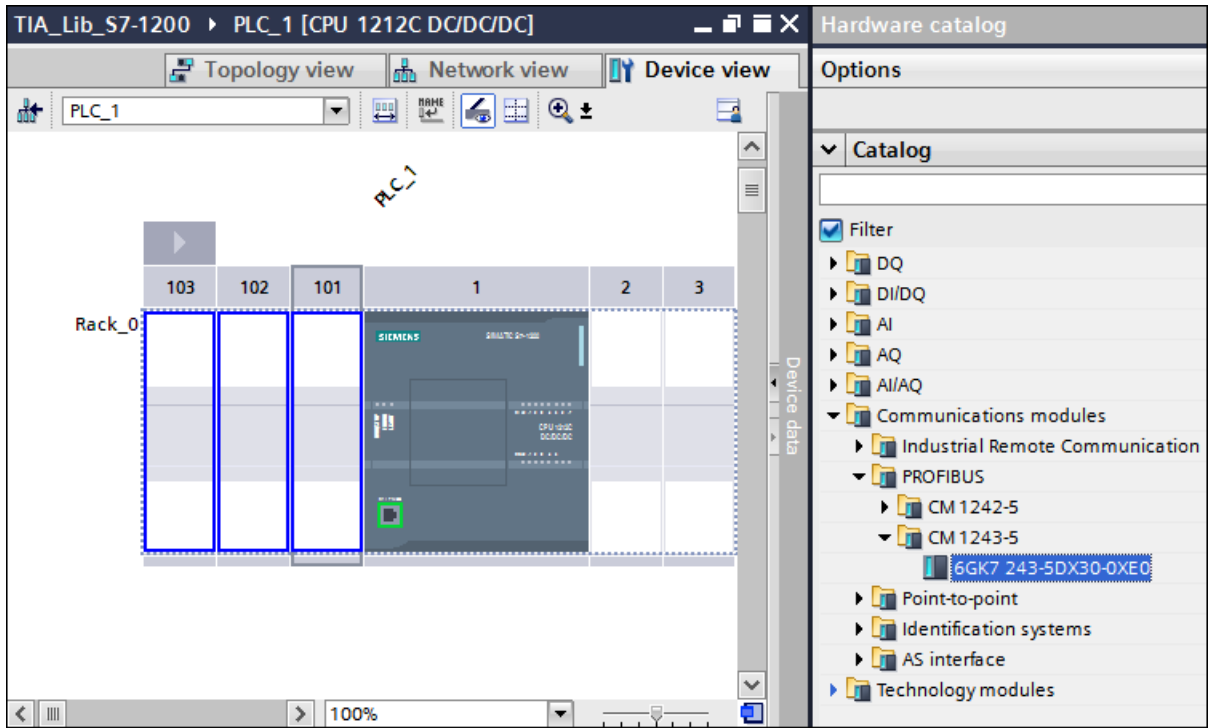
Figure 103 PROFIBUS configuration

PROFIBUS module is added to the slot.
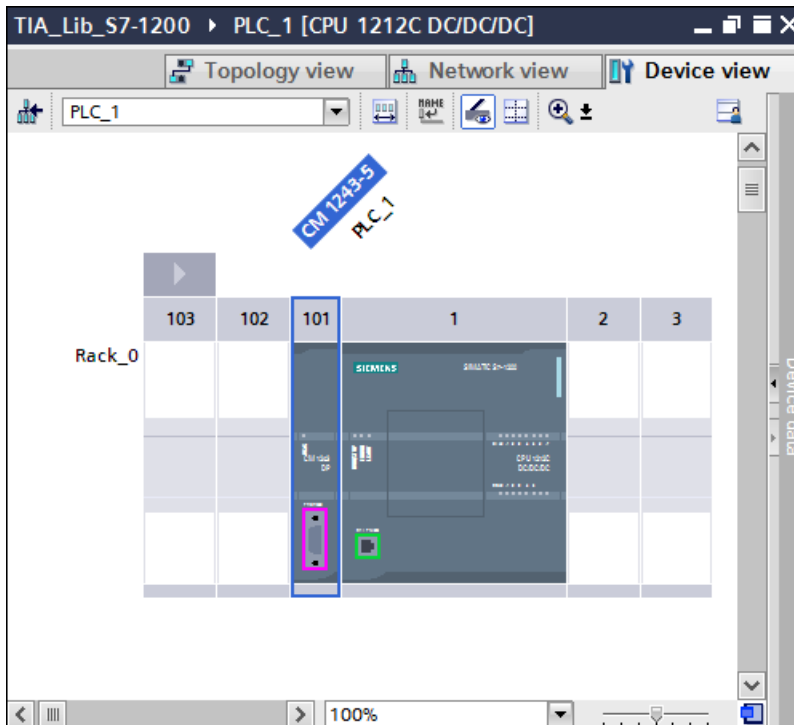
Figure 104 PROFIBUS module

Note: The user can select the desired version of the PROFIBUS module in the information pane.
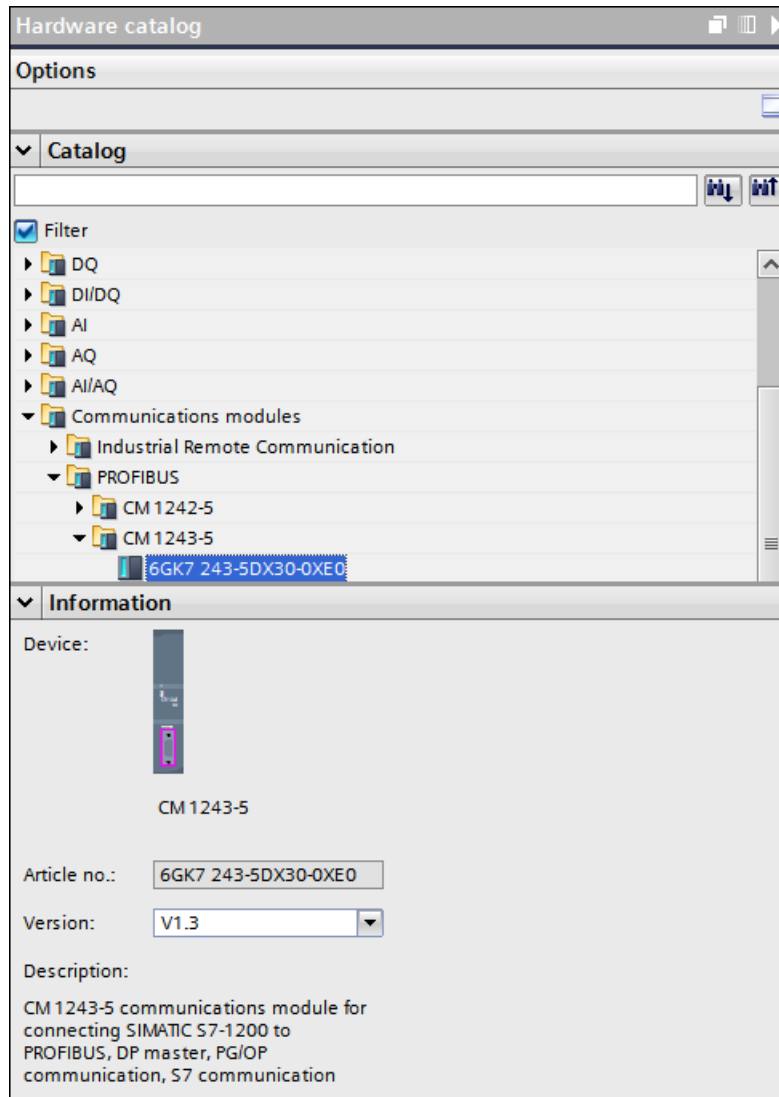
Figure 105 PROFIBUS module version info

4. In the Device view, select **PLC_1** (PROFINET interface) and in the General tab, select **Ethernet addresses** and click **Add new subnet** to add the subnet and then set IP Address in the IP protocol.
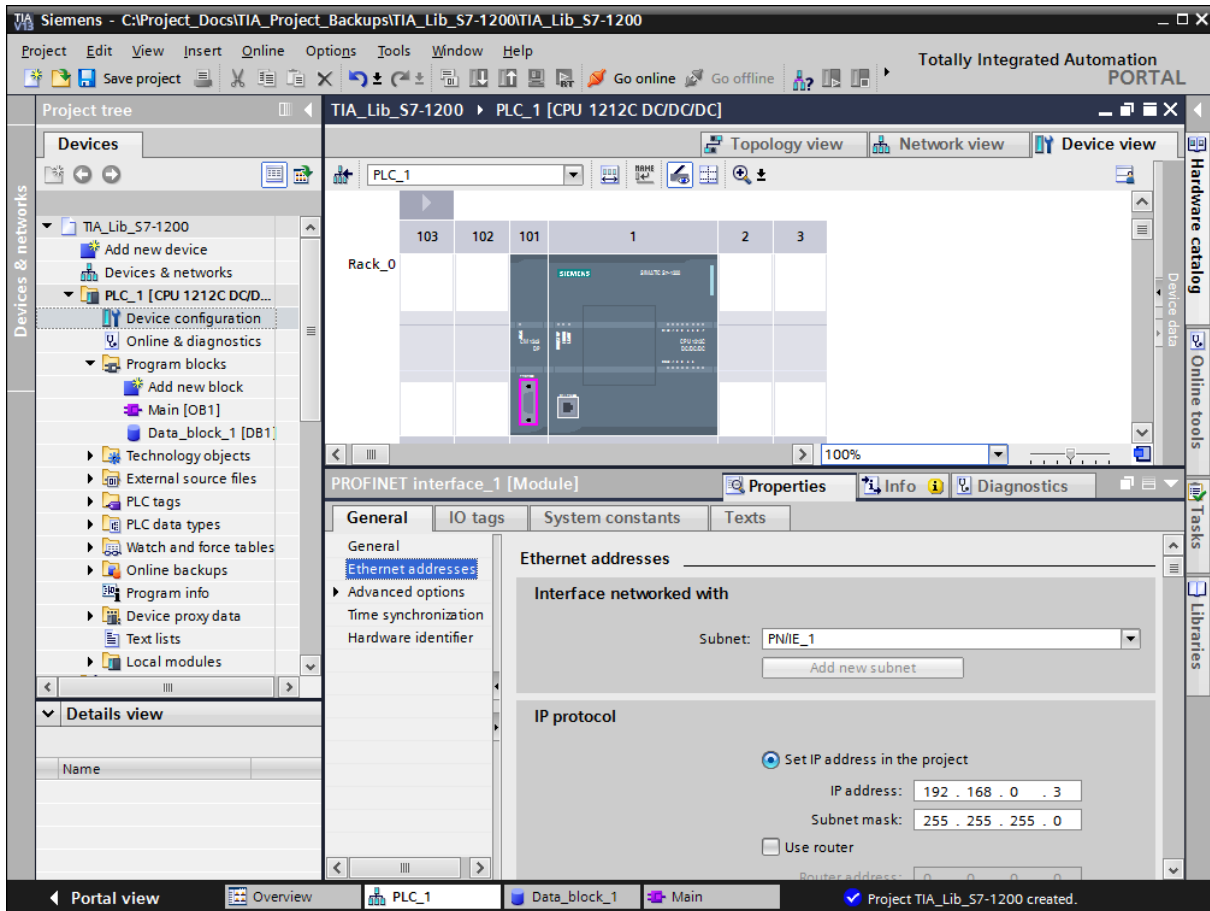
Figure 106 Assign IP protocol

5. In the Device view, select **PLC_1 (DP interface)**. In the General tab, select **PROFIBUS address** and click **Add new subnet** to add the subnet and then assign PROFIBUS address.
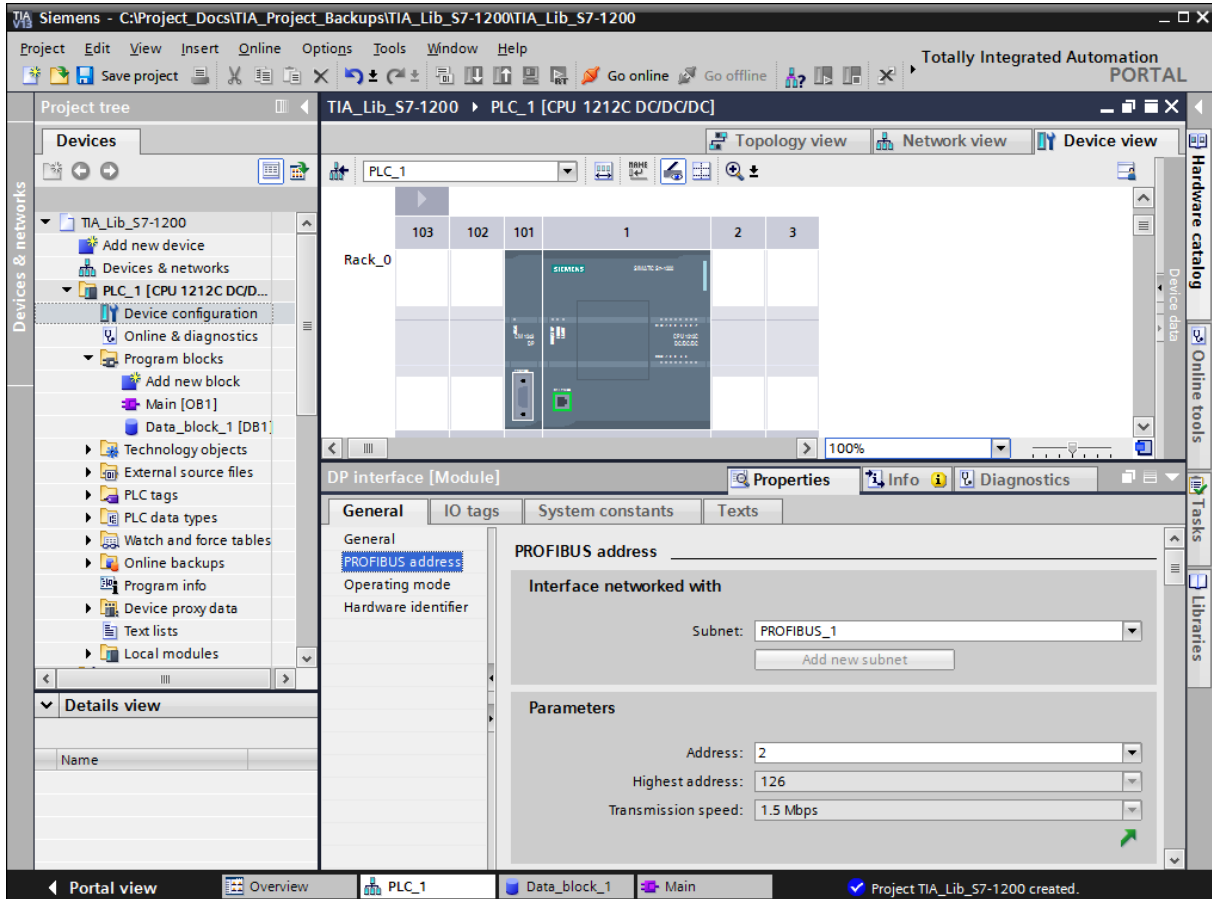
Figure 107 DP interface

6. After **configuring** PROFINET and PROFIBUS interfaces in TIA portal, click **Network view** to view the PLC connections.
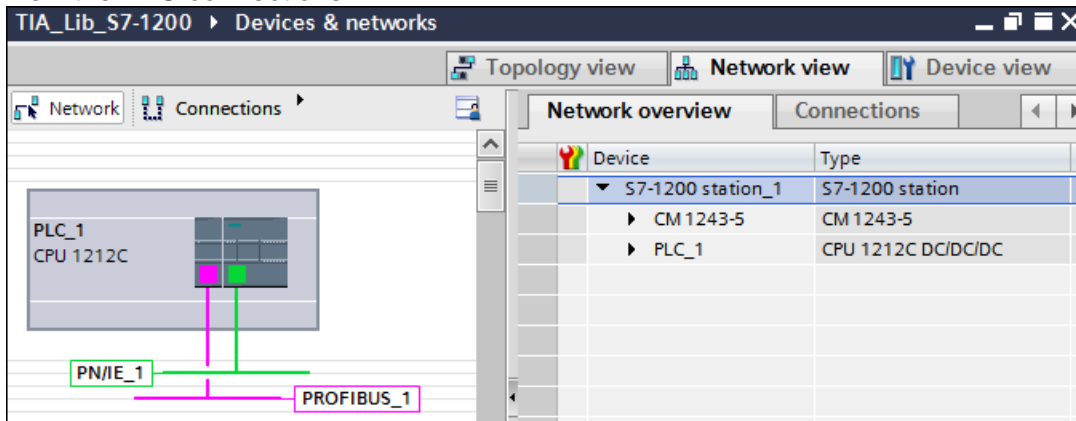
Figure 108 PROFIBUS and PROFINET configurations

Installing GSD file for S7-1200 PLC

To install GSD file, see section *Installing GSD file for S7-300 PLC* (page *75*).

PLC libraries for S7-1200

Adding ABB drives to PROFIBUS DP line

1. Launch TIA portal and click **Open the project view** in the **Start** options.

   Note: After installing GSD files, ABB drives are added to the hardware catalog of TIA portal.

2. In the Project tree pane, double-click **Device & networks**.
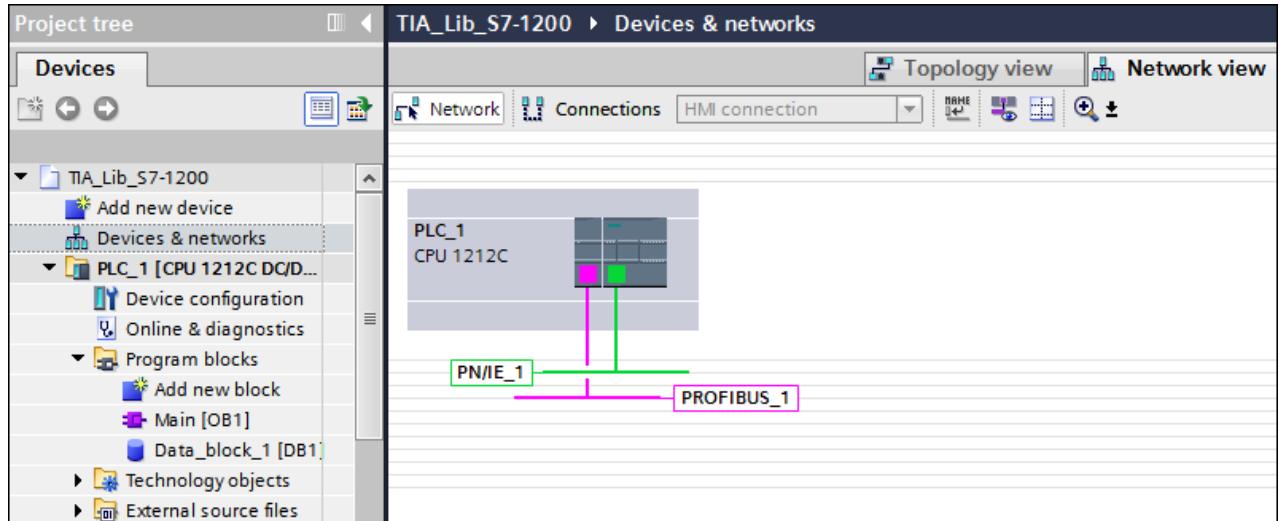
Figure 109 Devices and network for S7-1200

3. From the hardware catalog pane, drag and drop the required ABB Drive to the PLC.
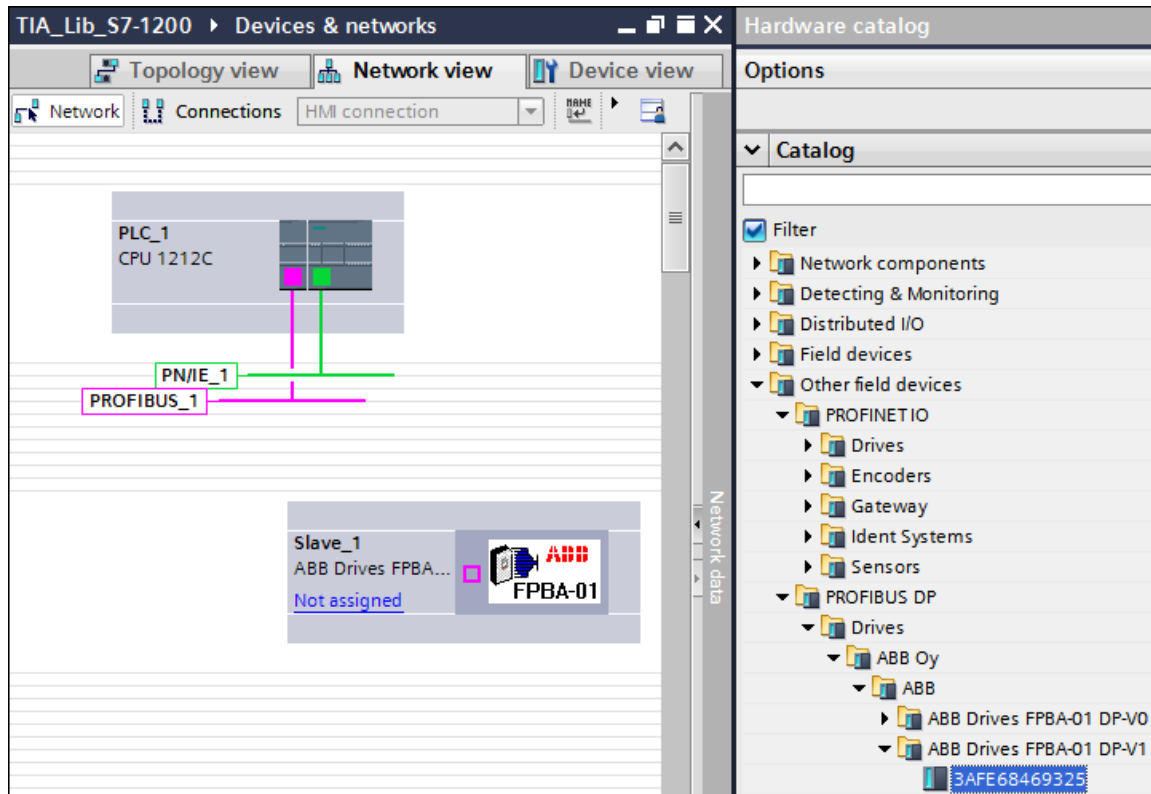
Figure 110 ABB drive and PLC

4. Double-click Slave_1 to navigate the device view for adding required PPO type.
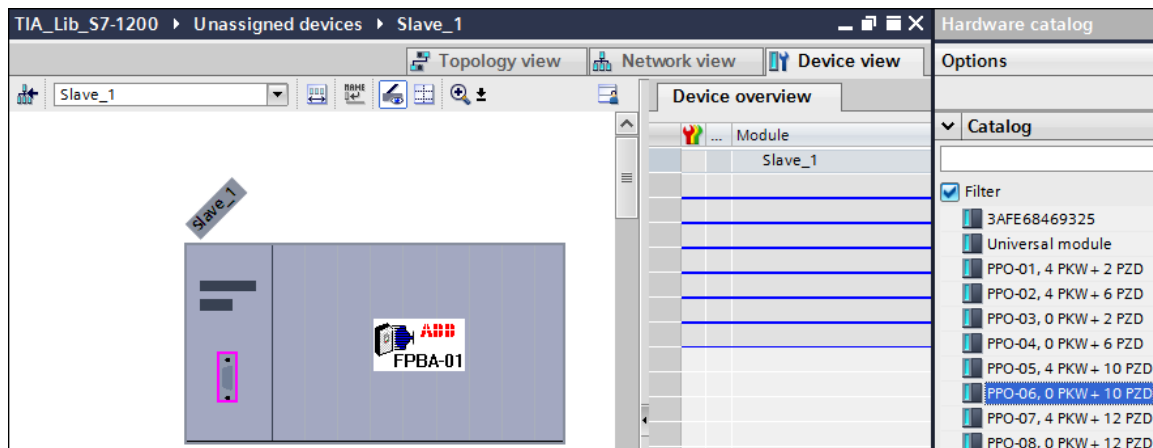
Figure 111 PPO type for S7-1200

5. Double-click on the required PPO type in the catalog to add in the Device overview.
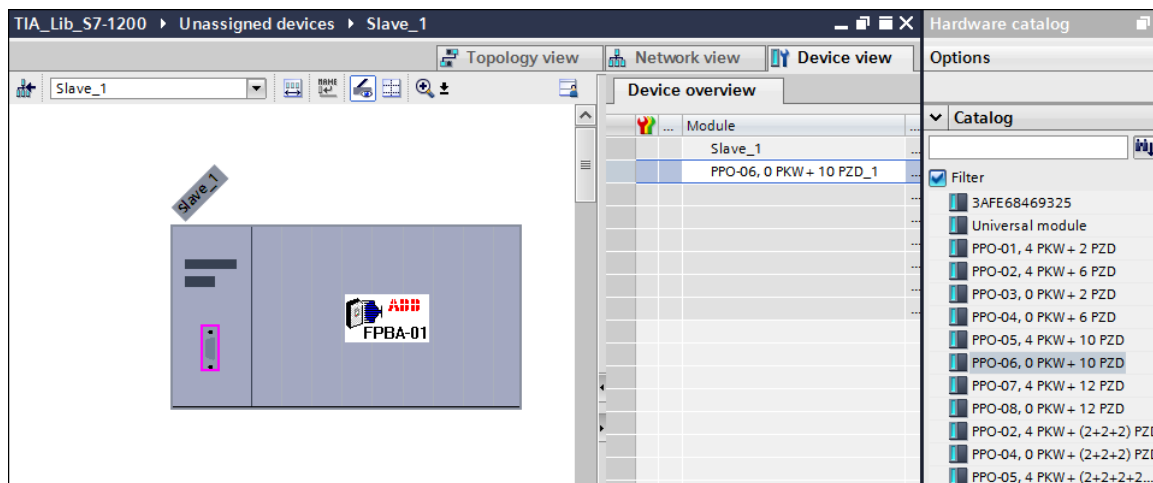
Figure 112 PPO type

6. Click **Slave_1** and in the General tab, select **PROFIBUS address** and select the already configured PROFIBUS interface and then set PROFIBUS address for slave.
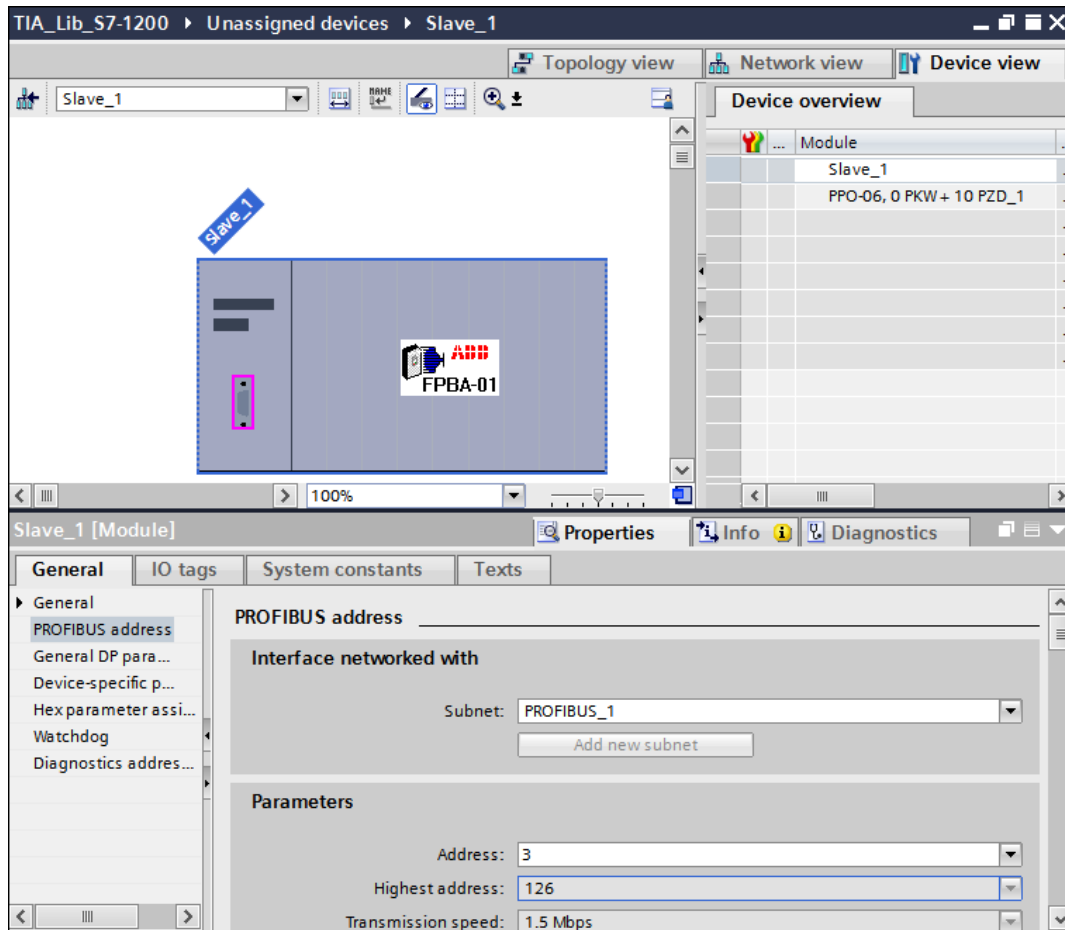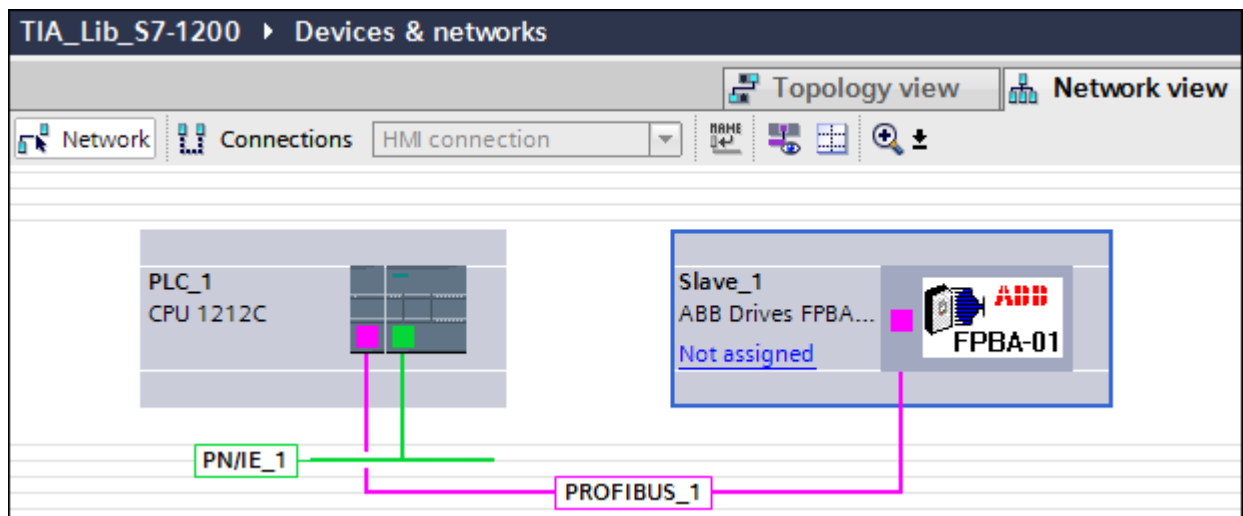

Figure 113 PROFIBUS address

7. Click **Network view** to see the configured Slave_1.

Figure 114 Configured slave

8. In the ABB drive, click **Not assigned** and select **PLC_1.CM 1243-5.DP interface** to assign PLC to the ABB drive.
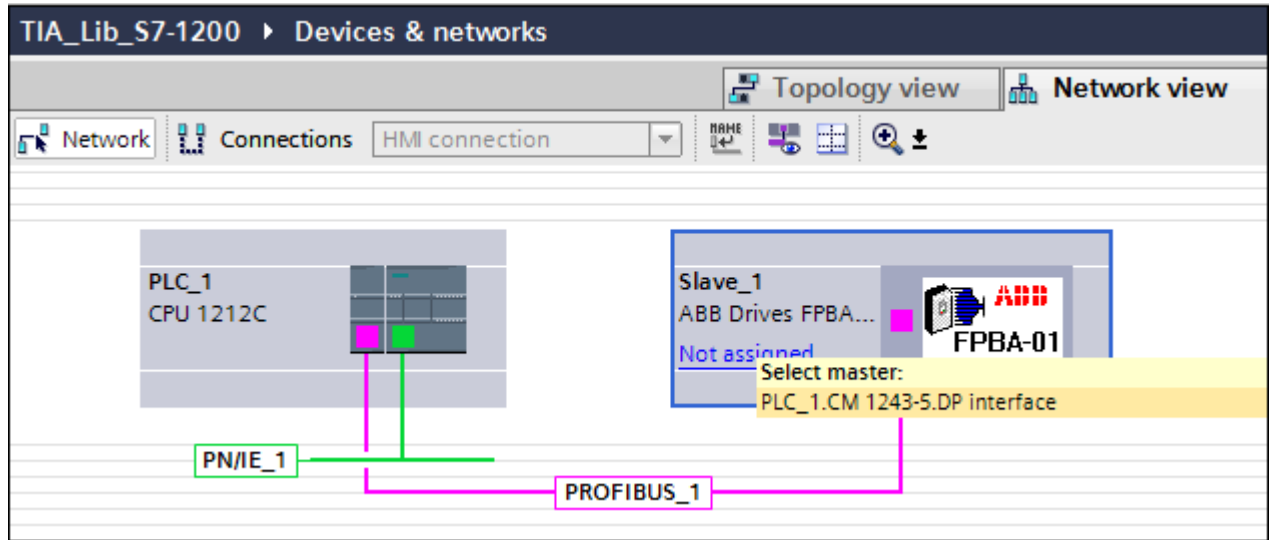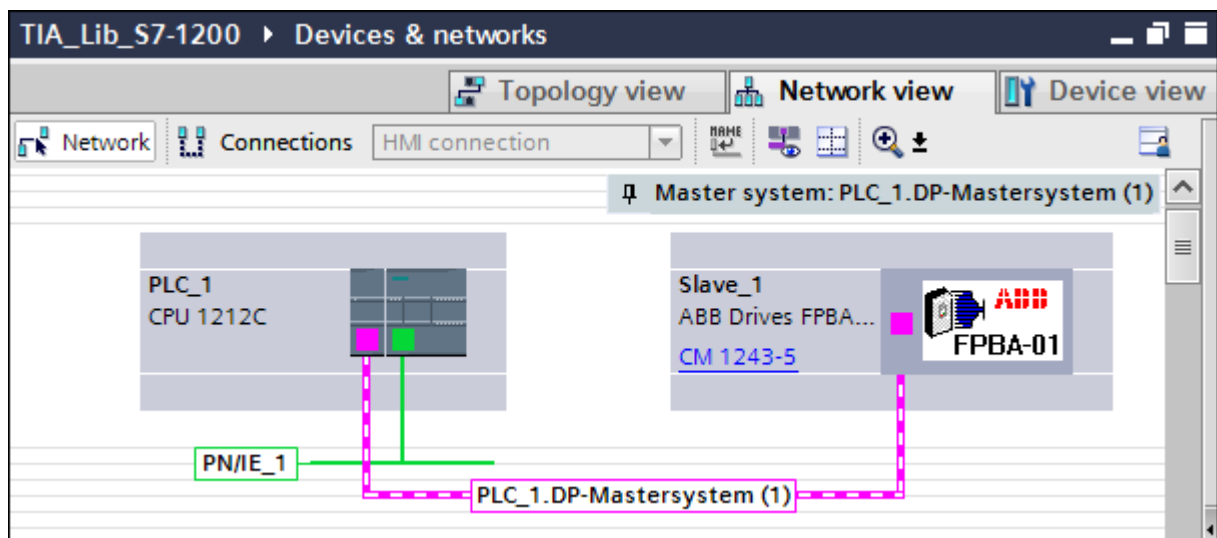

Figure 115 Assign PLC to ABB drive

The selected PLC is assigned to the ABB drive.

Figure 116 PLC assigned to ABB drive

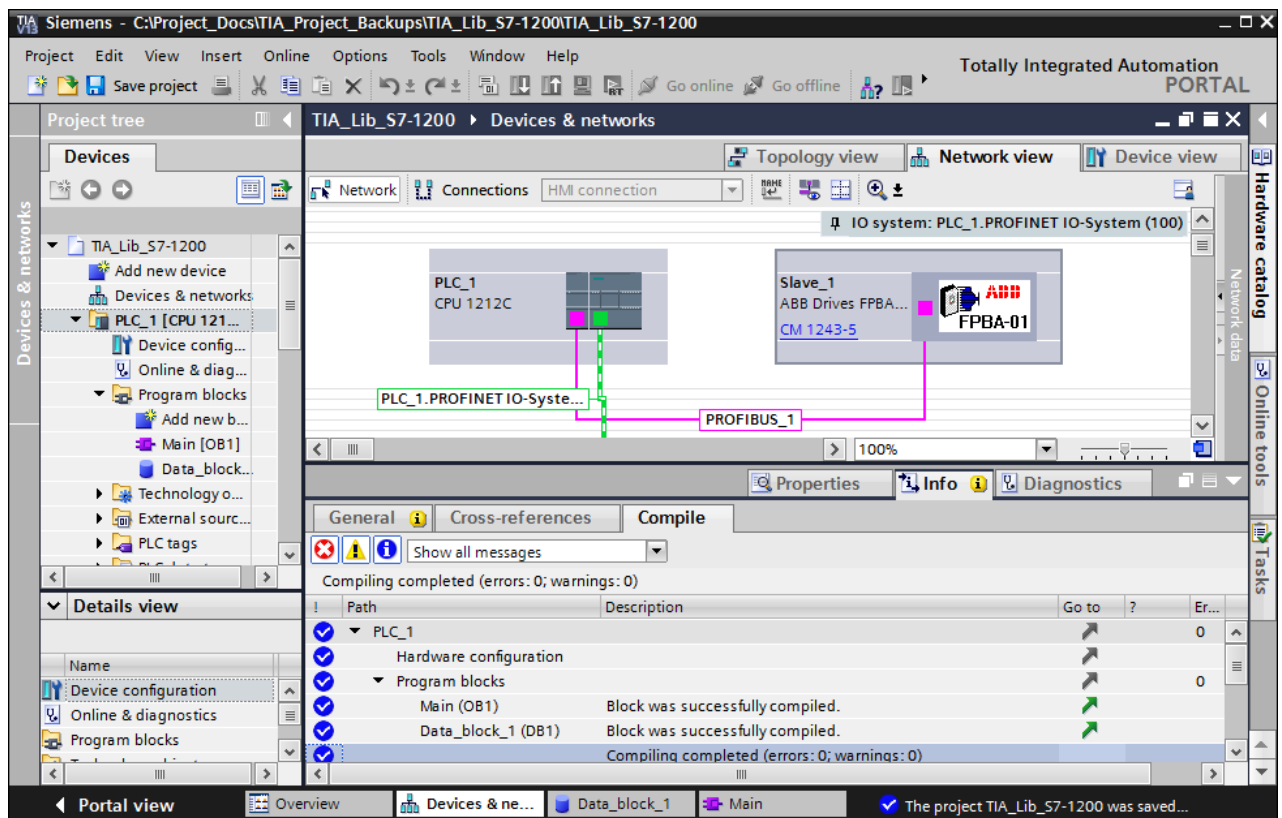9. In the Project tree pane, select **PLC_1** and then click ⊞ to compile.

Figure 117 Compile

Note: Before downloading the configuration to PLC, check the PC IP address. For more information, see section *Configuring PC IP address* (page *64*).

Adding ABB drives to PROFINET

1. Follow steps 1 and 2 of *Adding ABB drives to PROFIBUS DP line.*
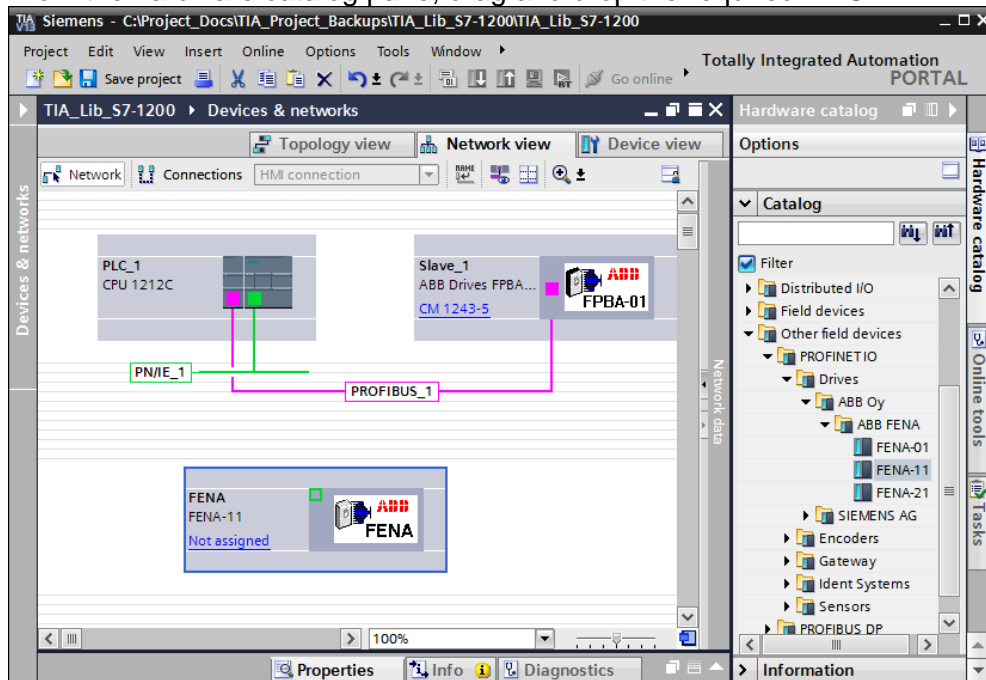2. From the hardware catalog pane, drag and drop the required PROFINET ABB Drive to the PLC.

Figure 118 FENA drive

3. In the ABB drive, connect PLC to the PROFINET interface.
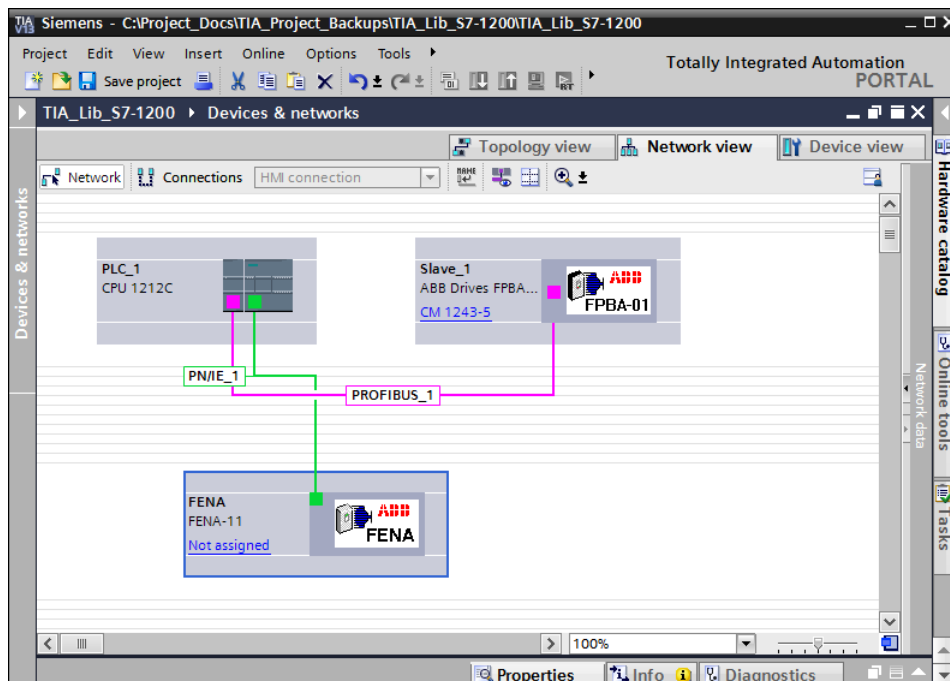

Figure 119 Connecting PLC and FENA drive

4. Select the ABB drive (FENA) and click Device view to configure PROFNET IP address and device name.

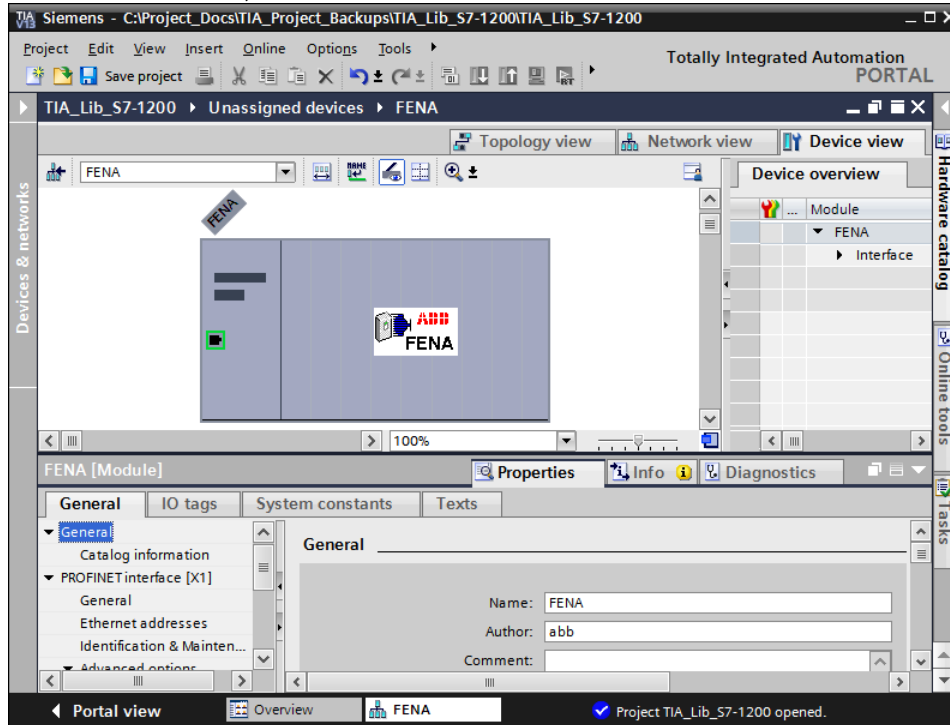- In the General tab, click General to add the device name.

Figure 120 Device name

- In the PROFINET interface [X1], click Ethernet address to add Subnet and IP address.
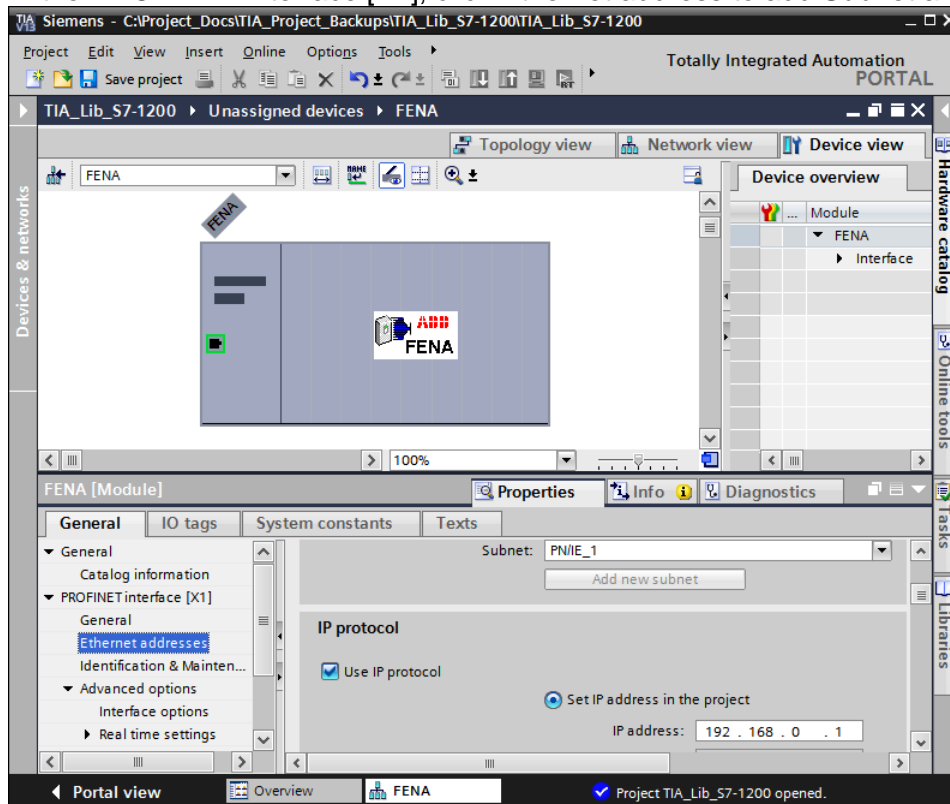

Figure 121 Ethernet address

5. In the Hardware catalog pane, double-click the required PPO type. The selected PPO type is displayed in the Device overview.
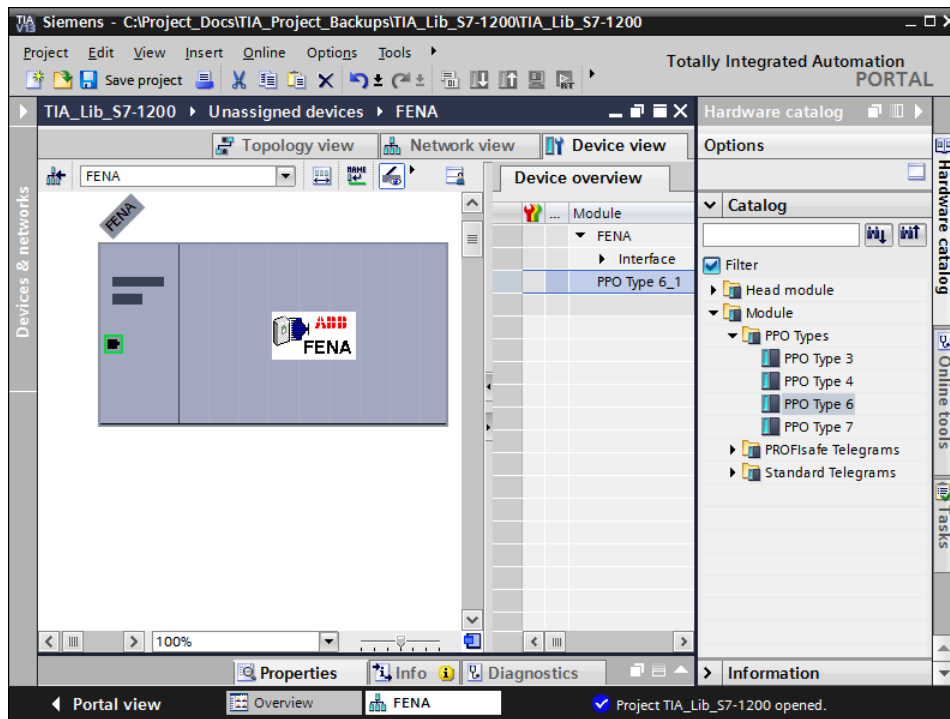

Figure 122 PPO type

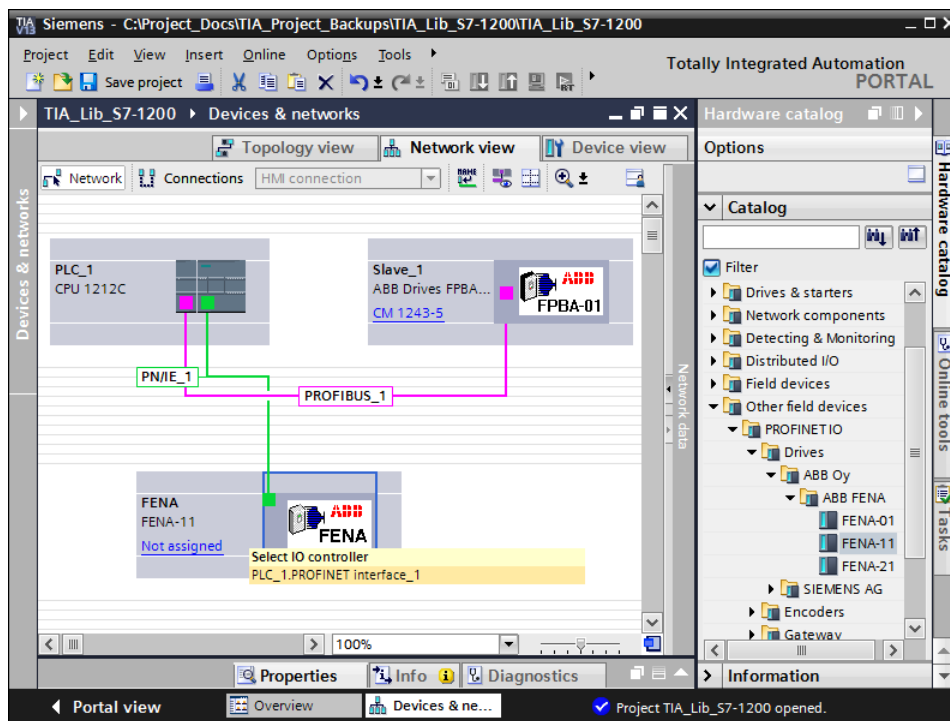6. In the Network view, click **Not assigned** and select PLC_1.PROFINET interface_1 to assign PLC to the ABB drive.

Figure 123 PROFINET interface
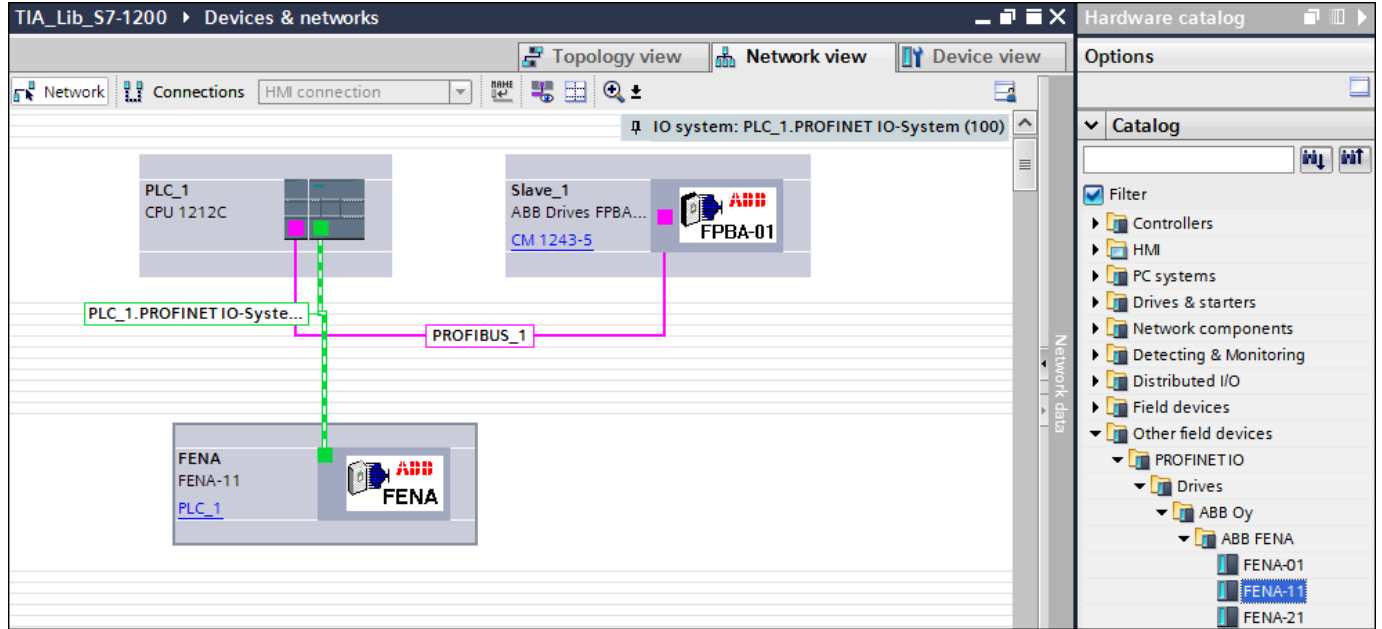
The selected PLC is assigned to the ABB drive.

Figure 124 PROFINET module

7. In the Project tree pane, select PLC_1 and then click **Save** project and click ⊞ to compile.

Downloading configuration to S7-1200 PLC

1. Click ⬇ to download the configuration to PLC.
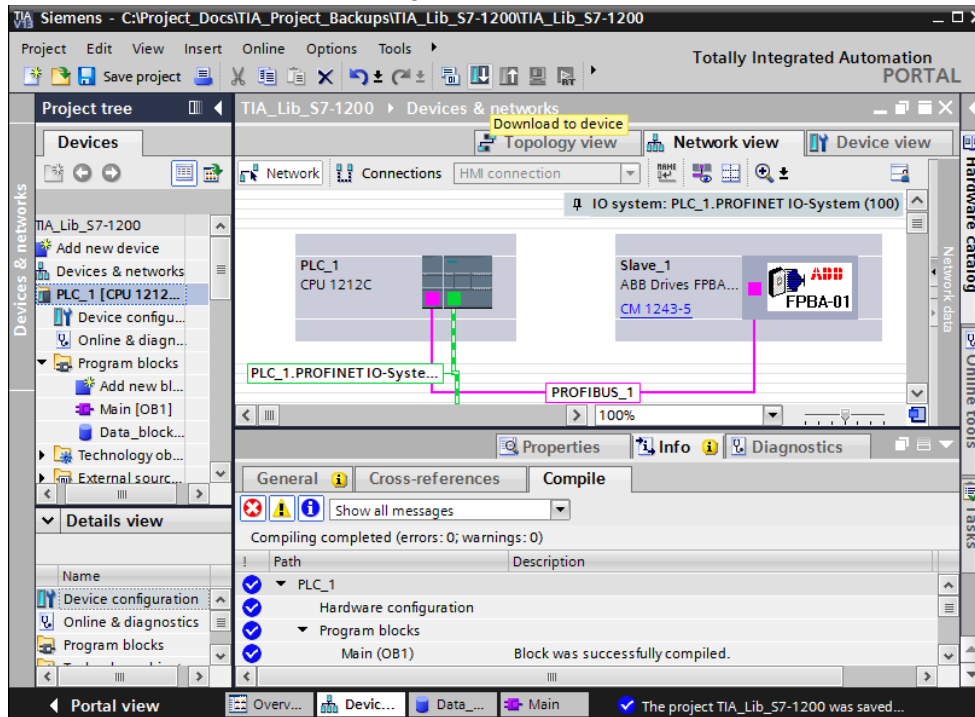

Figure 125 Download

2. From the Extended download to device window, select **PG/PC interface** and **Connection to subnet** from the drop-down list and click **Load**.
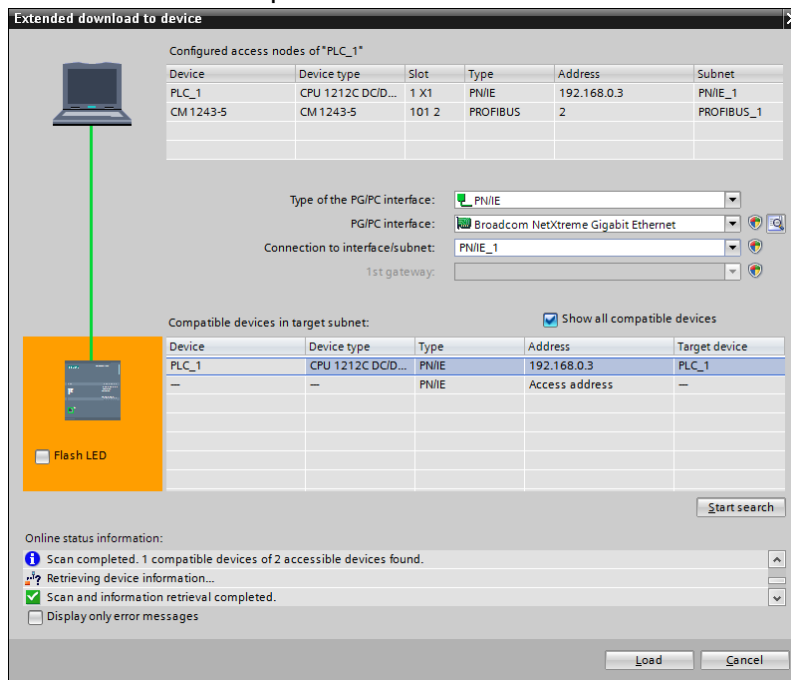
Figure 126 Load

3. Select the required PLC device and click **Load**.
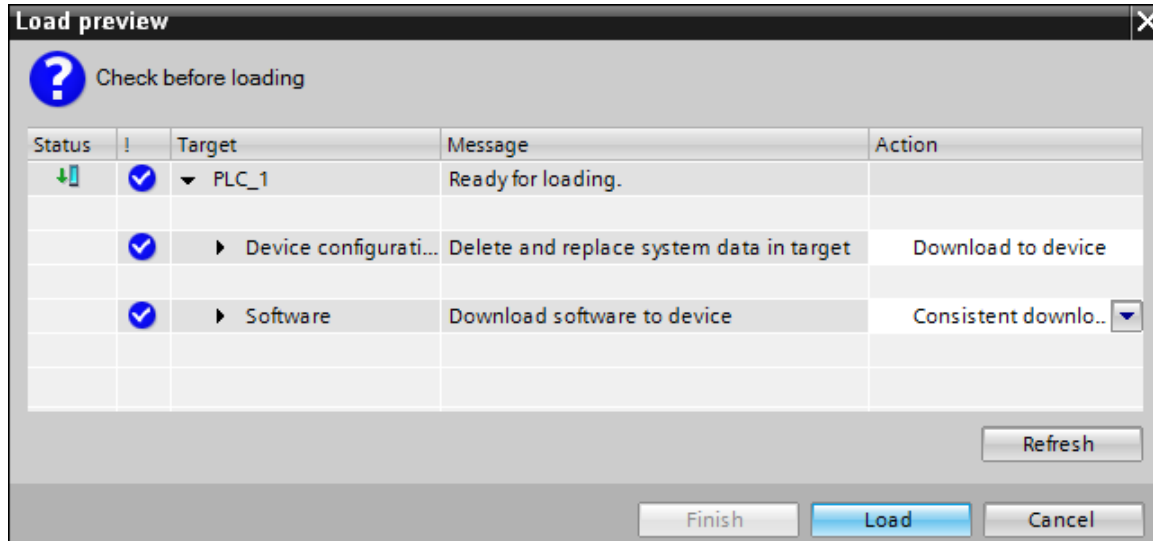
Figure 127 Load preview

4. In the Load results window, select **Start all** and click **Finish**.
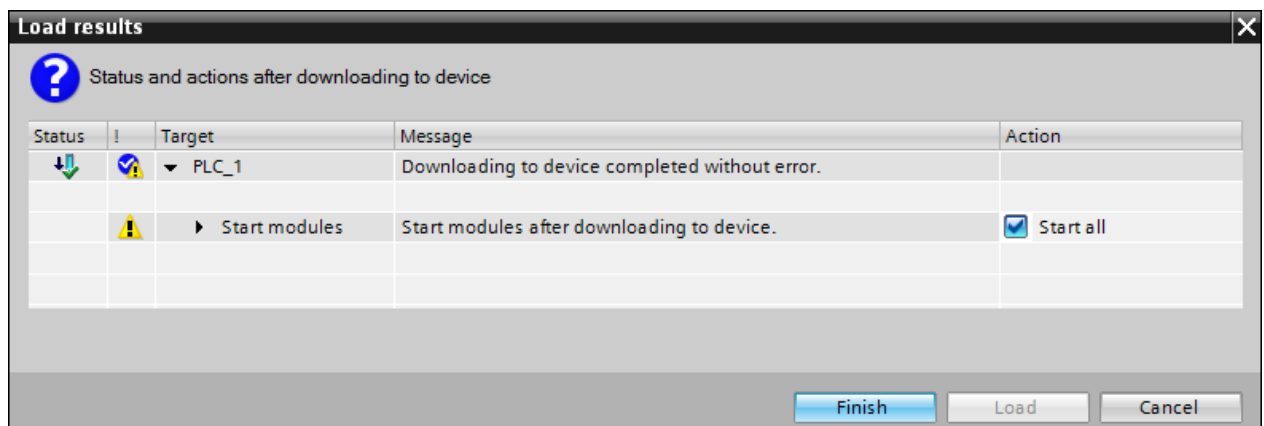
Figure 128 Load results

Copying ABB drive libraries to S7-1200 project

Prerequisites: Check for the ABB Drive libraries available in the user PC.

1. In the main menu of TIA portal, click **Options** and click **Global libraries** and then click **Open library**.
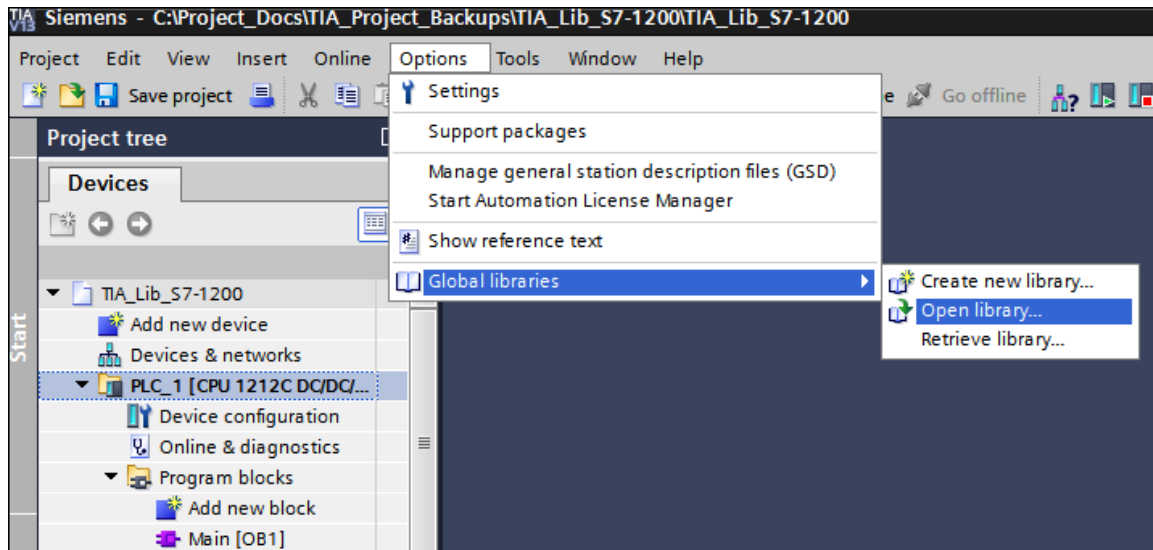
Figure 129 Global libraries

2. Select the required ABB Drive library in the file system and click **Open**.
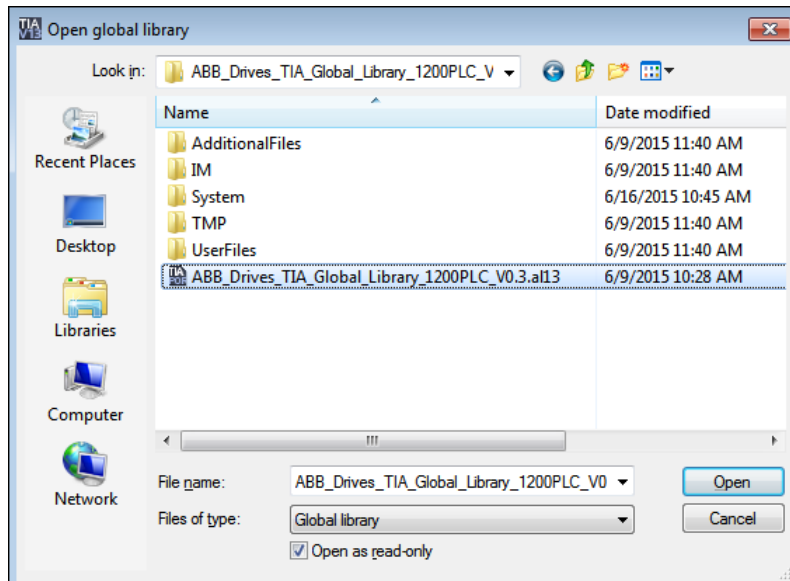
Figure 130 File system

After selecting ABB drive libraries, a Libraries pane is displayed with the selected libraries at the right side of the TIA portal window.
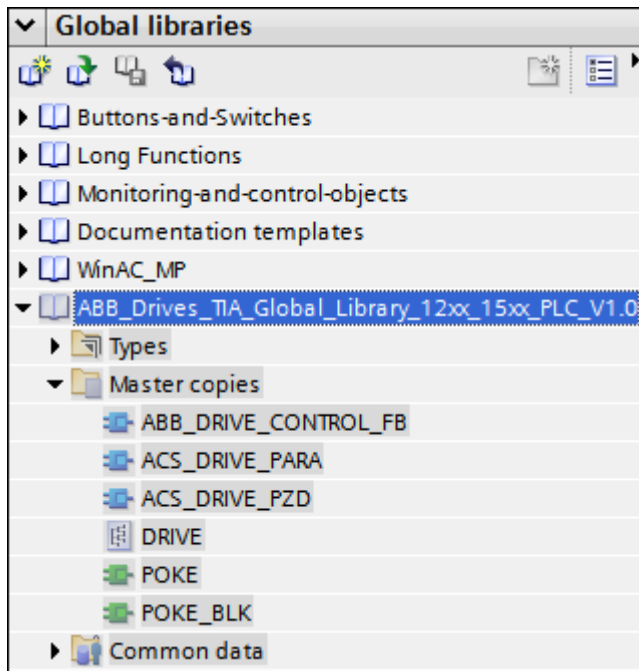
Figure 131 Global libraries pane

Following are the libraries available for S7-1200 series:
- ABB_DRIVE_CONTROL_FB (FB500)
- ACS_DRIVE_PARA (FB501)
- ACS_DRIVE_PZD (FB502)
- POKE (FC1, supported block for FB501 and FB502)
- POKE_BLK (FC2, supported block for FB501 and FB502)

Note! POKE and POKE_BLK are SCL language blocks used in FB501 and FB502. Copy these blocks to Program blocks, if not compilation errors are generated while compiling FB501 and FB502 blocks.

3. Expand **ABB_Drives_TIA_Global_Library** -> **Master copies** and right-click **ABB_DRIVE_CONTROL_FB** and then click **Copy**
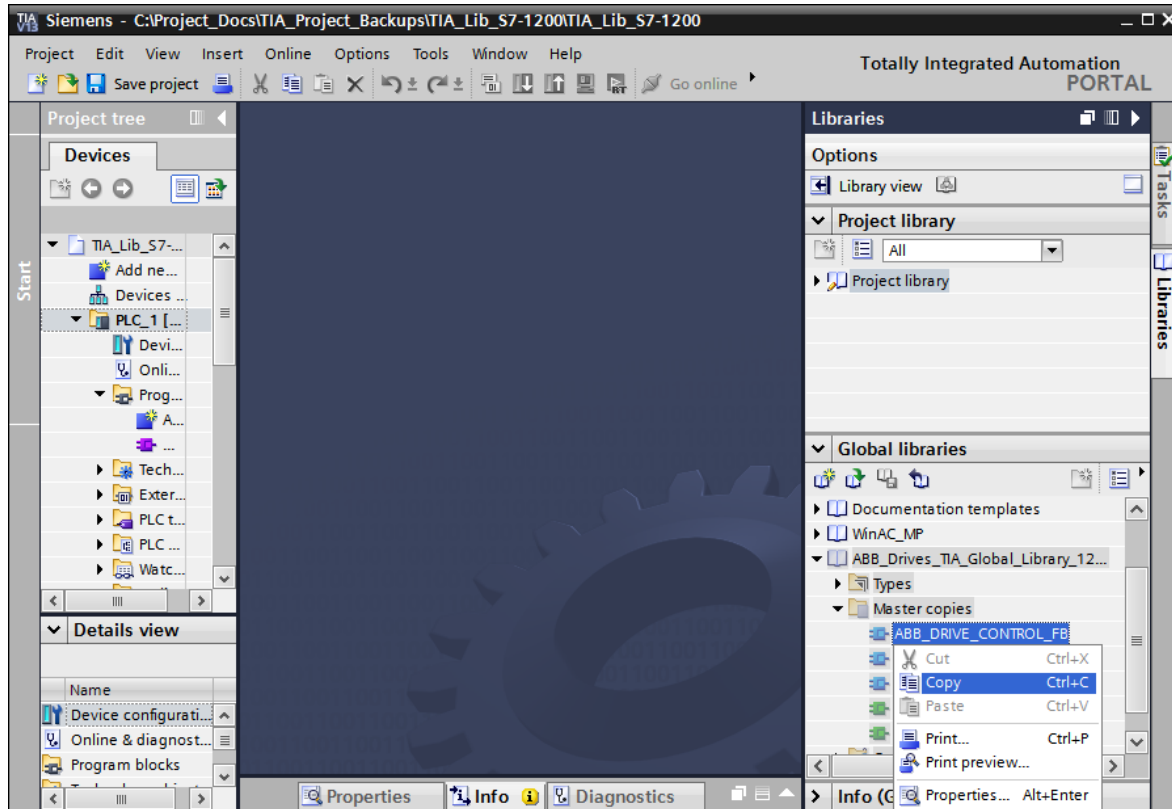
Figure 132 Libraries

4. In the Project tree pane, right-click **Program blocks** and click **Paste**.
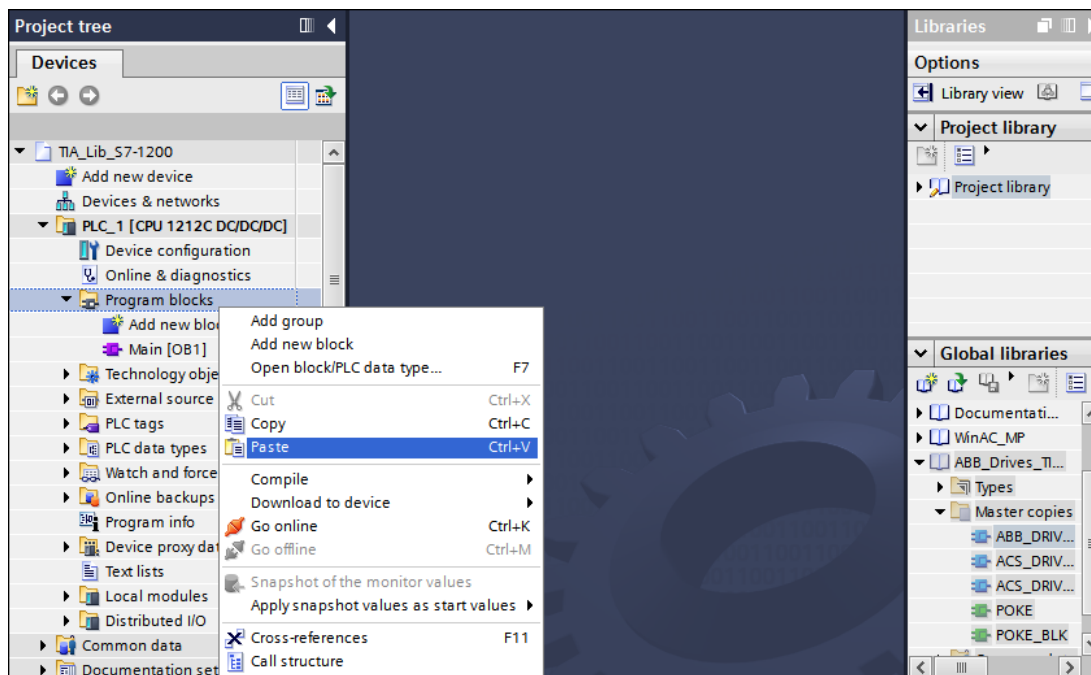
Figure 133 Program blocks

Similarly, copy other ABB drive libraries and paste it in **Program blocks**.
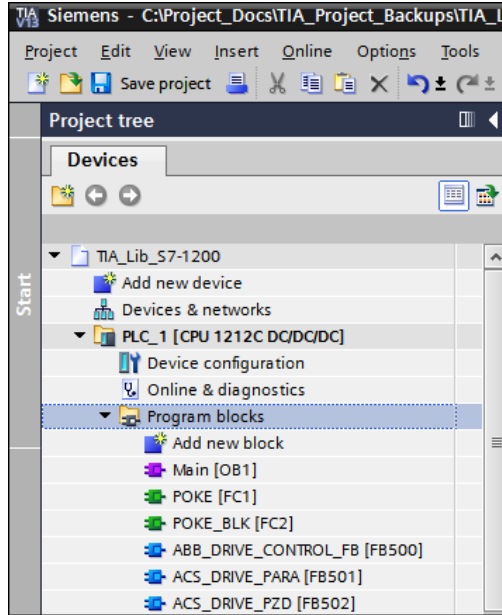
Figure 134 ABB drive libraries

5. In the Libraries pane, expand **ABB_Drives_TIA_Global_Library** -> **Master copies** and right-click **DRIVE** (data types) and click **Copy**.
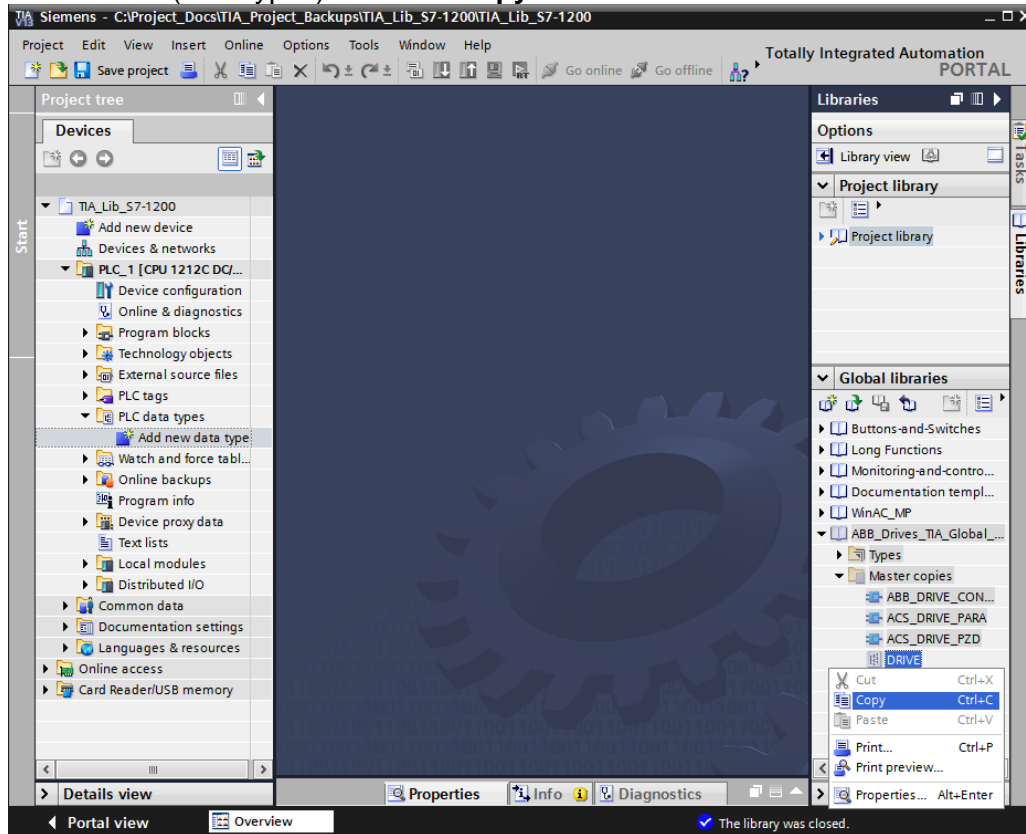
Figure 135 Drive data types

6. In the Project tree pane, right-click PLC data types and click Paste.
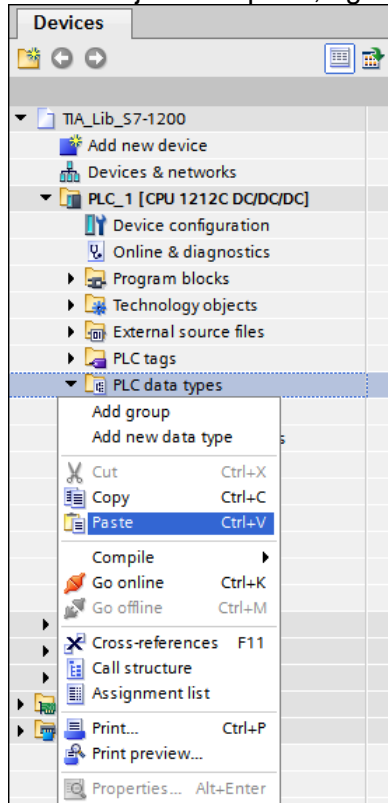
Figure 136 PLC data types

Note: After copying the libraries, right-click and click **Close library** in the global libraries pane.
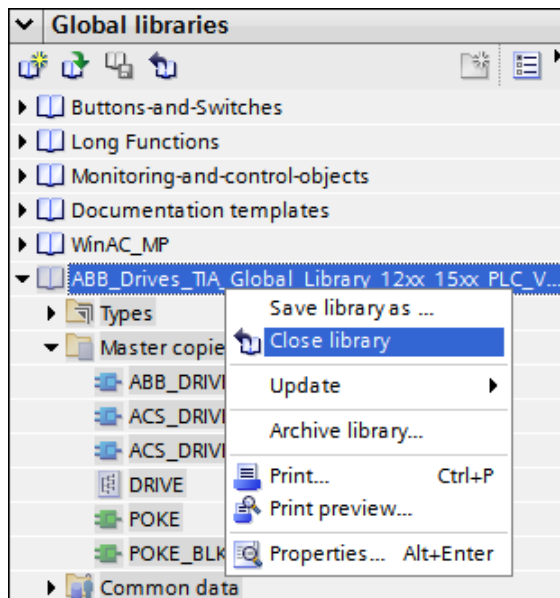
Figure 137 Closing library

Adding PLC tags to S7-1200

1. In the Project tree pane, expand PLC tags and double-click Add new tag table to create symbols.
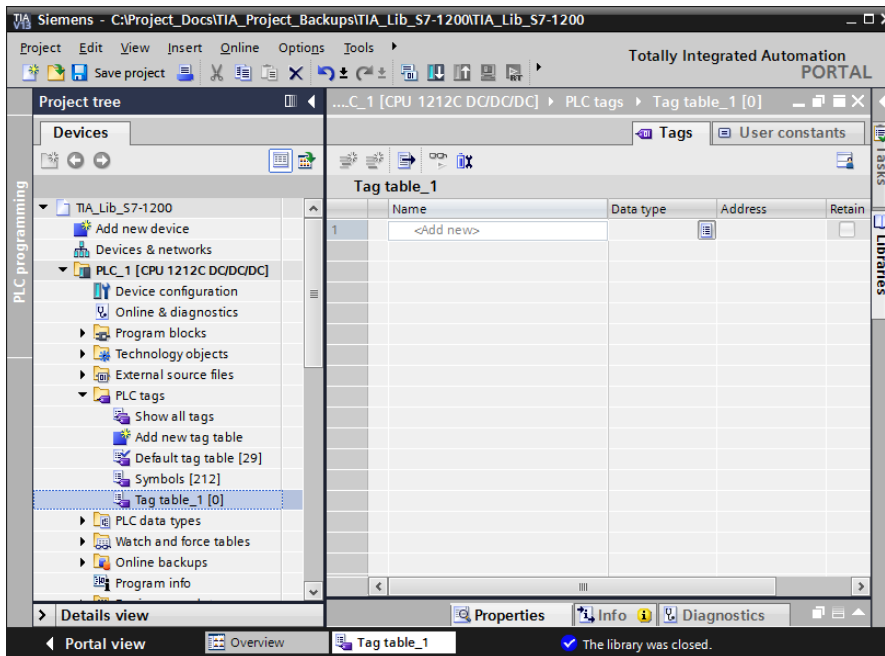
Figure 138 New tags table for S7-1200

2. Add tag Name, Data type and Address to connect block inputs and outputs.
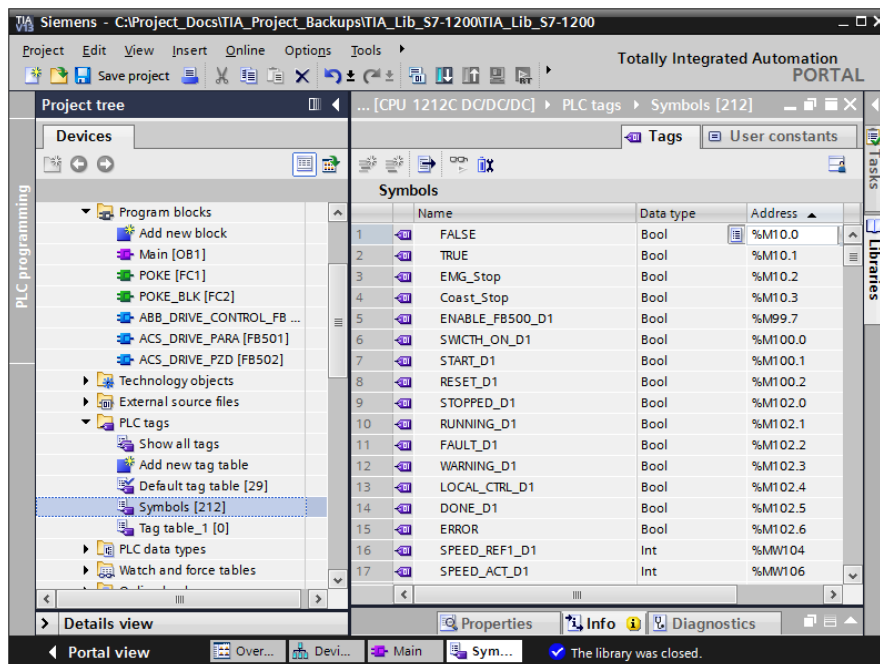
Figure 139 Symbols

For information on Data types of tags, see sections *Block variables and data types for FB500* (page *130*), *Block variables and data types for FB501* (page *135*) and *Block variables and data types for FB502* (page *141*).

FB500 ABB_DRIVE_CONTROL_FB

FB500 is used to control the drive (start, stop, reset, emergency stop, speed reference, etc.).

Note! The following description is an example, there are different ways to use the Control function block.

1. In the Program blocks, double-click **Main [OB1]** and create memory bits for logic True and logic False.
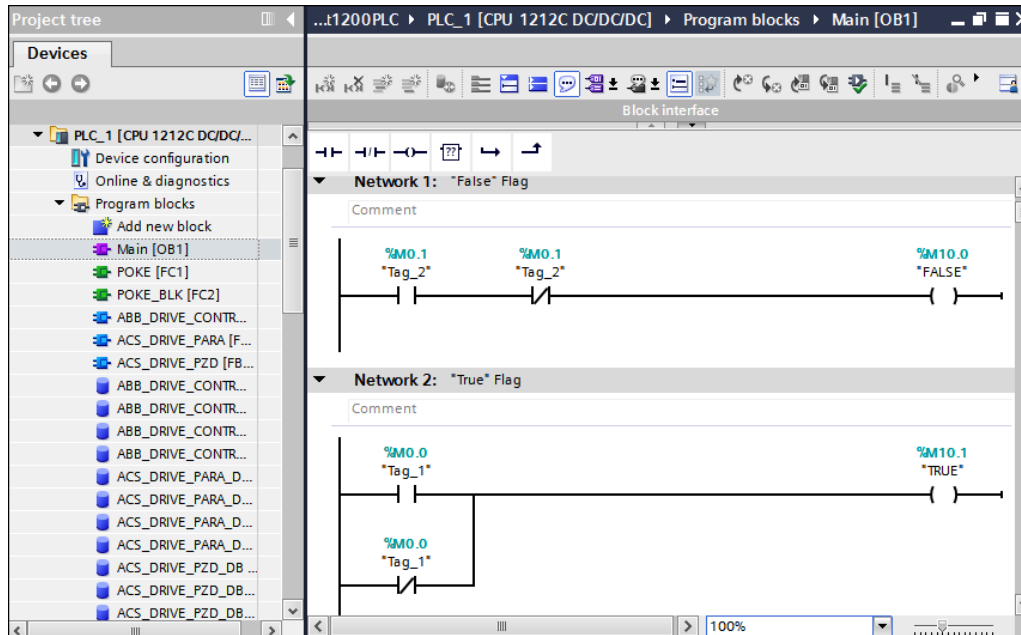
Figure 140 Logic TRUE/FALSE

2. Right-click in the next free network and choose **Insert empty box**.
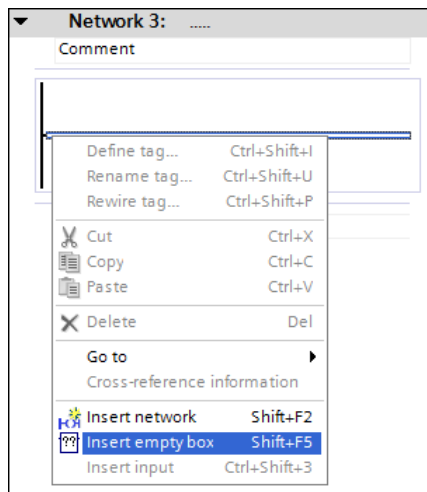
Figure 141 Inserting empty box

3. Write FB500 in the block field to create an instance of FB500.
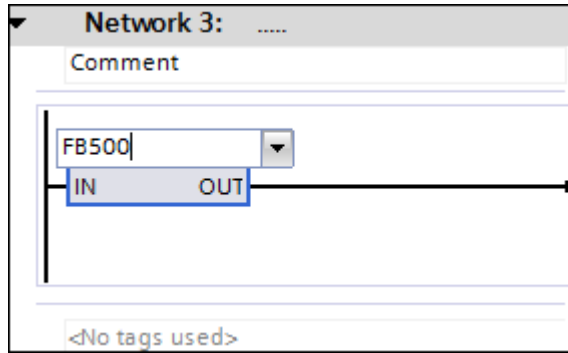
Figure 142 Instance of FB500

4. Create a **unique** Data block for the control of the drive by enabling **Manual** option and by selecting the Number.
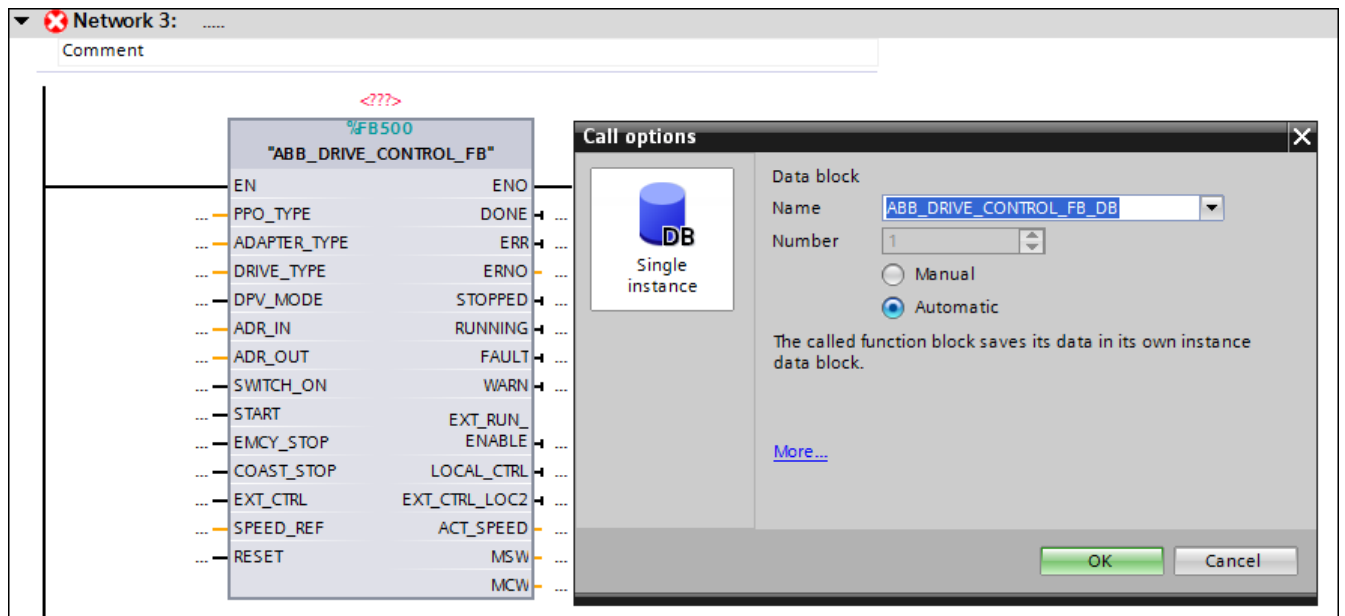
Figure 143 FB500 DB1

Drive status, inputs and outputs of the function block are stored in this Data block. Since it is a new Data block, you have to generate it by clicking **OK** in the pop-up window.

Note! If you add more drives to the program, make sure to create new unique Data blocks.

5. Connect the block inputs and outputs to variables according to your application.

Note! Select a block input or output and see more information in the Info.

Block variables and data types for FB500

Block variable	Data type	Comment
EN	BOOL	Enabling block. FALSE = block code is not executed. TRUE or unconnected = block code is executed.
PPO_TYPE	INT	The PPO type. 1, 2, 3, 4, 5 or 6; 0 = not allowed.
ADAPTER_TYPE	INT	PROFIBUS module type: FPBA-01 PROFIBUS DP module connected in the drive. 1=FPBA (or FENA), 2=RPBA (or RETA).
DRIVE_TYPE	INT	Drive type: ACS800=1, ACSM1=2, ACS350=3, ACS355=4, ACS550=5, ACS850=6, ACS880=7, ACS580=8, ACS380=9.
DPV_MODE	BOOL	FALSE=DP-V0[1], TRUE=DP-V1[2] (or PROFINET).
ADR_IN	HW_IO	Hardware ID of the module from which the data is to be read. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**).
ADR_OUT	HW_IO	Hardware ID of the module to which the data is to be written. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**).
SWITCH_ON	BOOL	FALSE=Drive control switched off, TRUE=Drive control switched on. After an EMERGENCY STOP a new rising edge of SWITCH_ON is needed before next start. SWITCH_ON also needs to be active (TRUE) for resetting drive faults.
START	BOOL	FALSE=Ramp stop with deceleration time according to drive parameter, TRUE=Start. Drive start via fieldbus requires parameter setting in the drive.
EMCY_STOP	BOOL	FALSE=Emergency stop according to emergency stop deceleration time set in drive parameter, TRUE=Normal operation.
COAST_STOP	BOOL	FALSE=Normal operation, TRUE=Coast stop (drive releases control of the motor).
EXT_CTRL	BOOL	Selection of external control location EXT2. FALSE=EXT1, TRUE=EXT2. Shifting to EXT2 via fieldbus requires parameter setting in the drive.

---

[1]To work on library block in DP-V0 mode, user need to configure the drive with DP-V0 GSD file in hardware configuration of TIA portal.

[2] To work on library block in DP-V1 mode, user need to configure the drive with DP-V1 GSD file in hardware configuration of TIA portal.

SPEED_REF	INT	Speed reference value: -20000 to 20000. See chapter "Drive configuration" for scaling. Setting speed reference via fieldbus requires parameter setting in the drive.
RESET	BOOL	FALSE=No operation, TRUE =Reset drive fault.
DONE	BOOL	FALSE=Block execution not finished, TRUE=Block execution finished.
ERR	BOOL	FALSE=No error, TRUE=Error occurred during block execution.
ERNO	INT	Error code when ERR=TRUE, see SIMATIC online help for SFC14 or SFC15.
STOPPED	BOOL	FALSE=Drive is not stopped, TRUE=Drive is stopped.
RUNNING	BOOL	FALSE=Drive is not running, TRUE=Drive is running and following the speed reference value.
FAULT	BOOL	FALSE=No drive fault active, TRUE=Drive fault active.
WARN	BOOL	FALSE=No drive warning active, TRUE=Drive warning active.
EXT_RUN_ENAB LE	BOOL	FALSE=No external run enable signal received in the drive, TRUE=External run enable signal received in the drive.
LOCAL_CTRL	BOOL	FALSE=Remote control (normal mode), TRUE=Local control (e.g. drive control panel or pc tool in local mode)
EXT_CTRL_LOC 2	BOOL	Actual control place, FALSE=EXT1, TRUE=EXT2.
ACT_SPEED	INT	Drive actual speed: -20000 to 20000. See chapter "Drive configuration" for scaling.
MSW	WORD	Drive main status word. See actual fieldbus adapter manual for detailed description.
MCW	WORD	Drive main control word. See actual fieldbus adapter manual for detailed description.

Example 1: The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB_1* and an ACS355 with FPBA-01 PROFIBUS module in DP-V0 mode has been used.
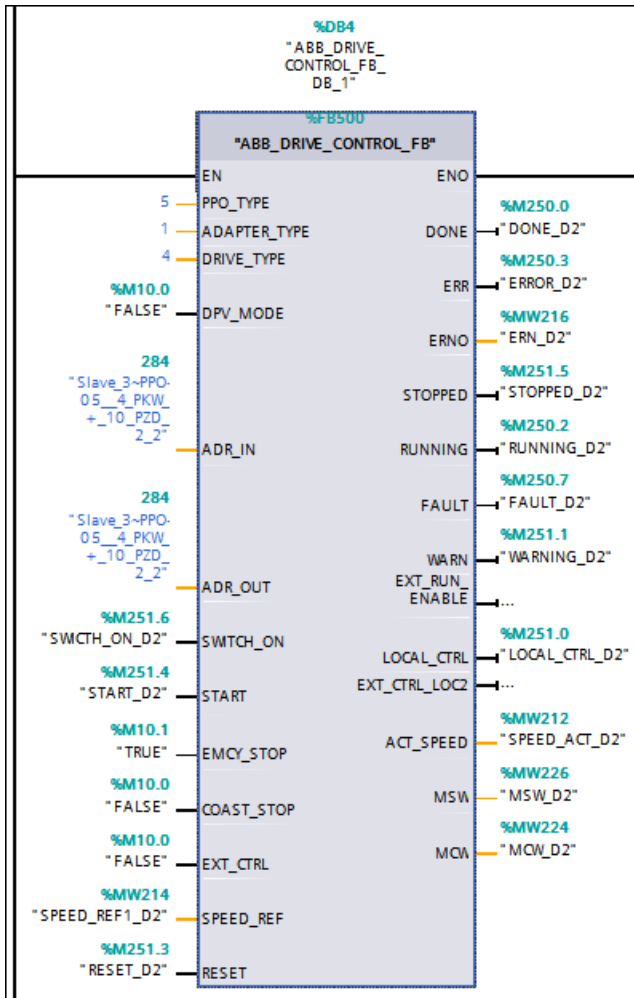
```
                          %DB4
                       "ABB_DRIVE_
                       CONTROL_FB_
                          DB_1"
                          %FB500
                   "ABB_DRIVE_CONTROL_FB"
              ┌─────────────────────────────────┐
              │ EN                          ENO  │
         5 ───┤ PPO_TYPE                          │      %M250.0
         1 ───┤ ADAPTER_TYPE              DONE ├──┤"DONE_D2"
         4 ───┤ DRIVE_TYPE                        │      %M250.3
    %M10.0    │                           ERR ├──┤"ERROR_D2"
   "FALSE" ───┤ DPV_MODE                          │      %MW216
              │                          ERNO ├──┤"ERN_D2"
       284    │                                  │      %M251.5
 "Slave_3~PPO-│                       STOPPED ├──┤"STOPPED_D2"
  05__4_PKW   │                                  │      %M250.2
  +_10_PZD_   │                                  │
    2_2"      │                       RUNNING ├──┤"RUNNING_D2"
          ────┤ ADR_IN                            │      %M250.7
       284    │                         FAULT ├──┤"FAULT_D2"
 "Slave_3~PPO-│                                  │      %M251.1
  05__4_PKW   │                                  │
  +_10_PZD_   │                          WARN ├──┤"WARNING_D2"
    2_2"      │                      EXT_RUN_     │
          ────┤ ADR_OUT               ENABLE ├──┤...
    %M251.6   │                                  │      %M251.0
"SWICTH_ON_D2"┤ SWITCH_ON           LOCAL_CTRL ├──┤"LOCAL_CTRL_D2"
    %M251.4   │                    EXT_CTRL_LOC2 ├──┤...
  "START_D2" ─┤ START                             │
    %M10.1    │                                  │      %MW212
   "TRUE" ────┤ EMCY_STOP            ACT_SPEED ├──┤"SPEED_ACT_D2"
    %M10.0    │                                  │      %MW226
   "FALSE" ───┤ COAST_STOP               MSW ├──┤"MSW_D2"
    %M10.0    │                                  │      %MW224
   "FALSE" ───┤ EXT_CTRL                 MCW ├──┤"MCW_D2"
    %MW214    │                                  │
"SPEED_REF1_D2"┤ SPEED_REF                        │
    %M251.3   │                                  │
  "RESET_D2" ─┤ RESET                             │
              └─────────────────────────────────┘
```

Figure 144 Example FPBA-01

Example 2: The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB* and an ACS880 with FENA-11 PRONET module has been used.
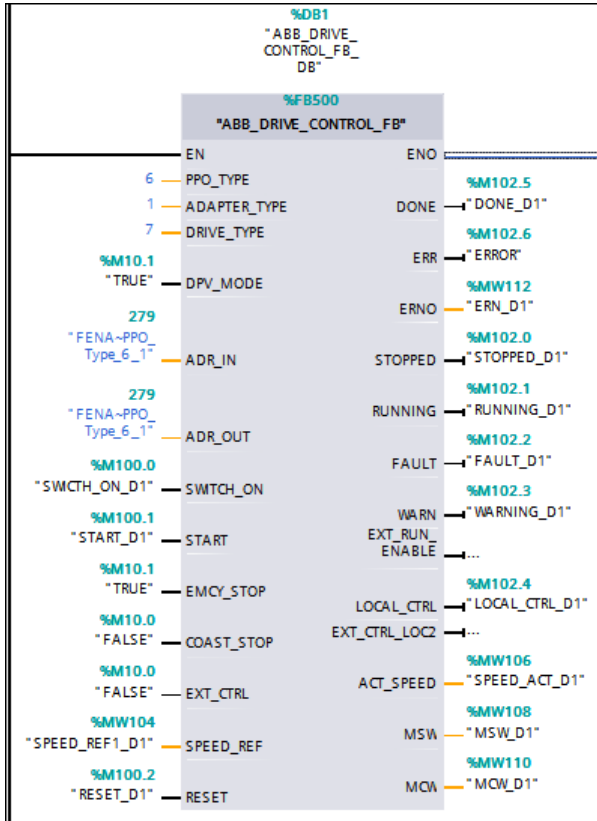

Figure 145 Example FENA-11
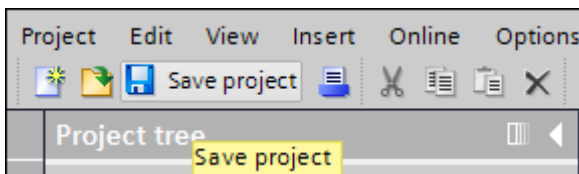
6. Click **Save** to save to the project.

Figure 146 Save project

After saving and compiling, download the program to a PLC.

To set the drive in standby mode and run mode, see sections *Setting the drive in standby mode* (page 55) and *Setting the drive in run mode* (page 55).

To add watch and force tables, see section *Adding watch and force tables* (page 98).

FB501 ACS_DRIVE_PARA

FB501 is used to read/write extra parameters between the PLC and the drive.

1. Right-click in the next free network and choose **Insert empty box**.
2. Write **FB501** in the block field to create an instance of FB501 (*FC500 ACS_DRIVE_PARA* is included in the ABB library ABB_DRIVE_LIB).

Figure 147 FB501

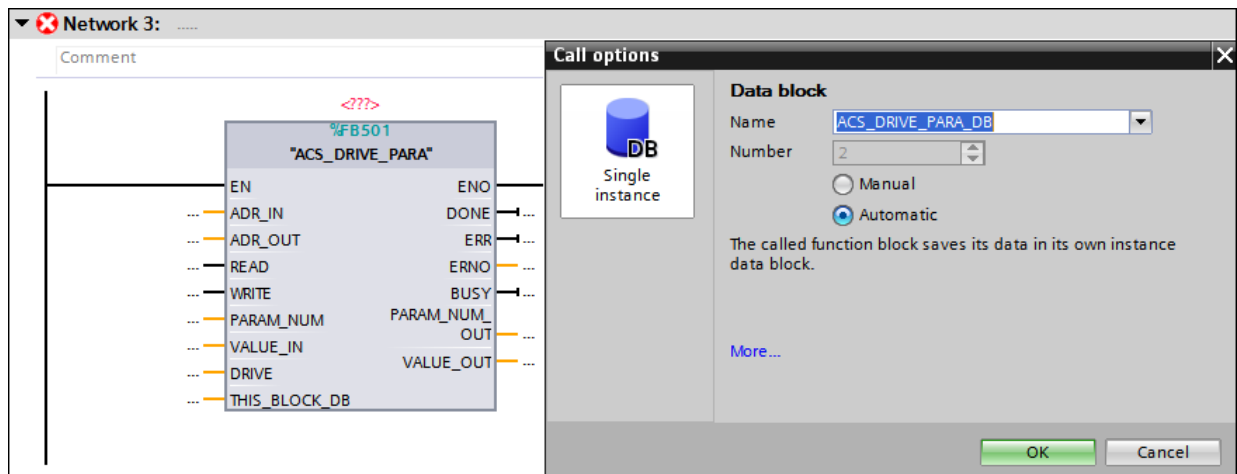3. Create a **unique** Data block by enabling **Manual** option and by selecting the Number.

Figure 148 FB501 DB1

Drive parameter number, its value are stored in the Data block. Since it is a new Data block, you have to generate it by clicking **OK** in the following pop-up window.

4. Connect the FB501 block inputs and outputs.

Block variables and data types for FB501

Block variable	Data type	Comment
ADR_IN	HW_IO	Hardware ID of the module from which the data is to be read. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**). • DP-V0: Hardware ID of the module must be provided and *DPV_Mode* flag should be *False* in FB500 block. • DP-V1 (or PROFINET): Hardware ID or 0 and *DPV_Mode* flag must be *True* in FB500 block.
ADR_OUT	HW_IO	Hardware ID of the module to which the data is to be written. The hardware ID can be found in the properties of the module in the device view or system constants (**PLC tags -> Default tag table -> System constants**). • DP-V0: Hardware ID of the module must be provided and *DPV_Mode* flag should be *False* in FB500 block. • DP-V1 (or PROFINET): Hardware ID or 0 and *DPV_Mode* flag must be *True* in FB500 block
READ	BOOL	Read the parameter value 0 -> 1 (executed on positive edge).
WRITE	BOOL	Write the parameter value 0 -> 1 (executed on positive edge).
PARAM_NUM	DINT	Read/written parameter: 3 numbers = group, 2 numbers = Index. For example, Par 20.06 = 2006.
VALUE_IN	DINT	Parameter value to be written.
DRIVE	DB_ANY	Instance Data Block. The drive variable is used for identifying to which drive FB501 *ACS_DRIVE_PARA* belongs. The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* must correspond to the variable FB501 ACS_DRIVE_PARA.
THIS_BLOCK_DB	DB_ANY	Instance data block of FB501. **Note:** POKE (FC1) and POKE_BLK (FC2) are used in FB501 block logic to exchange data between FB500 and FB501.
DONE	BOOL	FALSE=Block execution not finished, TRUE=Block execution finished.
ERR	BOOL	FALSE=No error, TRUE=Error occurred during block execution.
ERNO	WORD	Error code when ERR=TRUE.
BUSY	BOOL	FALSE=No operation active, TRUE=Operation active.
PARAM_NUM_OUT	DINT	Handled parameter number: 3 numbers = group, 2 numbers = index; for example, Par 20.06 = 2006.
VALUE_OUT	DINT	Read parameter value.

To configure ADR_IN and ADR_OUT inputs in DP-V0, user has to provide the hardware ID of the PPO type which is configured.
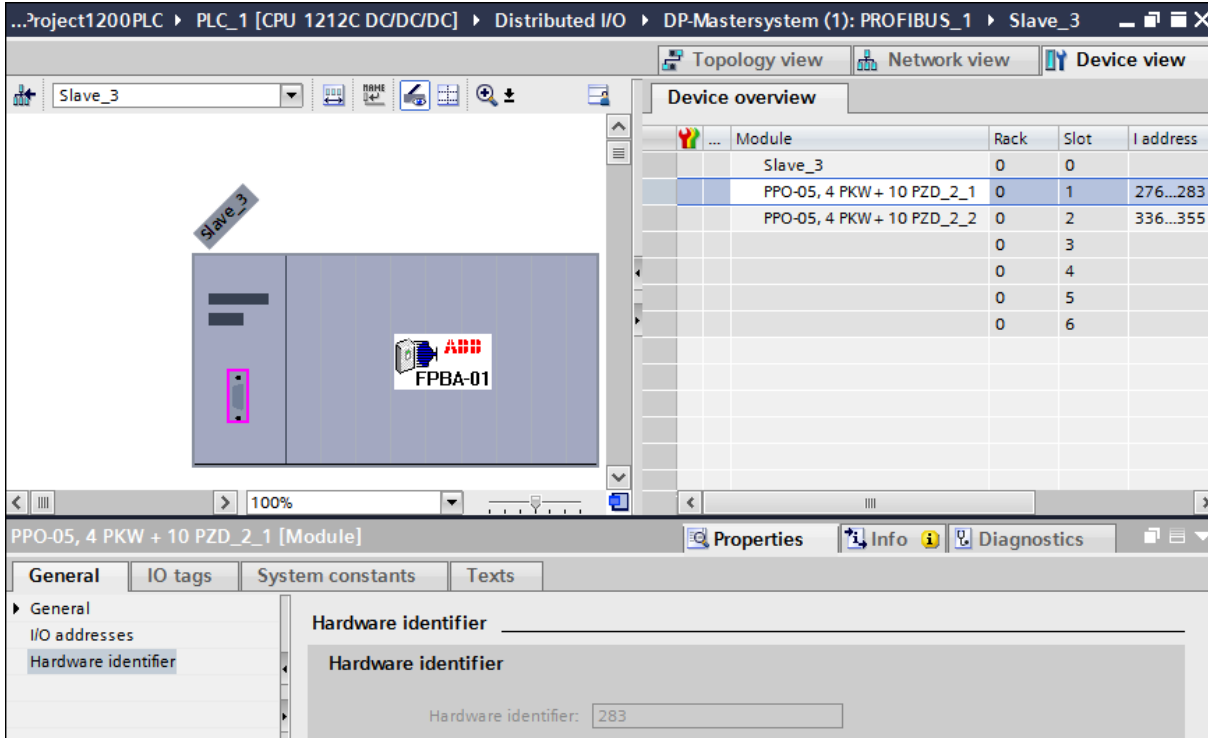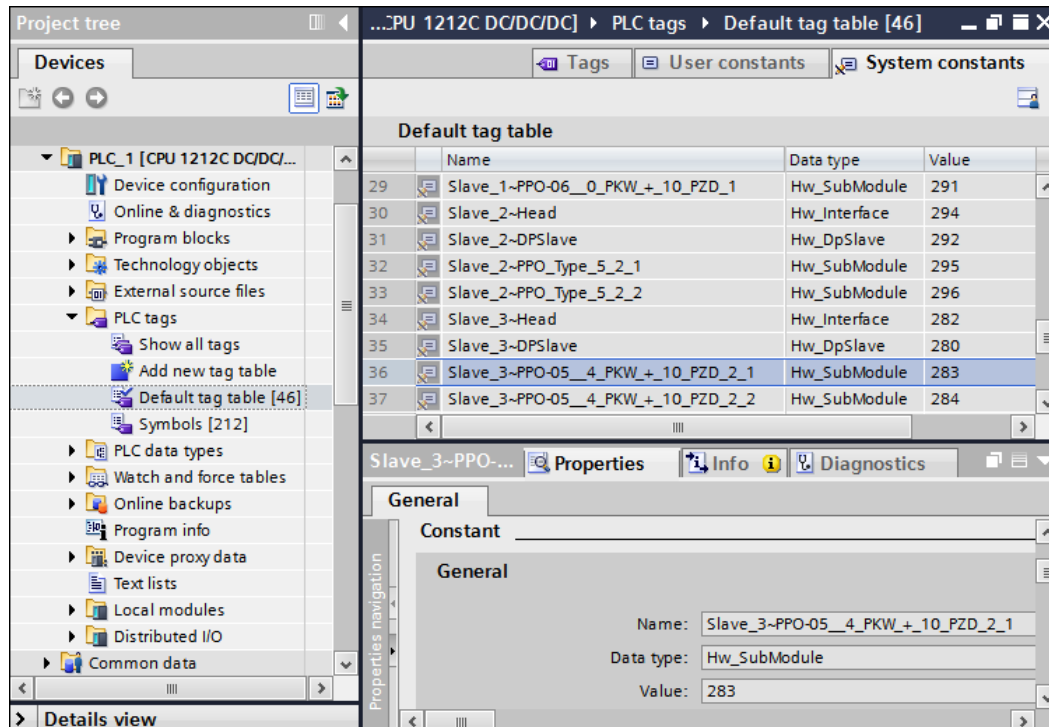

Figure 149 ADR_IN input in DP-V0

Figure 150 System constants

For further information, see the examples.

Example 1: The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB_1* and instance data block of FB501 *ACS_DRIVE_PARA* has been named *ACS_DRIVE_PARA_DB_1* and ACS355 with FPBA-01 PROFIBUS module in DP-V0 mode has been used.
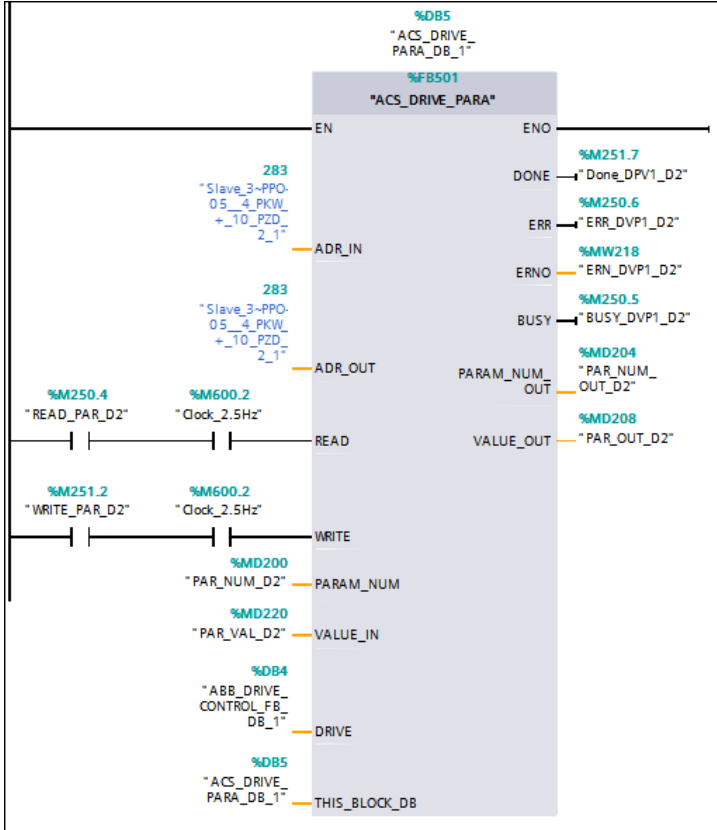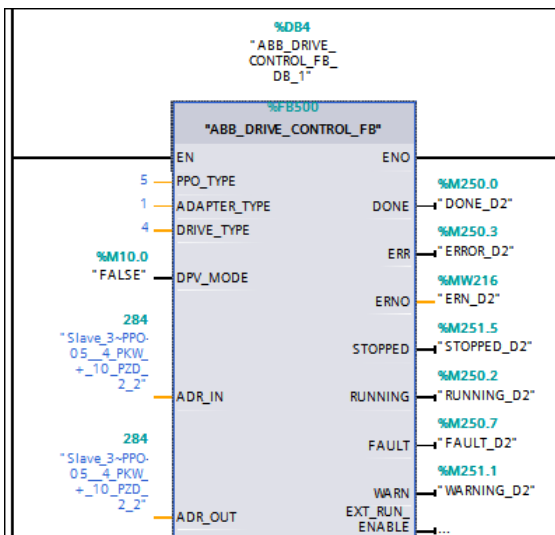

Figure 151 Example FB501 DP-V0

Figure 152 Example FB501 DP-V0

Example 2: The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB* and instance data block of FB501 has been named *ACS_DRIVE_PARA_DB* and ACS355 with FENA-11 PROFINET module has been used.
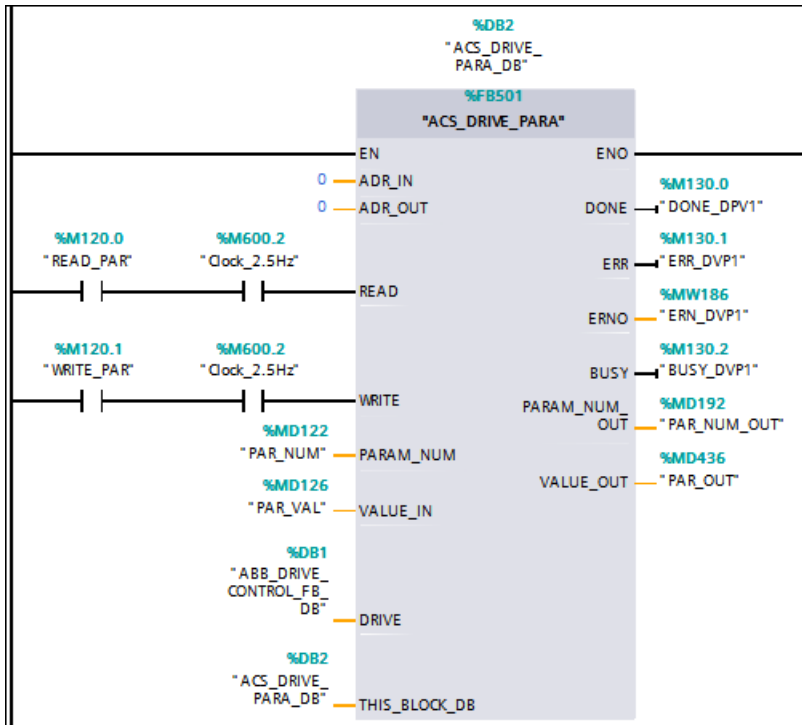

Figure 153 FB501 PROFINET

5. In the Project tree, right-click PLC_1 (CPU) and select **Properties** to execute read/write parameter value (0 -> 1 (executed on positive edge).
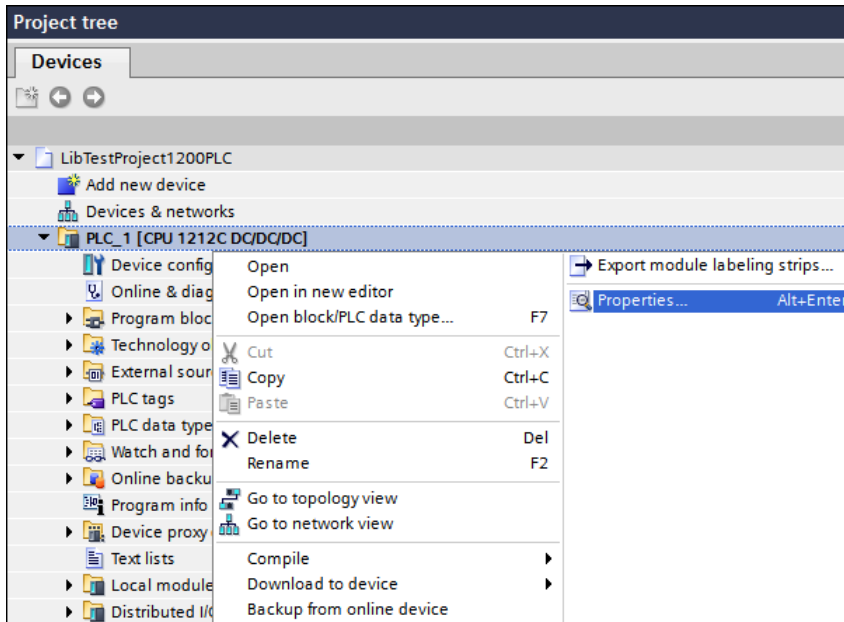

Figure 154 Clock memory properties

6. In the PLC_1 (CPU) window, click **System and clock memory** and enable **Enable the use of clock memory byte**, assign memory address (example, 600) and click **OK**.
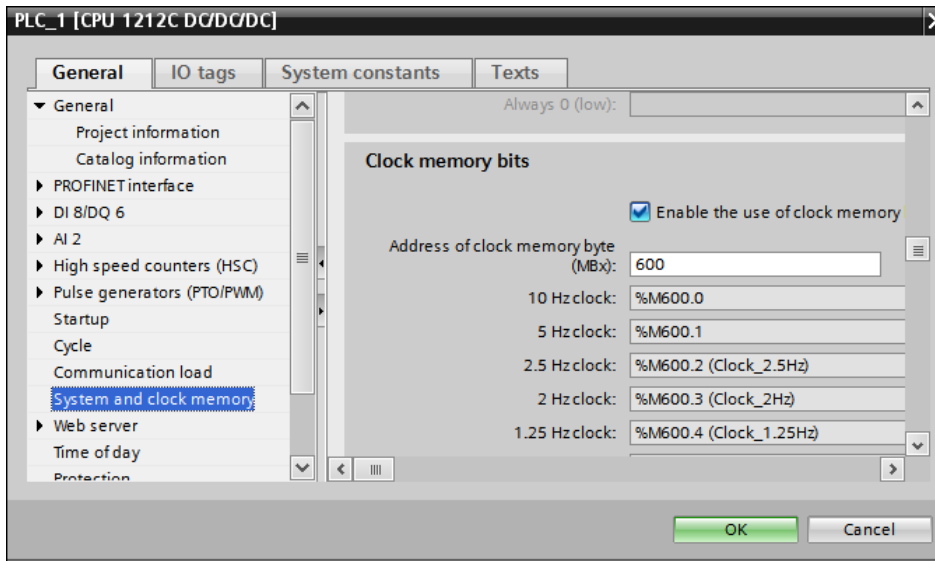

Figure 155 Clock memory

After saving and compiling, download the program to a PLC.

FB502 ACS_DRIVE_PZD

FB502 is used to send additional process data between the PLC and the drive.

Depending on actual PPO type, a certain number of data words (PZDs) are exchanged. PPO types 1 and 3 have only 2 PZDs in each direction, so FB502 (*ACS_DRIVE_PZD*) is not useful for those types. PPO types 2 and 4 have 6 PZDs in each direction, so a part of FB502 (PZD3 to PZD6) is useful here. PPO types 5 and 6 have 10 PZDs in each direction, so the full range of FB502 is useful for those types.

1. Right-click in the next free network and choose **Insert empty box**.
2. Write **FB502** in the block field to create an instance of FB502.
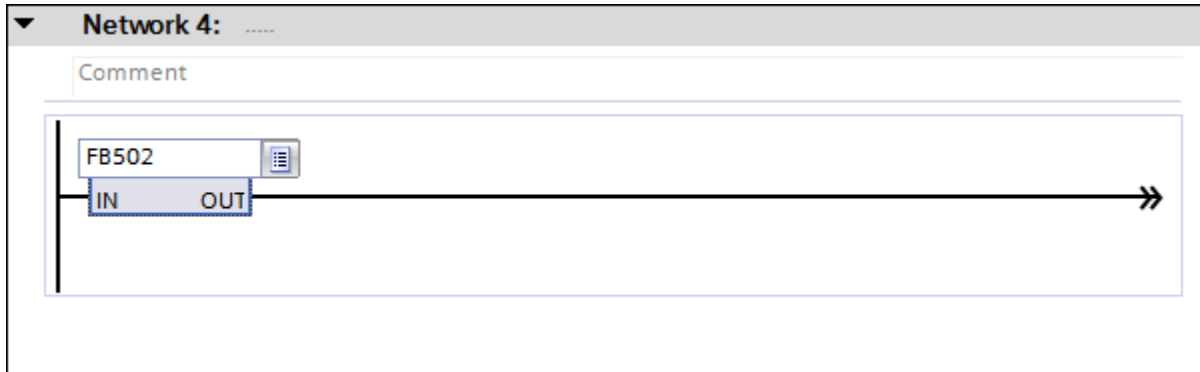

Figure 156 FB502

3. Create a **unique** Data block by enabling **Manual** option and by selecting the Number.
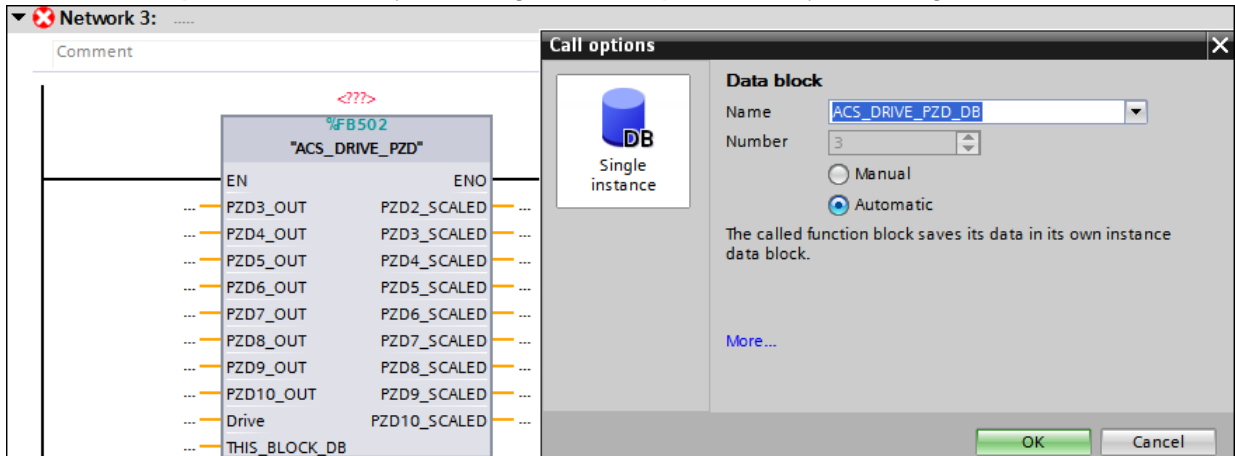

Figure 157 FB502 data block

Drive additional process data values are stored in the Data block. Since it is a new Data block, you have to generate it by clicking **OK** in the following pop-up window.

4. Connect the FB502 block inputs and outputs.

Block variables and data types for FB502

Block variable	Data type	Comment
PZD3_OUT	INT	Write PZD3 value to the drive
PZD4_OUT	INT	Write PZD4 value to the drive
PZD5_OUT	INT	Write PZD5 value to the drive
PZD6_OUT	INT	Write PZD6 value to the drive
PZD7_OUT	INT	Write PZD7 value to the drive
PZD8_OUT	INT	Write PZD8 value to the drive
PZD9_OUT	INT	Write PZD9 value to the drive
PZD10_OUT	INT	Write PZD10 value to the drive
PZD2_SCALED	REAL	Read PZD2 (actual speed / ACT) value from the drive, the default scaling value corresponds to -20 000 to 20 000 -> -100 to 100
PZD3_SCALED	REAL	Read PZD3 value from the drive, the default scaling value is 1 = no scaling
PZD4_SCALED	REAL	Read PZD4 value from the drive, the default scaling value is 1 = no Scaling
PZD5_SCALED	REAL	Read PZD5 value from the drive, the default scaling value is 1 = no Scaling
PZD6_SCALED	REAL	Read PZD6 value from the drive, the default scaling value is 1 = no Scaling
PZD7_SCALED	REAL	Read PZD7 value from the drive, the default scaling value is 1 = no Scaling
PZD8_SCALED	REAL	Read PZD8 value from the drive, the default scaling value is 1 = no Scaling
PZD9_SCALED	REAL	Read PZD9 value from the drive, the default scaling value is 1 = no Scaling
PZD10_SCALED	REAL	Read PZD10 value from the drive, the default scaling value is 1 = no scaling
Drive	DB_ANY	Instance Data Block. The drive variable is used for identifying to which drive FB502 *ACS_DRIVE_PZD* belongs. The Instance Data Block of FB500 *ABB_DRIVE_CONTROL_FB* must correspond to the variable FB502 *ACS_DRIVE_PZD*.
THIS_BLOCK_DB	DB_ANY	Instance data block of FB502. **Note:** POKE (FC1) and POKE_BLK (FC2) are used in FB502 block logic to exchange data between FB500 and FB502.

In the example below, the Instance Data block of FB500 *ABB_DRIVE_CONTROL_FB* has been named *ABB_DRIVE_CONTROL_FB_DB_1* and instance Data block of FB502 has been named *ACS_DRIVE_PZD_DB_1* and ACS355 with FPBA-01 PROFIBUS module is been used.
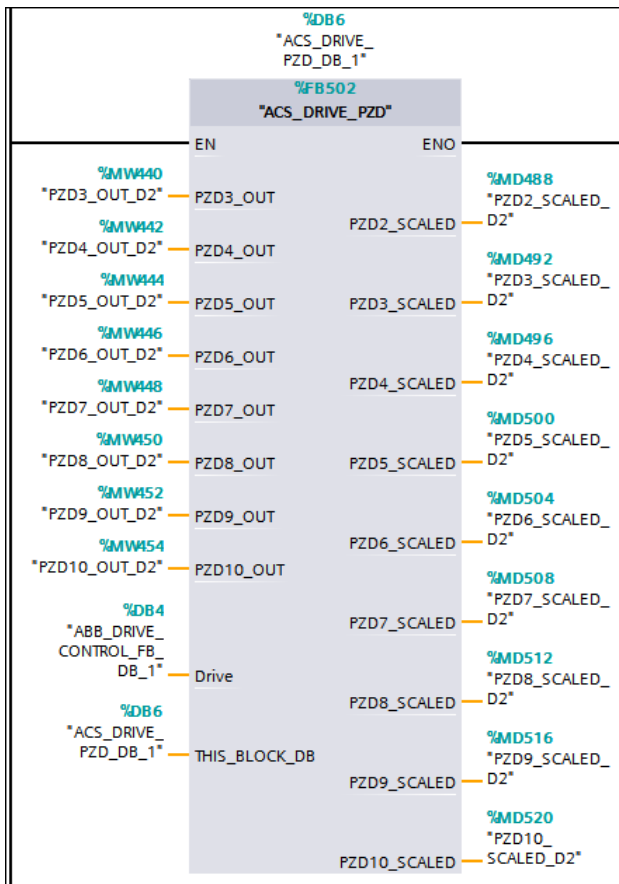

Figure 158 Example FB502

If you want to scale Process Data values, set the scaling values in the program before calling FB502 *ACS_DRIVE_PZD* block. See example below where PZD3 is scaled with the value 100.
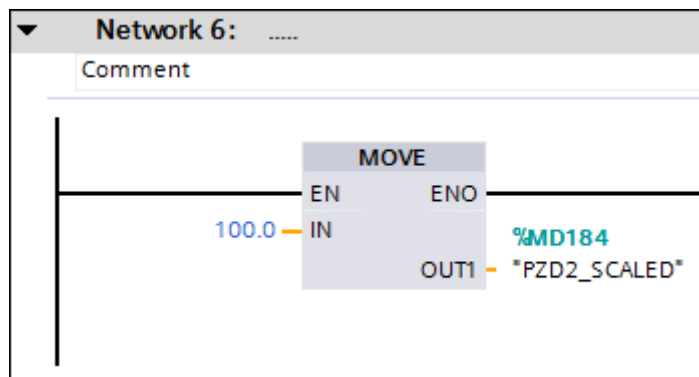

Figure 159 Scaling process data values

For more information on mapping fieldbus process data parameters of the drive according to the application, see section *Scalable variables and data types* (page *61*).

After saving and compiling, download the program to a PLC.

Further information

Product and service inquiries

Address any inquiries about the product to your local ABB representative, quoting the type designation and serial number of the unit in question. A listing of ABB sales, support and service contacts can be found by navigating to *www.abb.com/searchchannels*.

Product training

For information on ABB product training, navigate to *new.abb.com/service/training*.

Providing feedback on ABB Drives manuals

Your comments on our manuals are welcome. Navigate to *www.abb.com/drives/manuals-feedback-form*.

Document library on the Internet

You can find manuals and other product documents in PDF format on the Internet at *www.abb.com/drives/documents*.

Contact us