WITTENSTEIN | alpha

cynapse®

SIEMENS PLC – SIEMENS IO-Link Master

Getting started

2022-D076956

Revision: 03

WITTENSTEIN alpha GmbH
Walter-Wittenstein-Straße 1
D-97999 Igersheim
Germany

Cybertronic support

If you have questions about this implementation example, please contact:
cybertronic-support@wittenstein.de

Customer service

		✉	✆
Deutschland	WITTENSTEIN alpha GmbH	service@wittenstein-alpha.de	+49 7931 493-12900
Benelux	WITTENSTEIN BVBA	service@wittenstein.biz	+32 9 326 73 80
Brasil	WITTENSTEIN do Brasil	vendas@wittenstein.com.br	+55 15 3411 6454
中国	威腾斯坦（杭州）实业有限公司	service@wittenstein.cn	+86 571 8869 5856
Österreich	WITTENSTEIN GmbH	office@wittenstein.at	+43 2256 65632-0
Danmark	WITTENSTEIN AB	info@wittenstein.dk	+45 4027 4151
France	WITTENSTEIN sarl	info@wittenstein.fr	+33 134 17 90 95
Great Britain	WITTENSTEIN Ltd.	sales.uk@wittenstein.co.uk	+44 1782 286 427
Italia	WITTENSTEIN S.P.A.	info@wittenstein.it	+39 02 241357-1
日本	ヴィッテンシュタイン株式会社	sales@wittenstein.jp	+81-3-6680-2835
North America	WITTENSTEIN holding Corp.	technicalsupport@wittenstein-us.com	+1 630-540-5300
España	WITTENSTEIN S.L.U.	info@wittenstein.es	+34 93 479 1305
Sverige	WITTENSTEIN AB	info@wittenstein.se	+46 40-26 50 10
Schweiz	WITTENSTEIN AG Schweiz	sales@wittenstein.ch	+41 81 300 10 30
台湾	威騰斯坦有限公司	info@wittenstein.tw	+886 3 287 0191
Türkiye	WITTENSTEIN Güç Aktarma Sistemleri Tic. Ltd. Şti.	info@wittenstein.com.tr	+90 216 709 21 23

Table of contents

1 About this manual

This guide contains procedures for the exemplary use of the WITTENSTEIN sensor cynapse®.

This guide uses example code. If you require any code examples, please contact: cybertronic-support@wittenstein.de

The original was prepared in German, all other language versions are translations of the original instructions.

1.1 Information symbols and cross references

The following information symbols are used:

- Indicates an action to be performed

➲ Indicates the results of an action

ⓘ Provides additional handling information

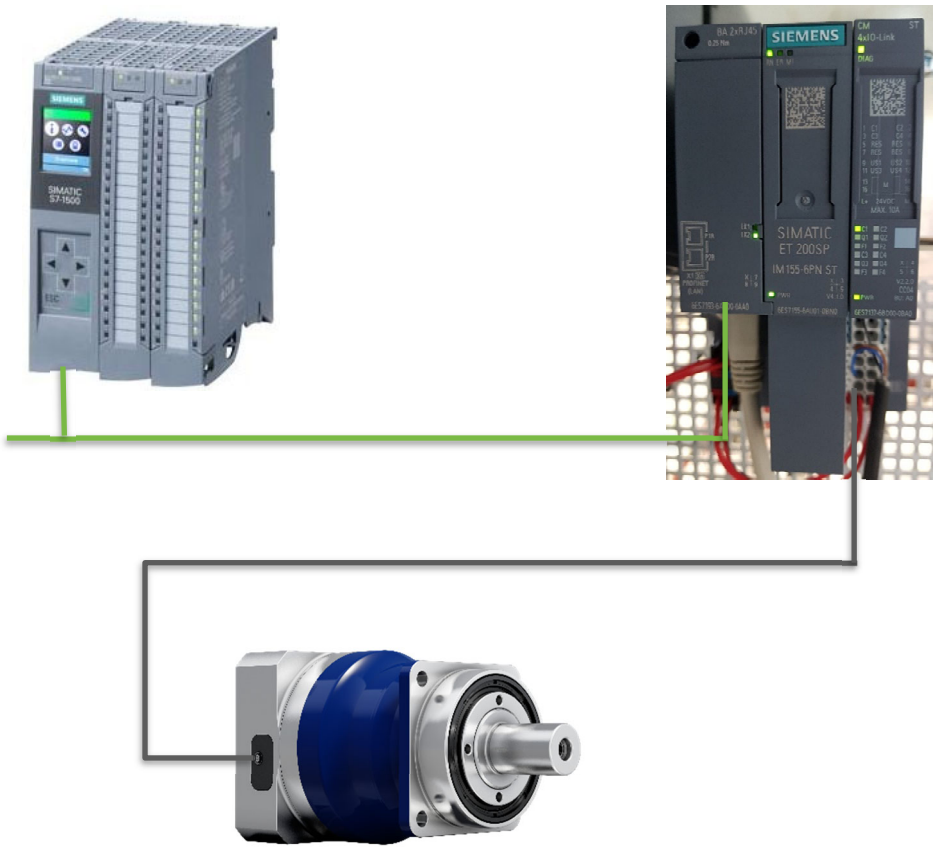
A cross reference refers to the chapter number and the header of the target section (e. g. chapter 5 “Parameter”).

A cross reference to a table refers to the table number (e.g. table “Tbl - 1”).

2 Hardware structure

The hardware structure of the sample project consists of the following components:

- Control system: SIEMENS S7-1500 (6ES7511-1AK02-0AB0)
- Bus adapter: BA 2xRJ45 (6ES7193-6AR00-0AA0)
- PROFINET interface module: SIMATIC ET 200SP IM 155-6PN (6ES7155-6AU01-0BN0)
- IO-Link master: CM 4xIO link (6ES7137-6BD00-0BA0)
- IO link device: WITTENSTEIN cynapse®



The Siemens IO system (with IO-Link Master) is connected to the control unit via the bus adapter via PROFINET (green). cynapse® is connected to one of the IO-Link ports of the master (black).

Knowledge of the correct wiring of all components is assumed and is not covered in this example description.

3 Commissioning in the SIEMENS TIA Portal V15.1

Requirement

To carry out the commissioning of cynapse® you need an open project in the TIA portal.

- ➲ The hardware has been set up.
- ➲ An IP address and the subnet mask have already been assigned for the existing CPU.

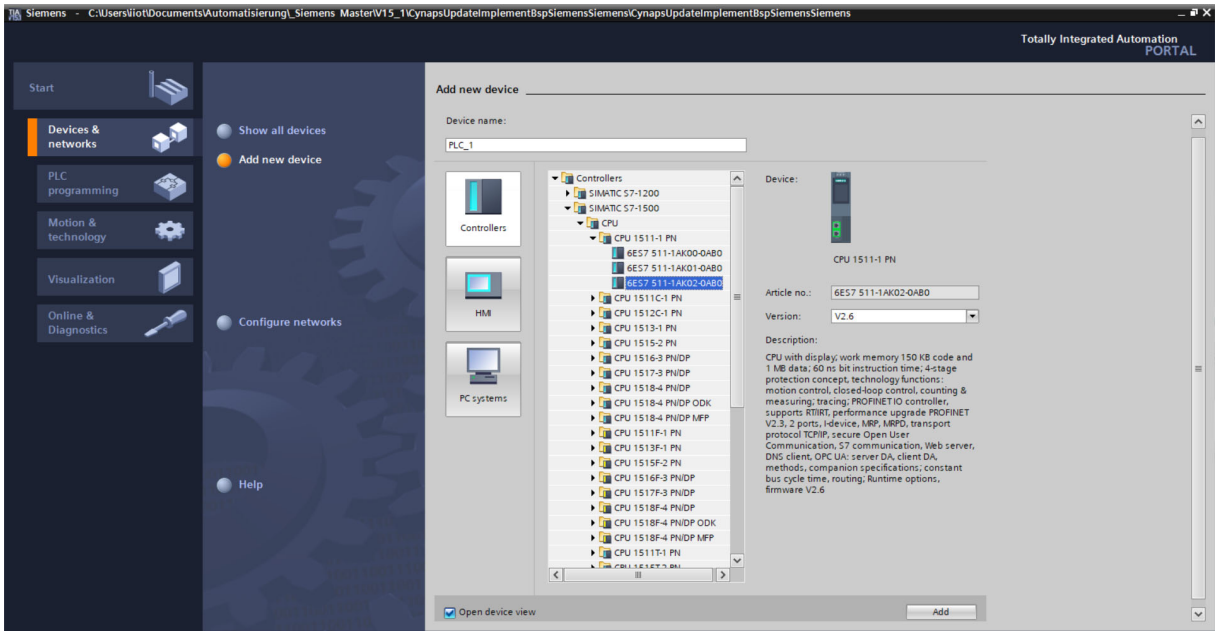
3.1 Hardware configuration Siemens components

Introduction

In the following, you create the CPU, the decentralized peripheral system ET 200SP and the IO-Link Master in the hardware configuration and network them together.

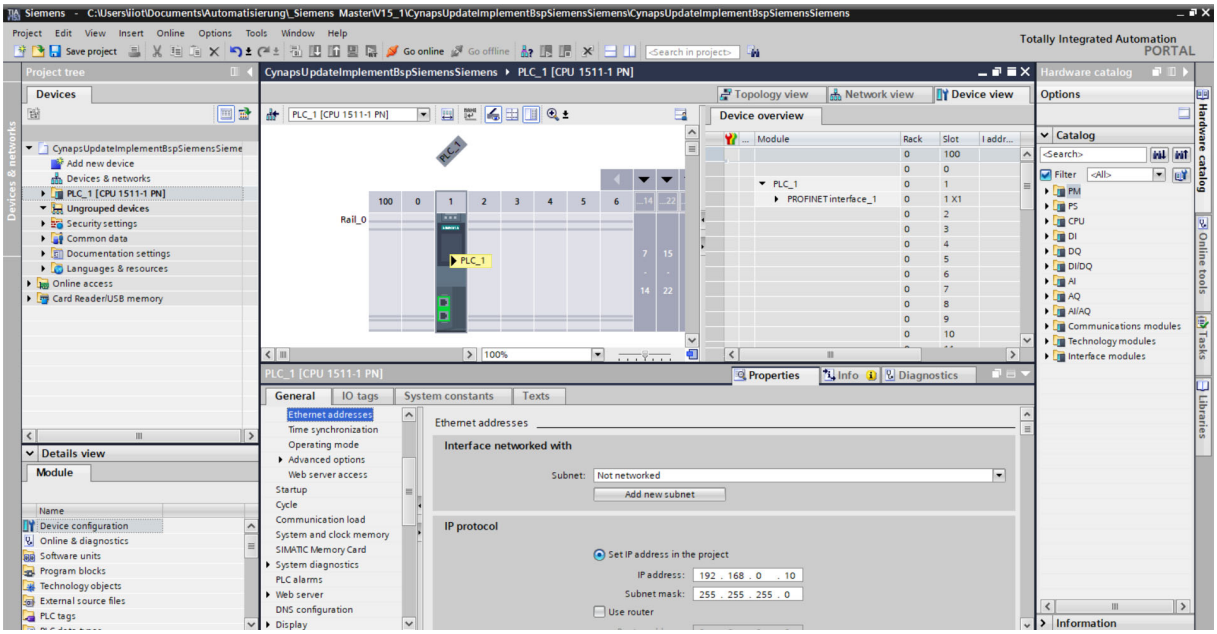
Procedure

1. Open the “Devices & networks” portal.
2. Insert a new device.
3. Open the folder “SIMATIC S7-1500”.
4. Select the CPU you are using.
5. If necessary, adjust the version of your hardware.

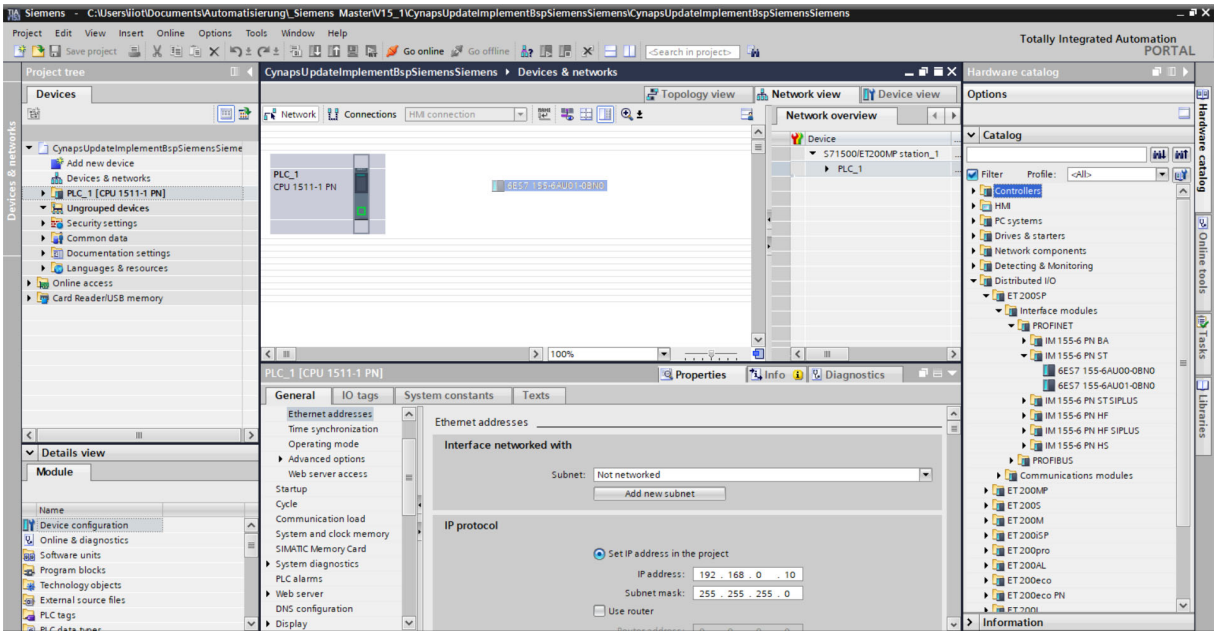


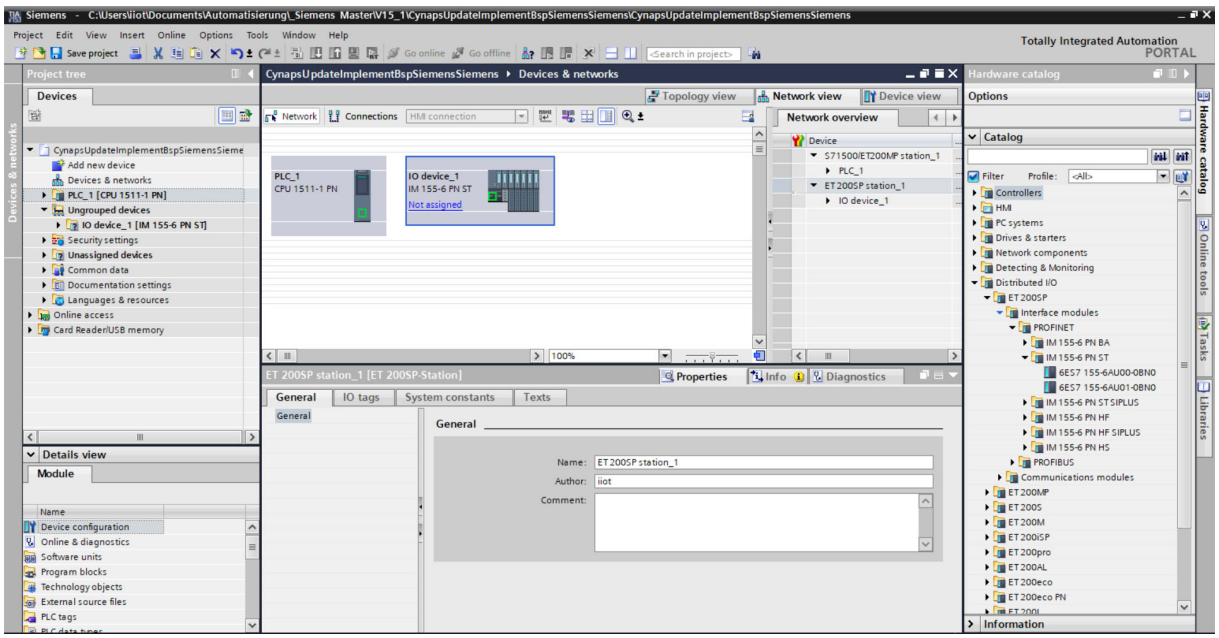
6. Create the CPU by double-clicking on the name.
7. Double-click on the CPU in the automatically opened project view and jump into the CPU settings.

8. Under “Ethernet addresses”, enter the assigned IP address and the subnet mask.

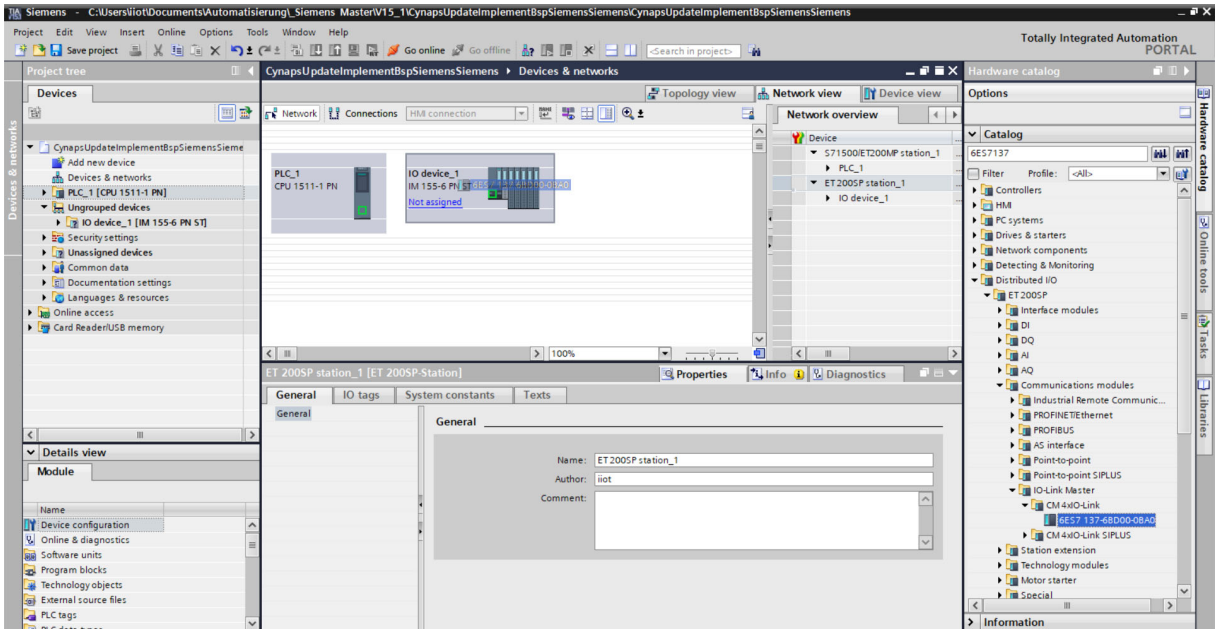


9. Open the “Hardware catalog”.
10. Switch to the “Network view”.
11. Open the “Distributed I/O” folder and the “ET 200SP” folder.
12. Open the “Interface modules”, “PROFINET” folders and the “IM 155-6 PN ST” folder.
13. Drag the interface module you are using and drop it into the white background of the network view.

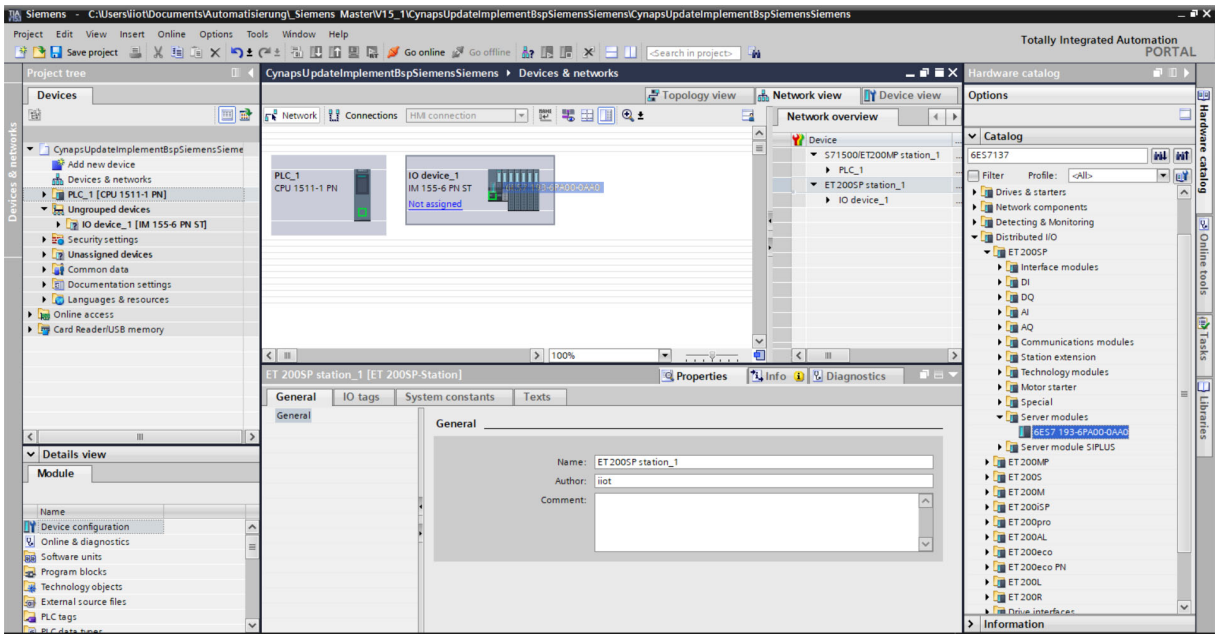




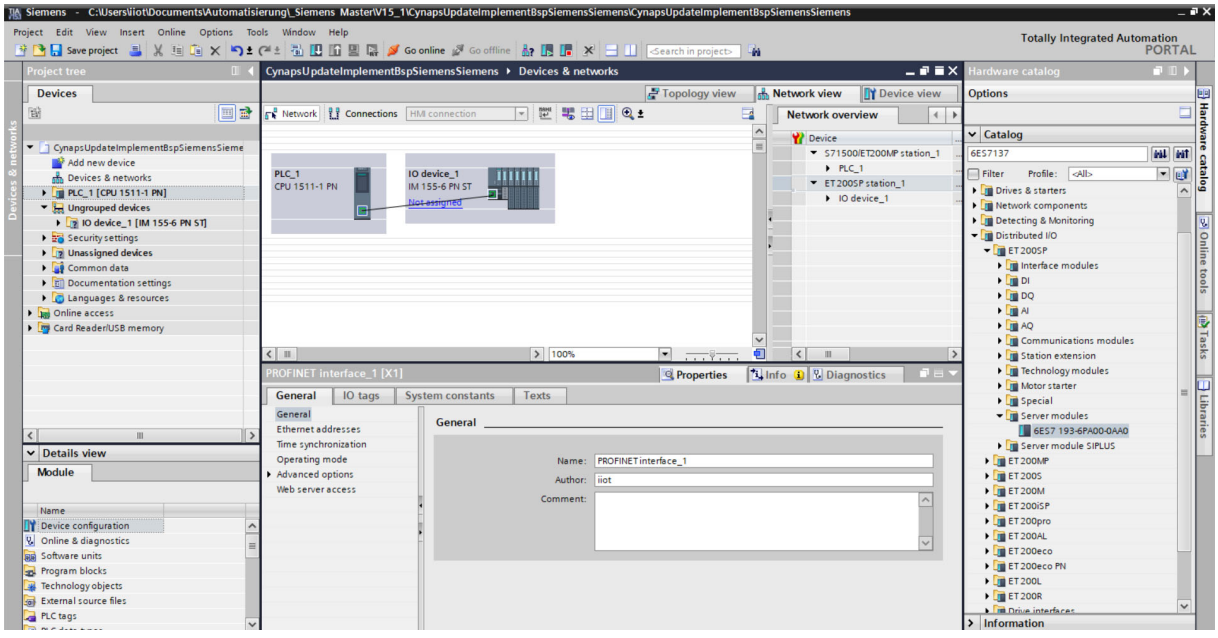
14. In the “ET 200SP” folder, open the “Communications modules”, “IO-Link Master” folders and select the master you are using.
15. Drag the master you are using and drop it into the interface module.

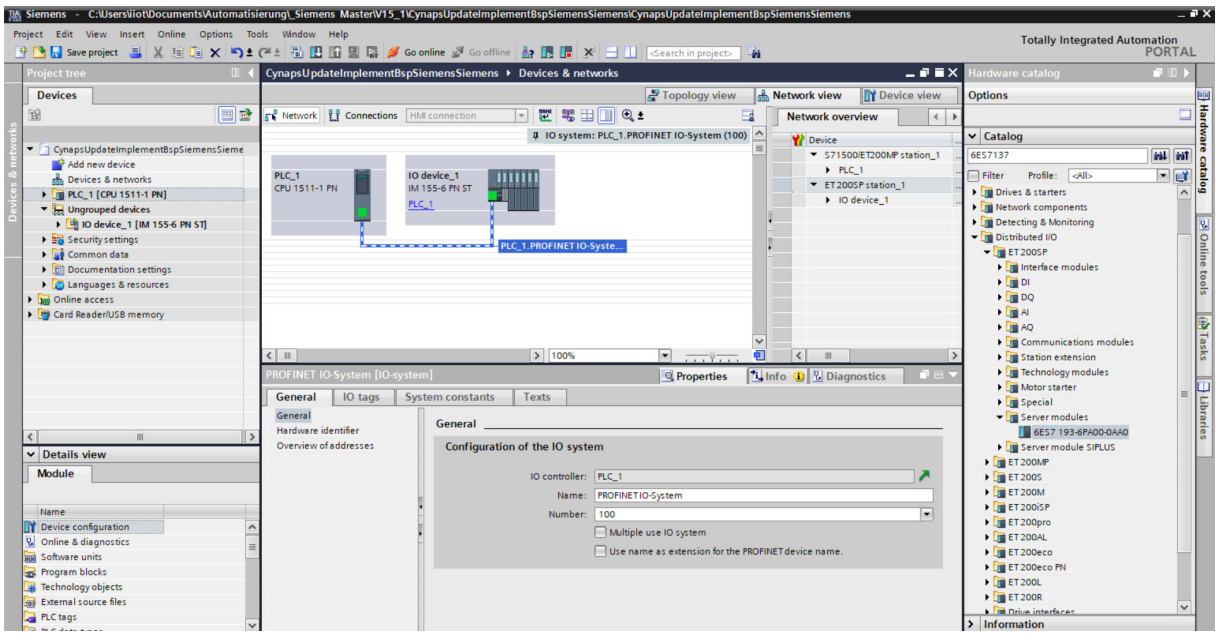


16. In the “ET 200SP” folder, open the “Server modules” folder.
17. Drag the server module as well and drop it into the interface module.



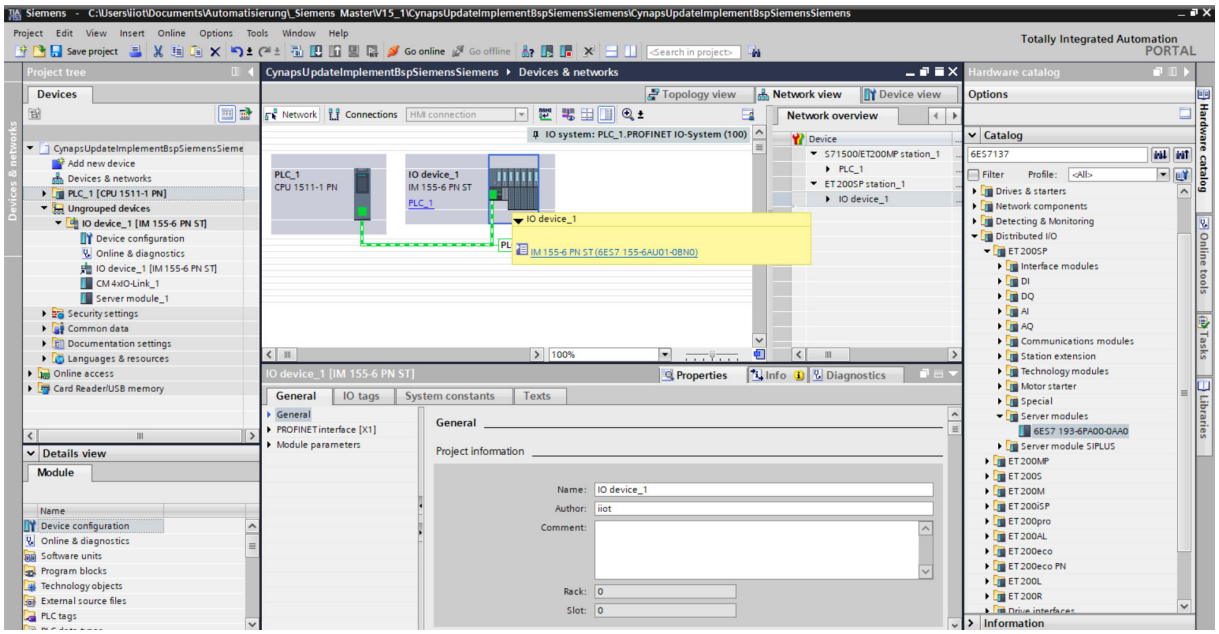
18. Drag & drop a connection from the interface of the CPU to the interface of the interface module in order to link these via PROFINET.



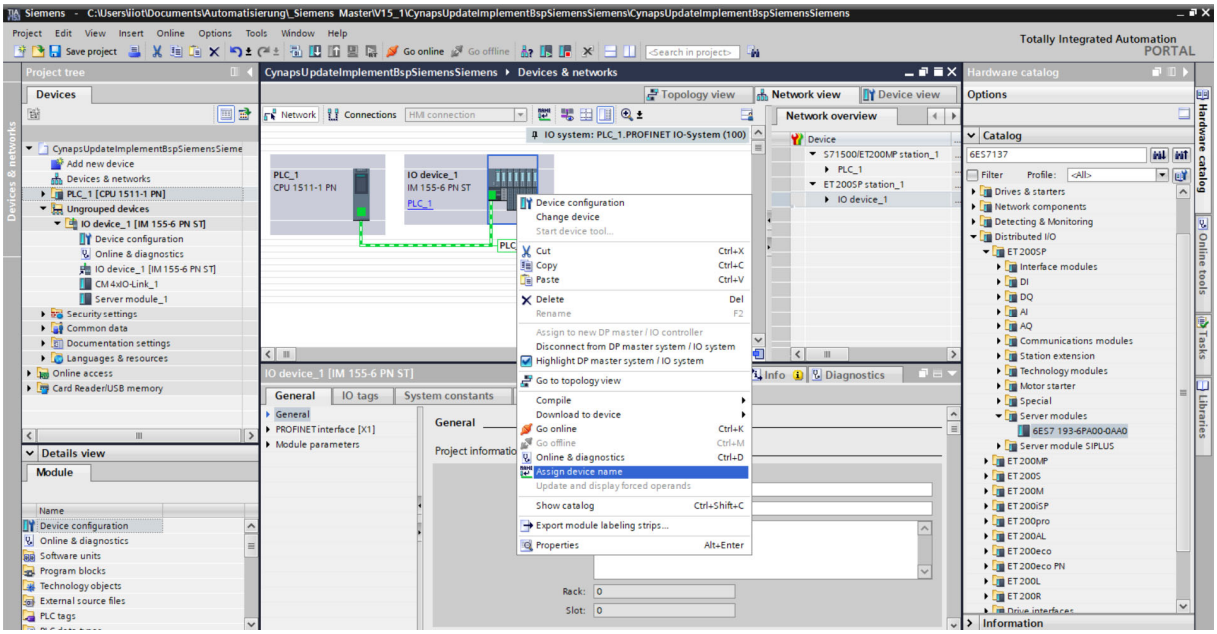


19. For successful communication, the device name of the reachable subscriber must match that of the hardware configuration. Check it as follows:

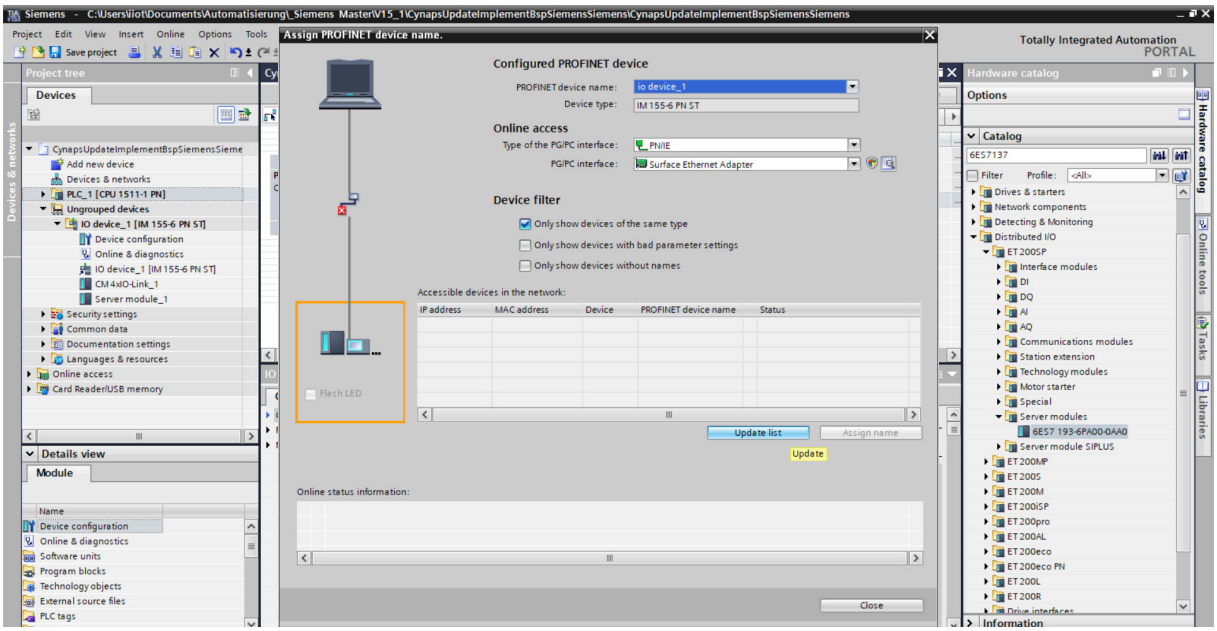
- Right click on the IO device.



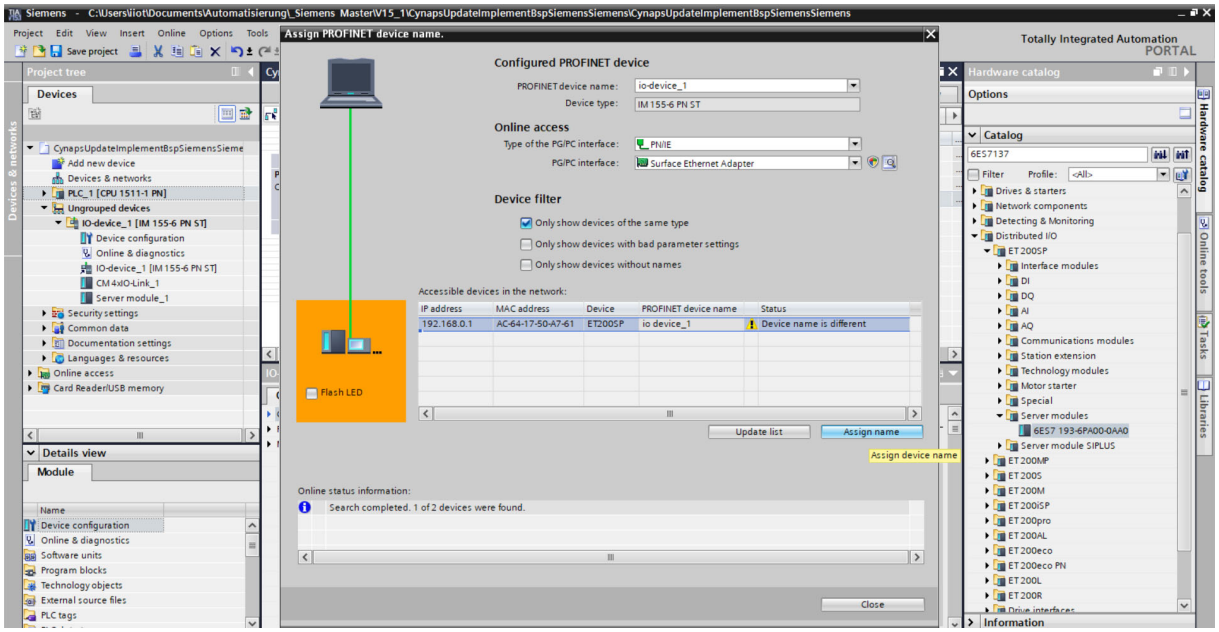
- Select “Assign device name”.



- Click on “Update list”.



- If the device name is different: Select the line of the participant and click “Assign name”.



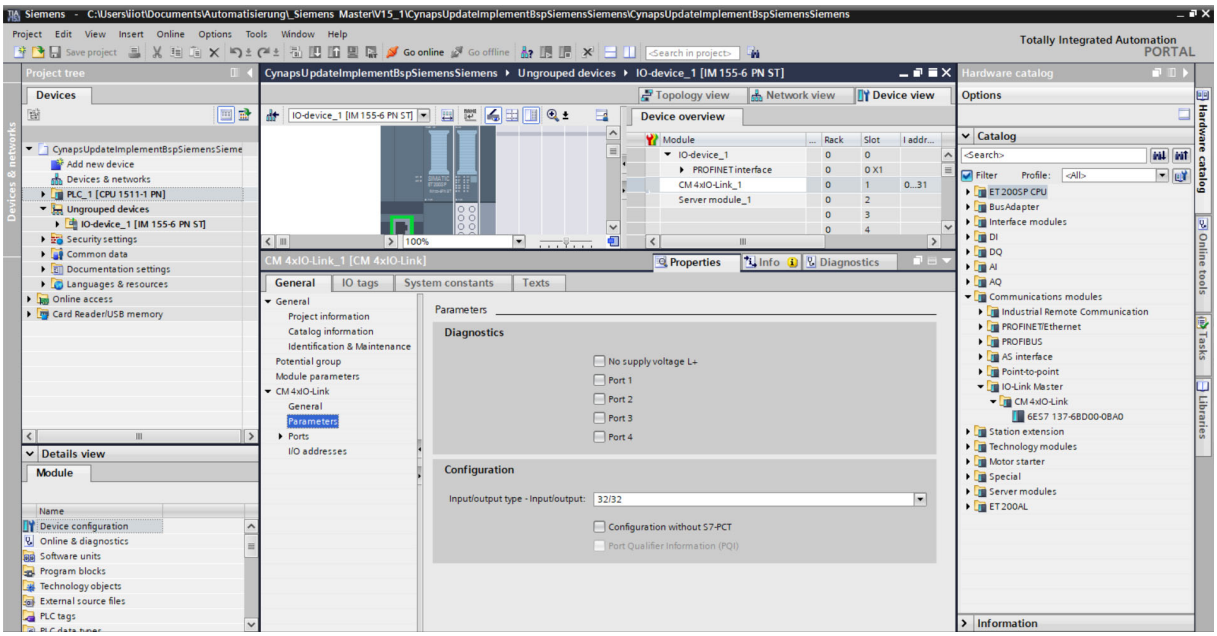
Result

➲ Successful communication between TIA Portal, CPU and IO-Link Master is possible.

3.2 Hardware configuration cynapse® with S7-PCT

Requirement

➲ When configuring the IO-Link Master in STEP 7, you have unchecked the option box “Configuration without S7-PCT”.



3.2.1 Loading IODD

Requirement

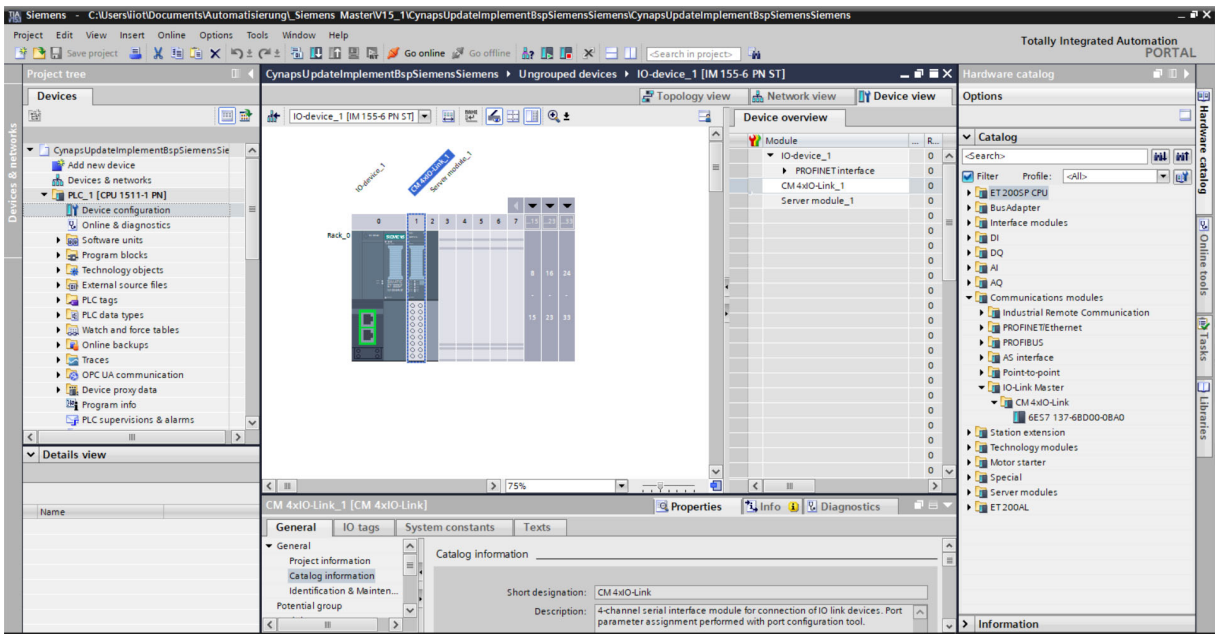
➲ You have obtained the current IODD cynapse® from the following sources:
IODD Finder (https://ioddfinder.io-link.com)

Introduction

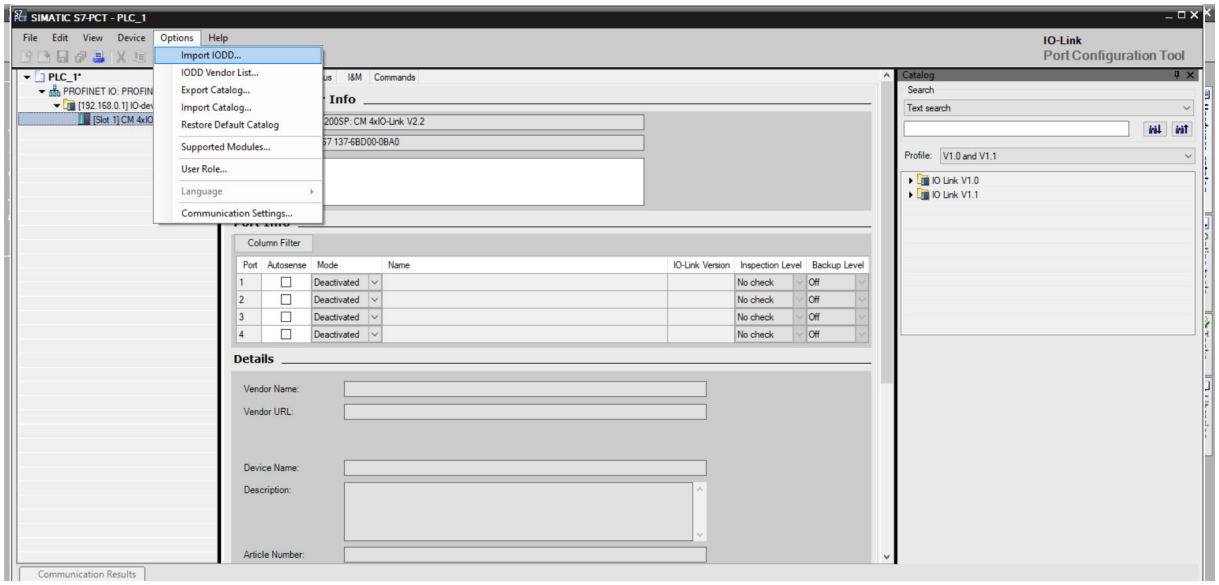
In the following, download the cynapse® IO device description to add the sensor to the selection catalog of the PCT.

Procedure

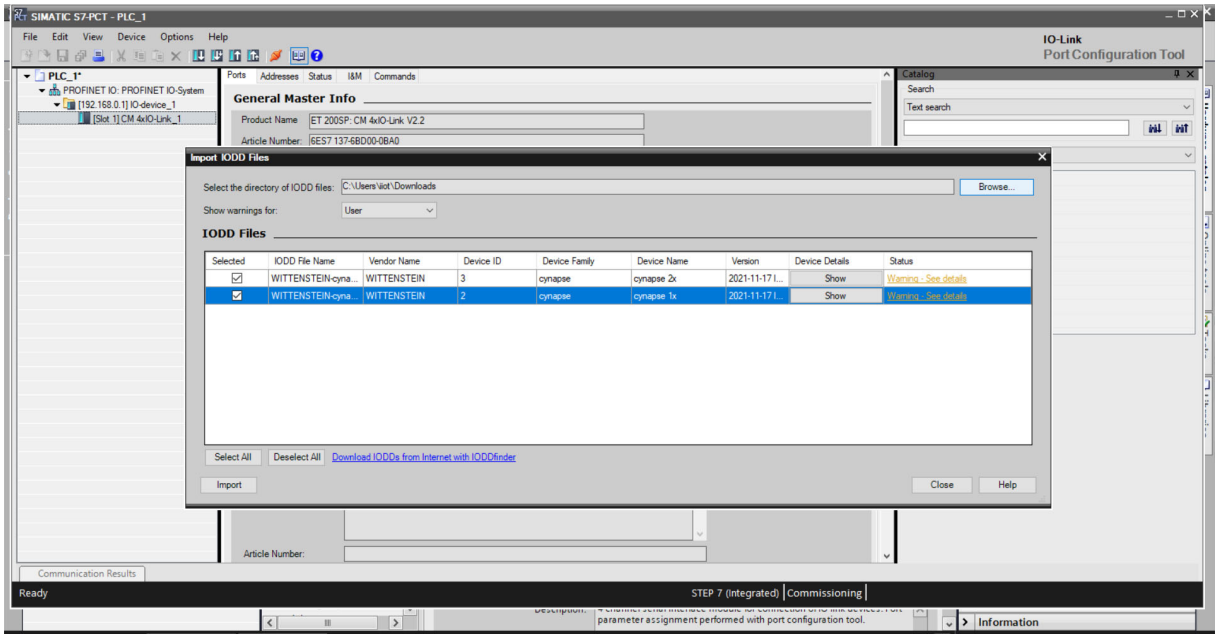
1. In the TIA Portal, open the device configuration.
2. Switch to the device view of the IO device.



3. Move your mouse pointer over the CM 4xIO-Link module.
4. Right-click to open the dialog box.
5. Click on “Start Device Tool”.
6. Download the wizard to import the IO device description via Options > import IODD.



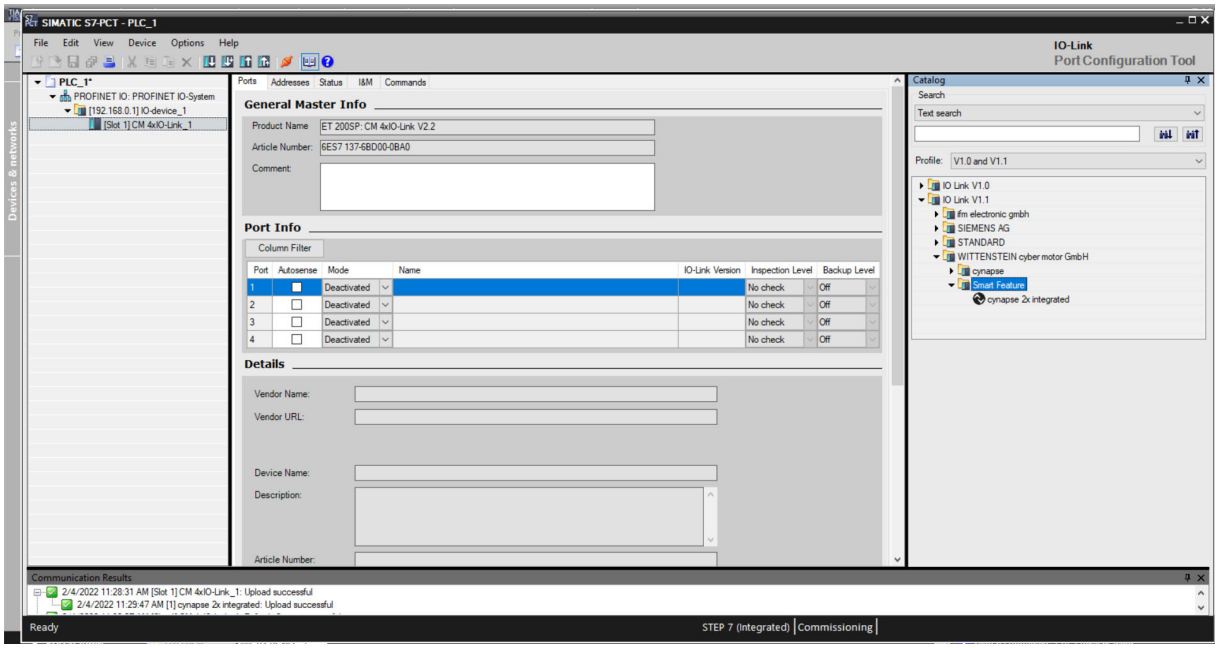
7. Select the IODD .xml via “Browse...” in your location. If you have not saved them yet, you can do so by clicking on “Download IODDs from Internet with IODDfinder”.



8. Click on “Import”.

Result

➲ cynapse® is included in the PCT catalog and can be detected online or configured manually.



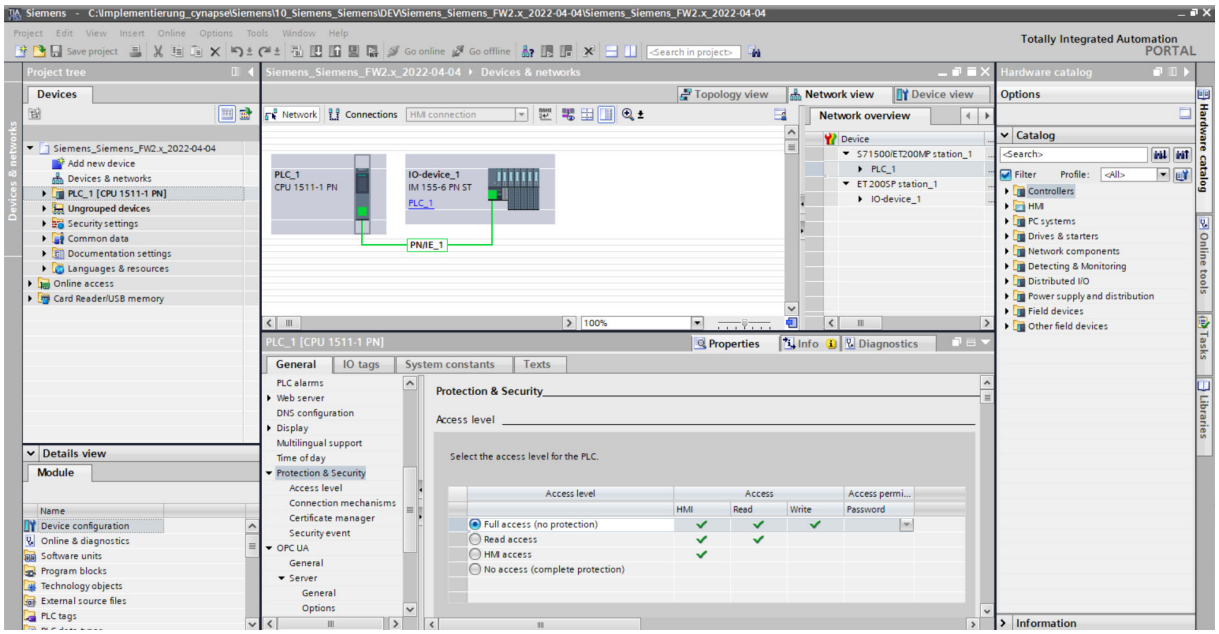
3.2.2 Import cynapse® online

Requirement

- ➲ You have read the current IODD of cynapse® as described in Chapter 3.2.1 “Loading IODD”.
- ➲ The hardware configuration is error free.
- ➲ Communication with the hardware is possible and the S7-PCT is open.

ⓘ Information about possible errors / online reading of the device is not possible:

- In order for the S7-PCT to establish a connection, full access must be granted in the CPU security settings. In addition, no password may be assigned.



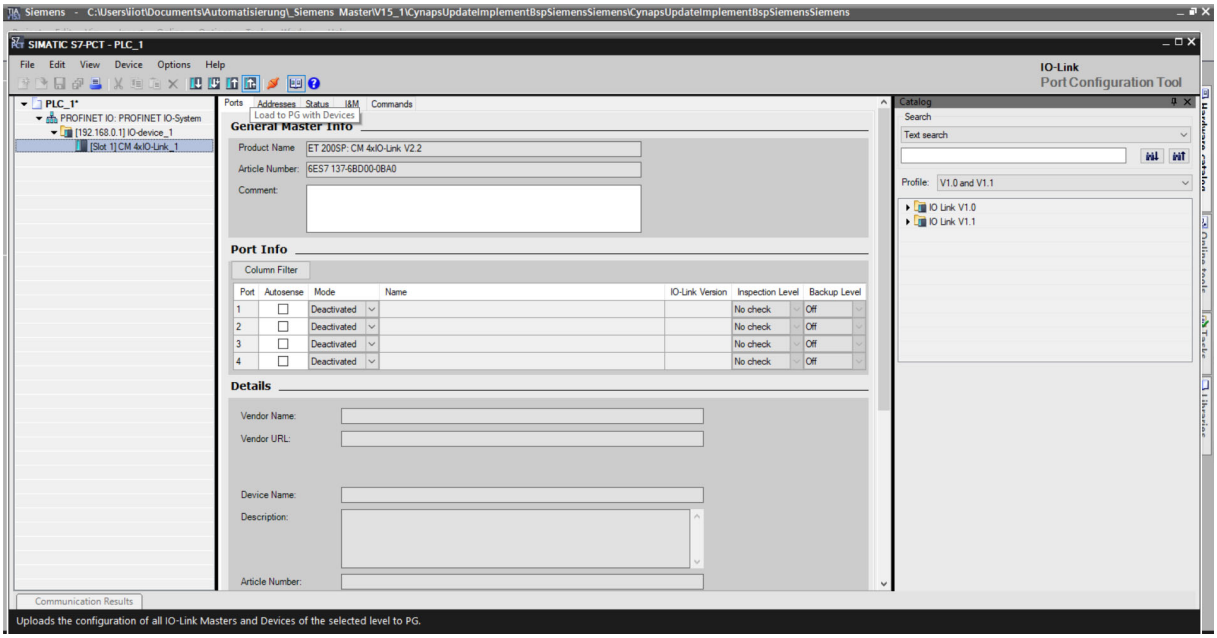
- PROFINET device names of the online and offline configuration must match, otherwise a communication is not possible (help for this in chapter 3.1 “Hardware configuration Siemens components”).

Introduction

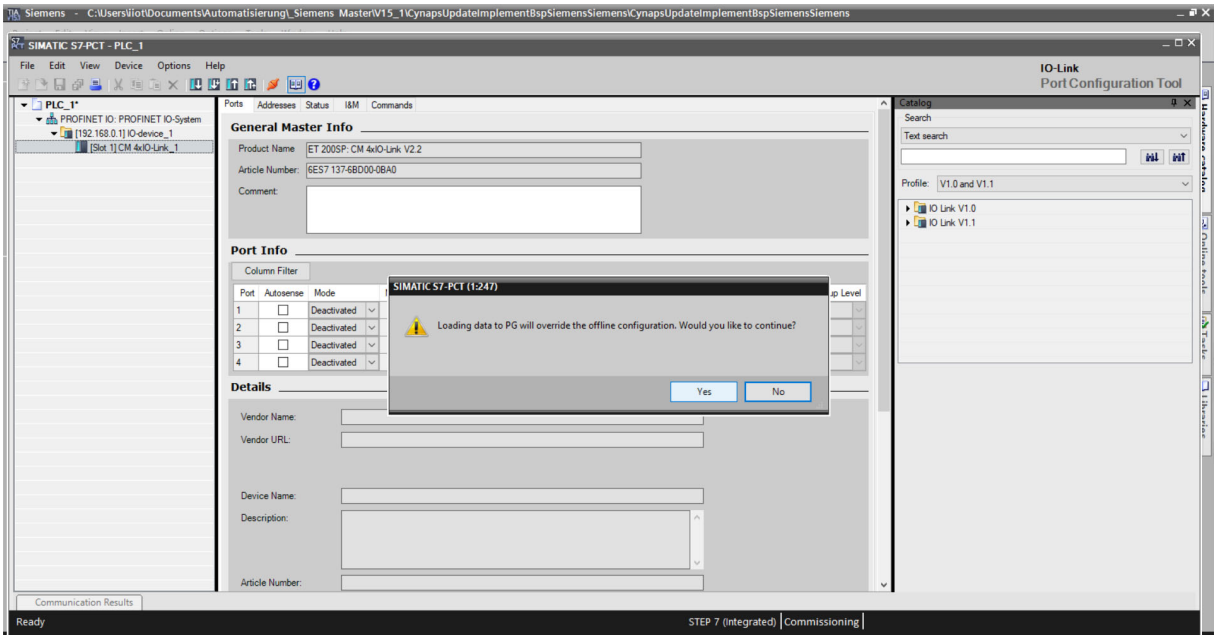
In the following, you will read the connected hardware on the IO-Link Master using the S7-PCT.

Procedure

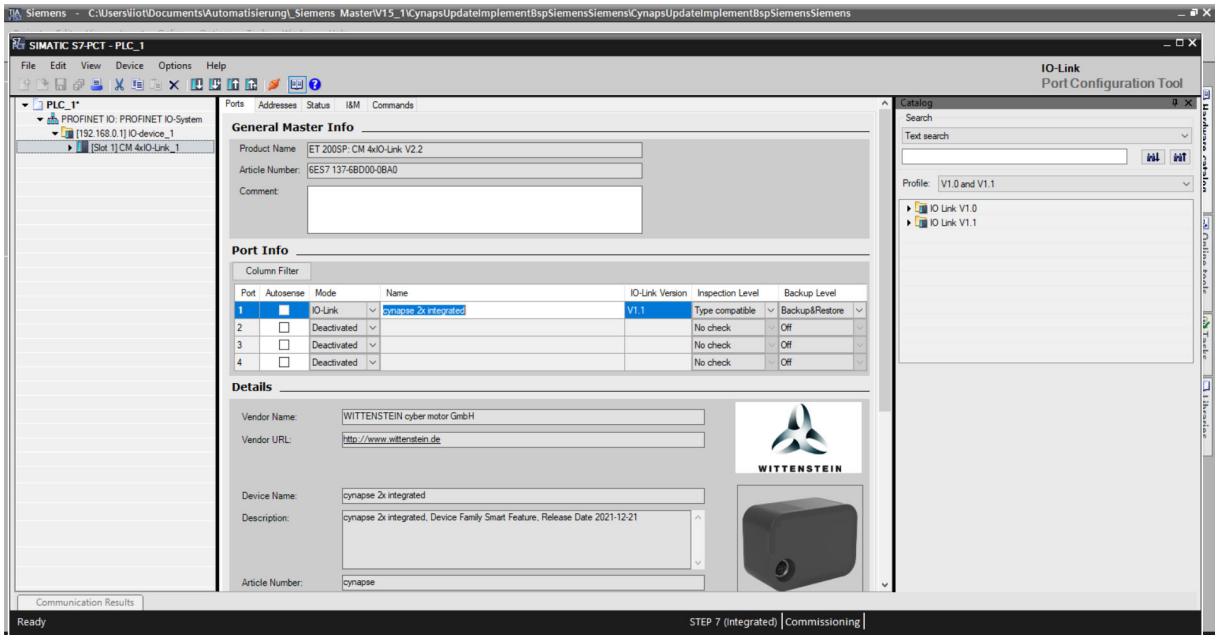
1. Click the “Load to PG with Devices” button to find devices online.



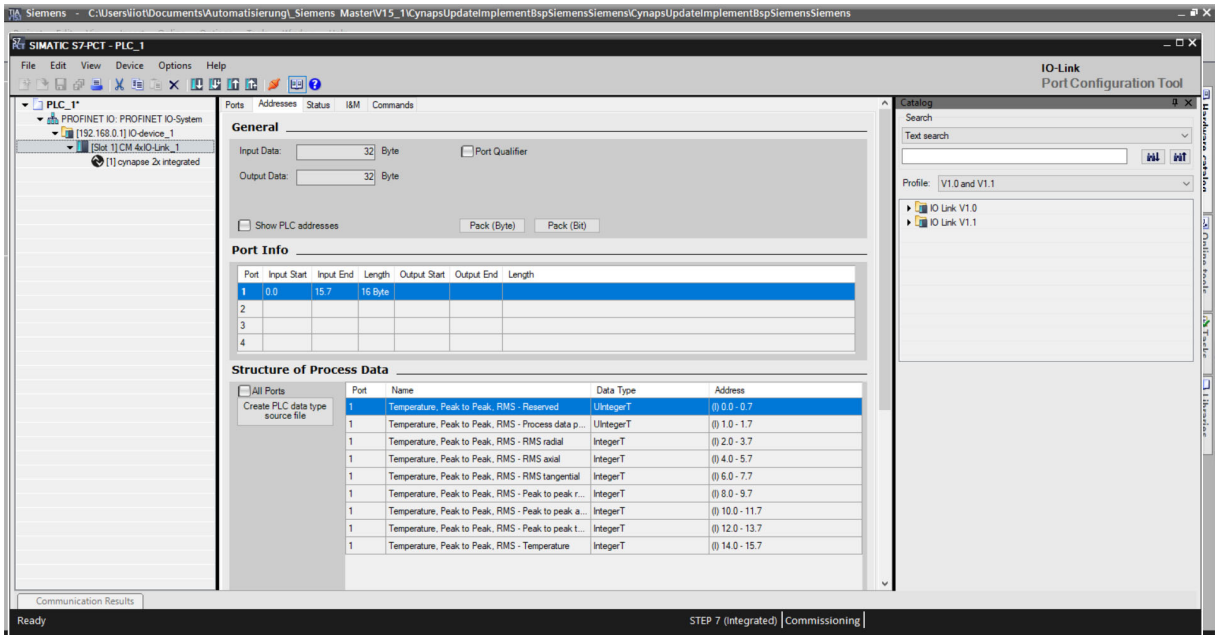
2. Confirm the warning with “Yes”.



3. After successfully adding devices, cynapse® will appear at the corresponding port.



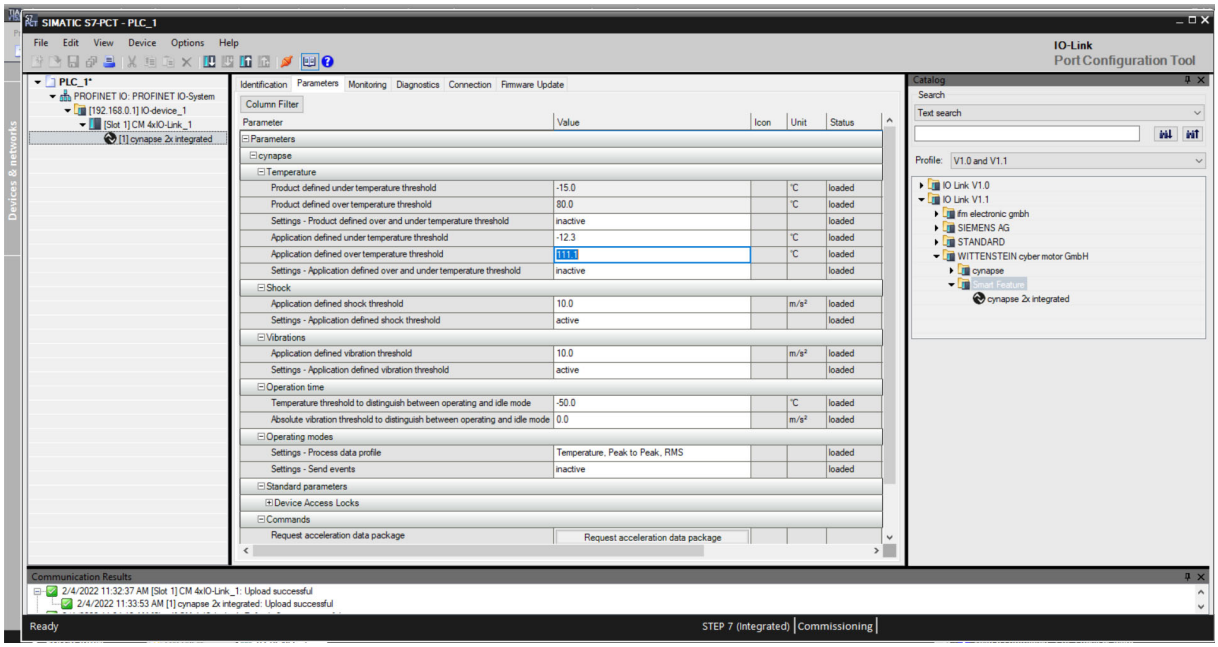
4. Click on the “Addresses” tab to view the cynapse® process data.



5. Close the S7-PCT and confirm the save to save the configuration read online.

Result

- ➲ cynapse® was searched, found and saved online. The hardware is present when the S7-PCT is reopened.
- ➲ In addition, the S7-PCT can display further information such as warning thresholds, pending error codes, hardware version or firmware version. These cannot be changed in this view. For more information, see Chapter 5 “Parameter”.



4 Process data

4.1 Definition

Process data is understood to mean cyclically communicated data between the IO-Link Master and control. In each cycle, these data are transferred. The process data sent by cynapse® depends on the version status of the hardware and software. For more information, see the cynapse® operating manual.

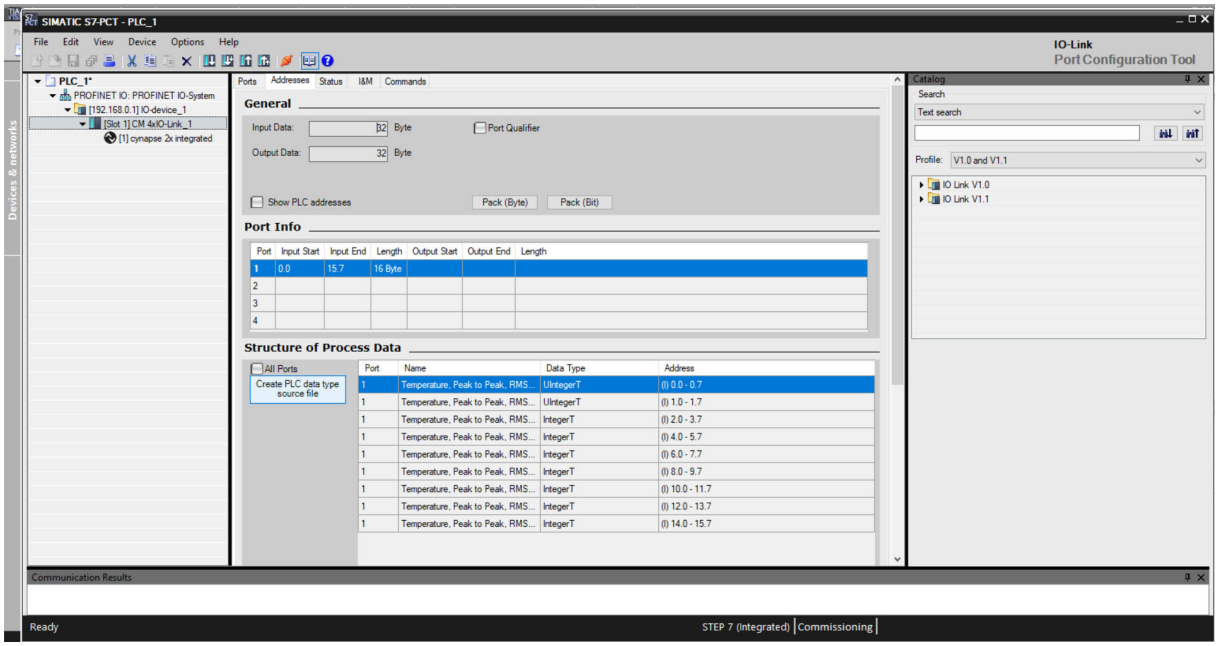
4.2 Providing PLC program process data

Introduction

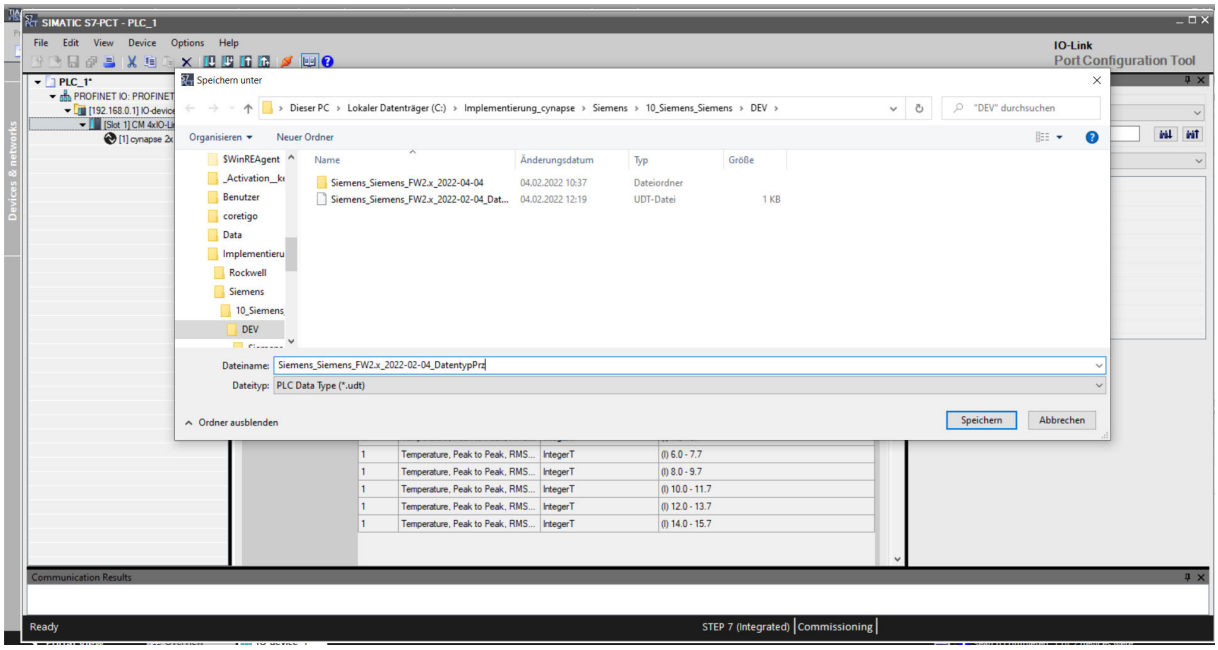
In the following, you generate a PLC data type source file, read this into the control and create a variable of this type in the variable table.

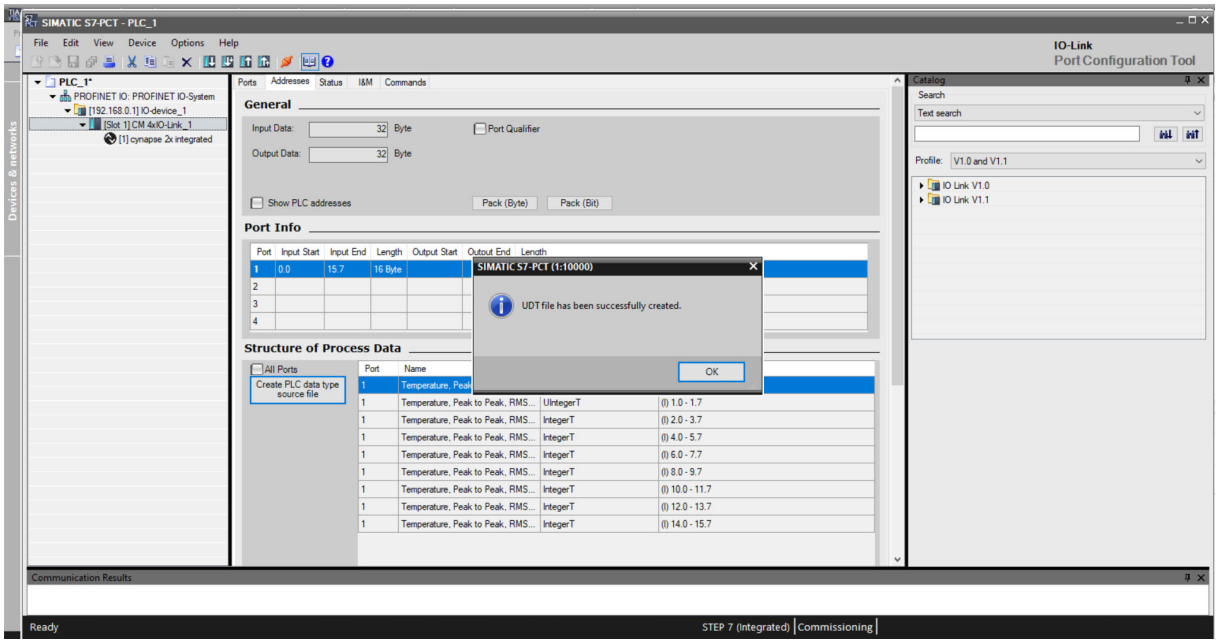
Procedure

1. Open S7-PCT.
2. Click on “Create PLC data type source file”.



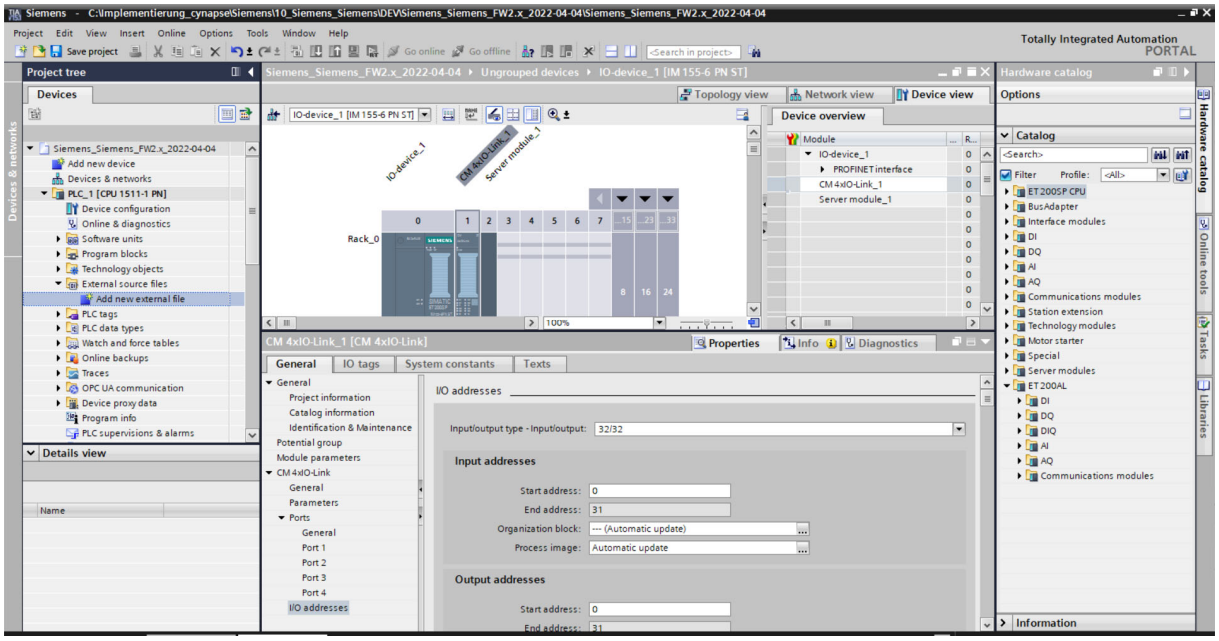
3. Give a name and save the .udt-file by clicking on “Save”.



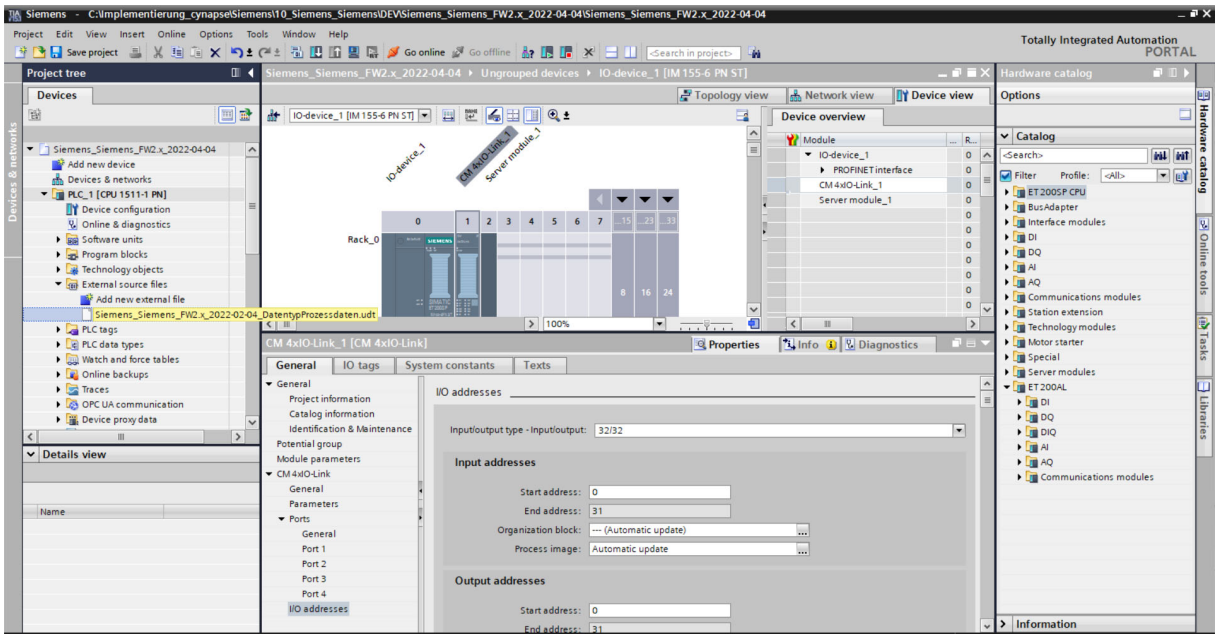


4. Close S7-PCT.
5. In the project tree, select “External source files”.

6. Double click on “Add new external file”.

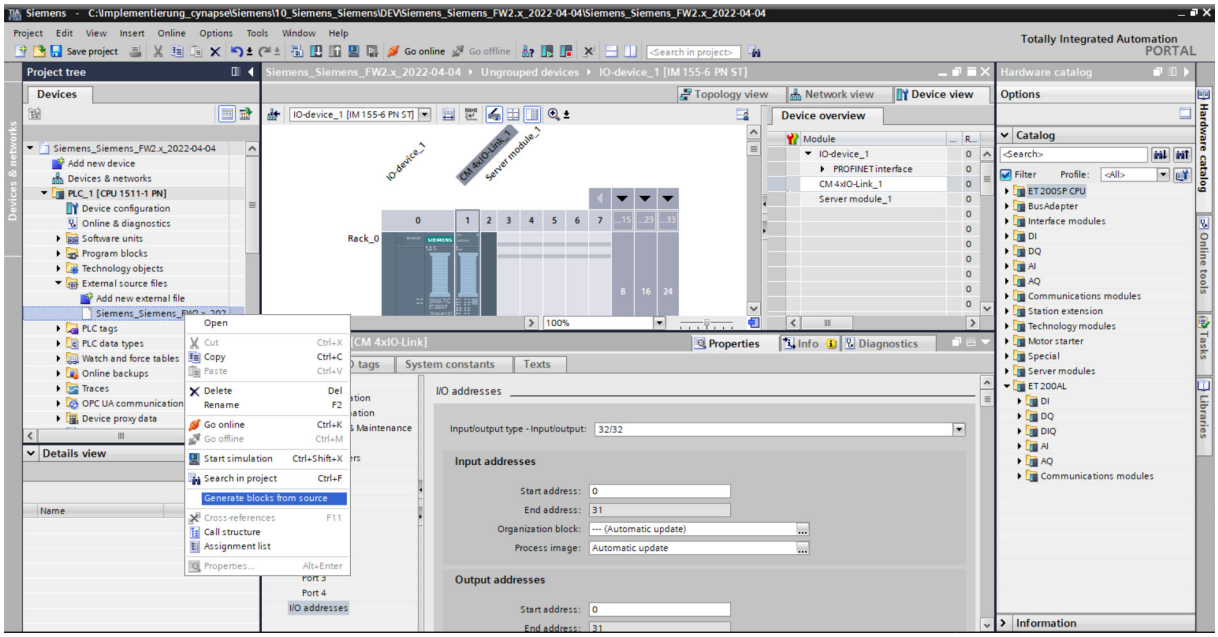


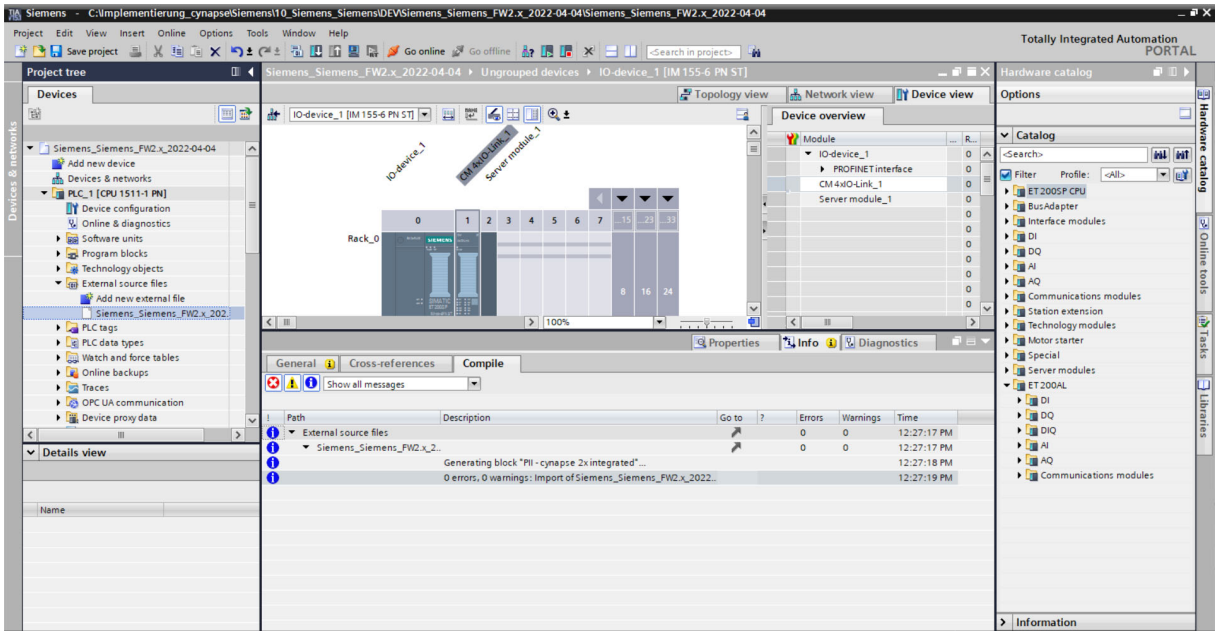
7. Select the previously saved .udt file.
8. This is now under the tab “External source files”.



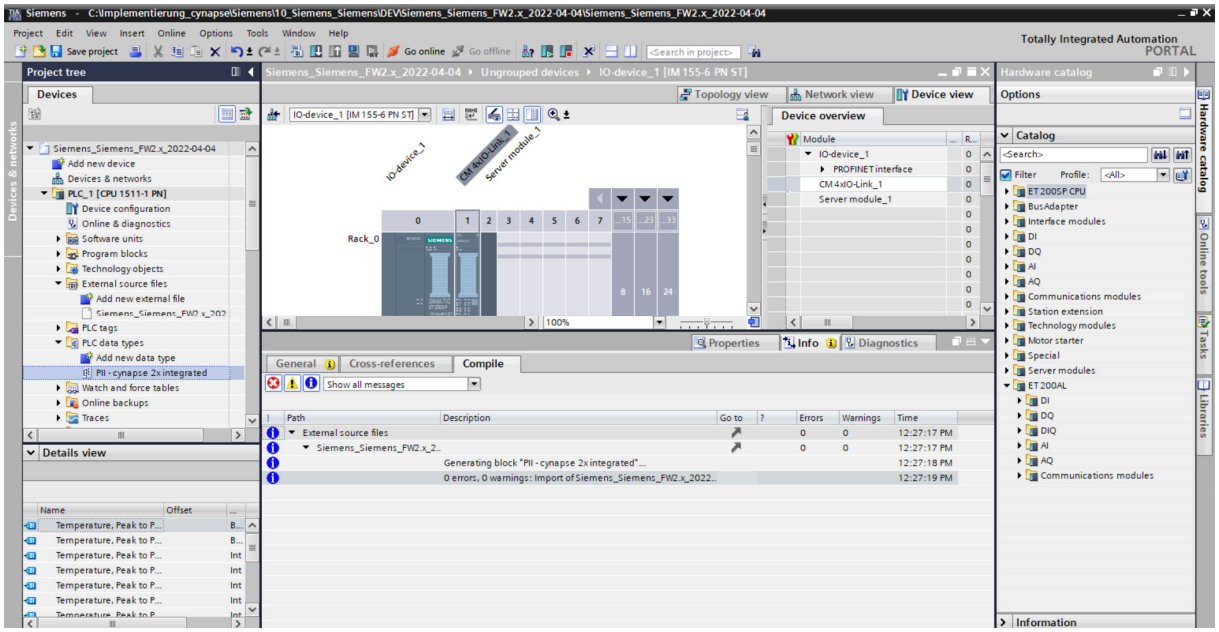
9. Right-click on the .udt file.

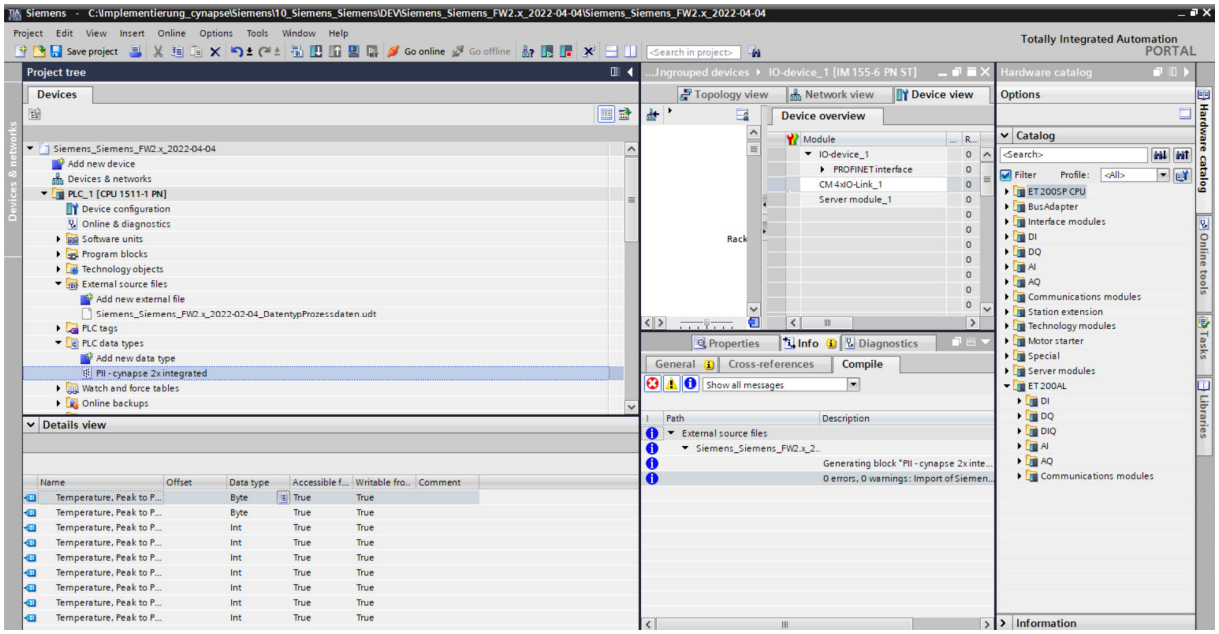
10. Click on “Generate blocks from source file”.



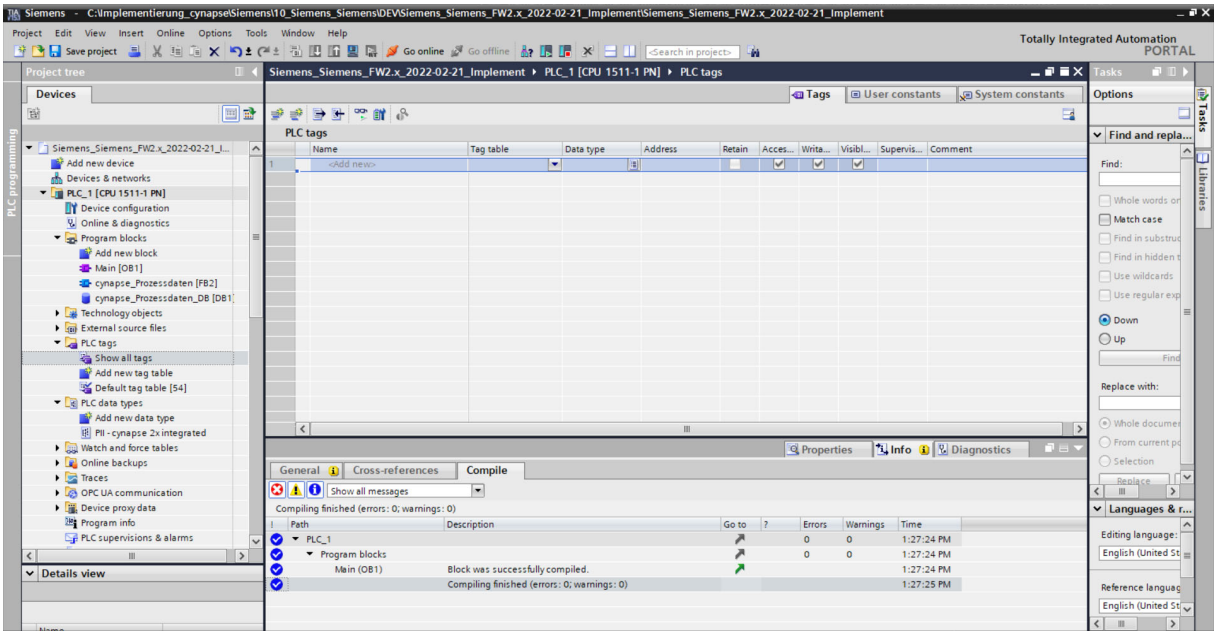


11. The data type exists under “PLC data types”.

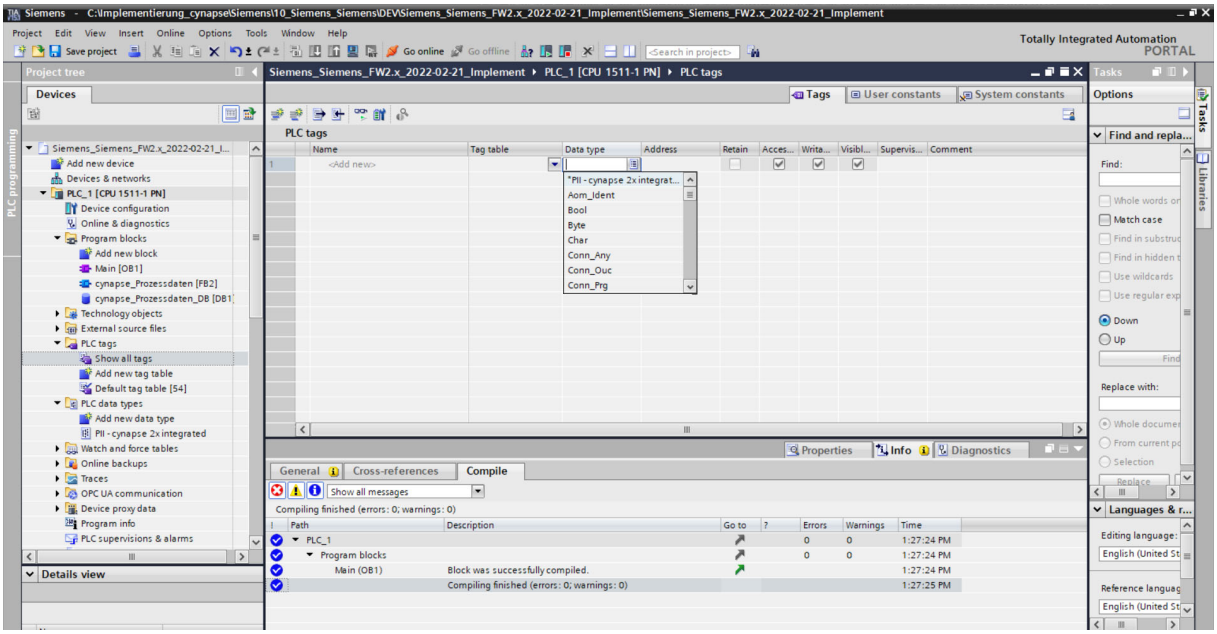




12. Under “PLC tags”, open “Show all tags”.



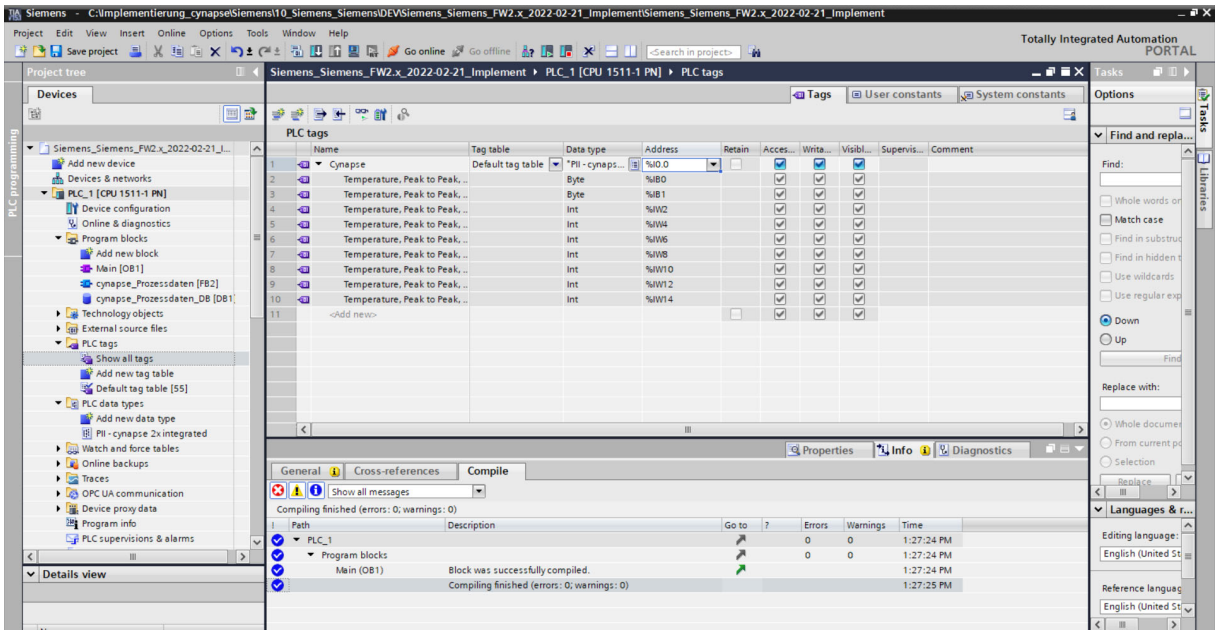
13. Under Data type, select the previously generated data type.



14. Confirm your input with Enter.
15. Adjust any linked addresses to the hardware.

Result

➲ cynapse® is linked as hardware and the output variables can be linked to blocks in the PLC program.



4.3 Read process data using the “cynapse process data” FB

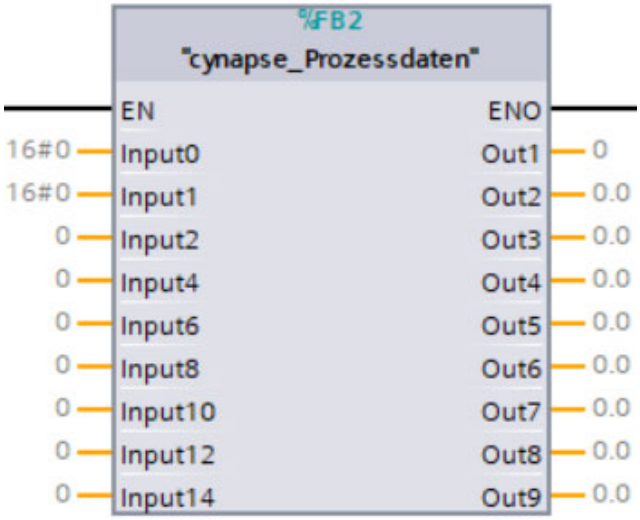
Requirement

➲ You have obtained a sample project for reading out the process data from the following source:
cybertronic-support@wittenstein.de

Introduction

In the following, you will read process data from a sample project using a function module. The latter takes over the scaling of the measured values.

cynapse® offers different process data formats to offer different data for further processing while maintaining the same process data length. These process data can be selected by the parameter Settings. For more information, see the operating manual cynapse® and Chapter 5.4 “Writing parameters”.

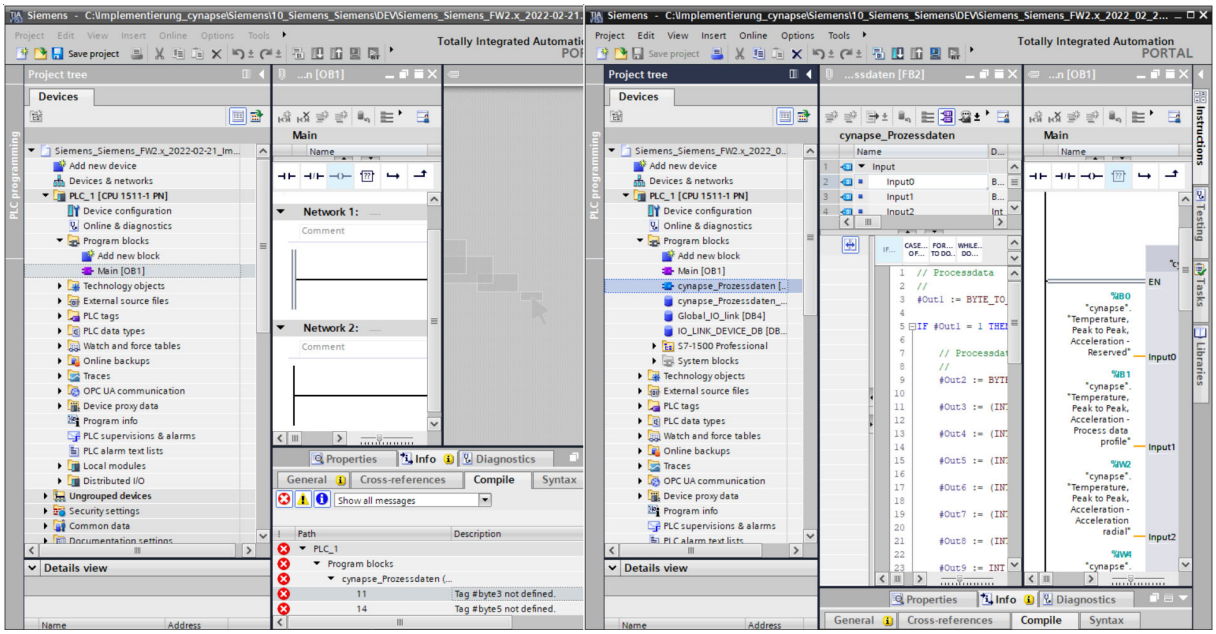


Input/output	Data type	Function
Input 0	BYTE	Byte 0 reserved
Input 1	BYTE	Byte 1 process data profile
Input 2-14	INT	Process data profile dependent input variables – more information on this in the operating manual
Out 1	INT	-
Out 2	REAL	Current active process data profile
Out 3-9	REAL	Scaled values of inputs 2-14

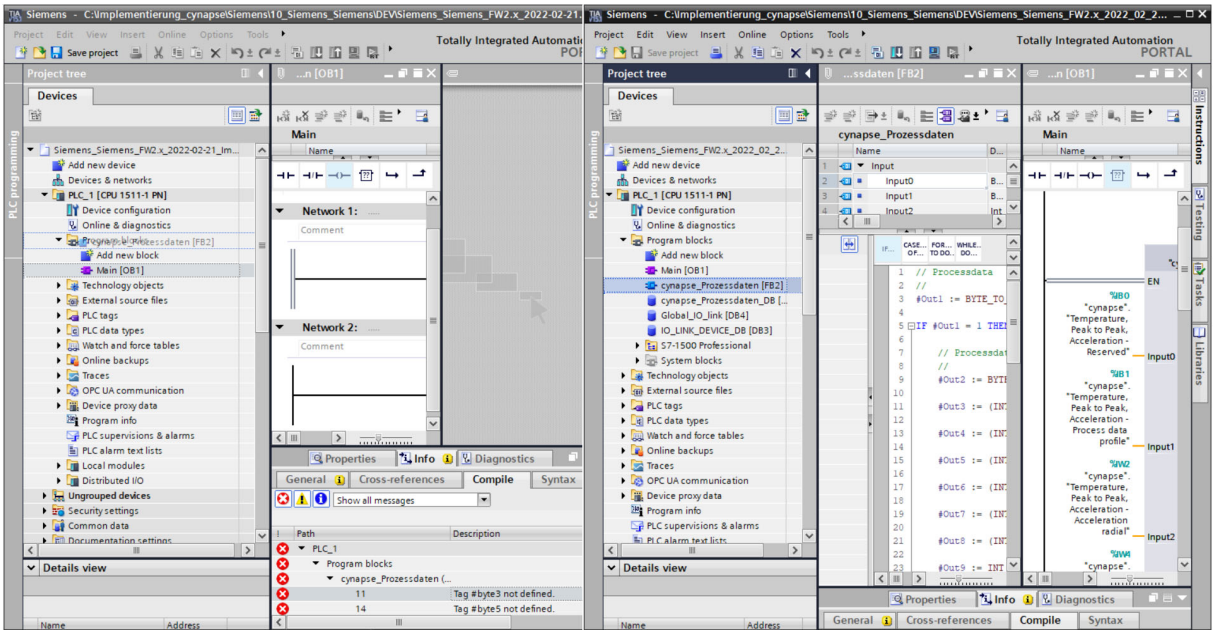
Tbl - 1

Procedure

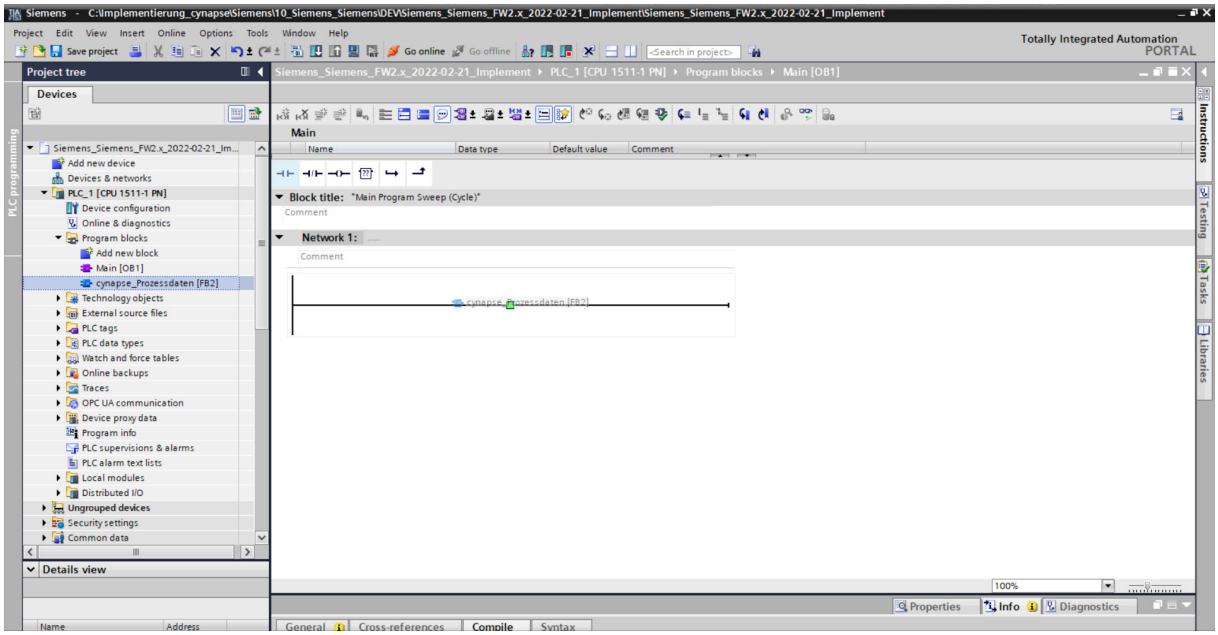
1. Open the resulting sample project.
2. In parallel, open the project in which you want to read process data.
3. Select the “cynapse_Process Data” FB in the sample project.



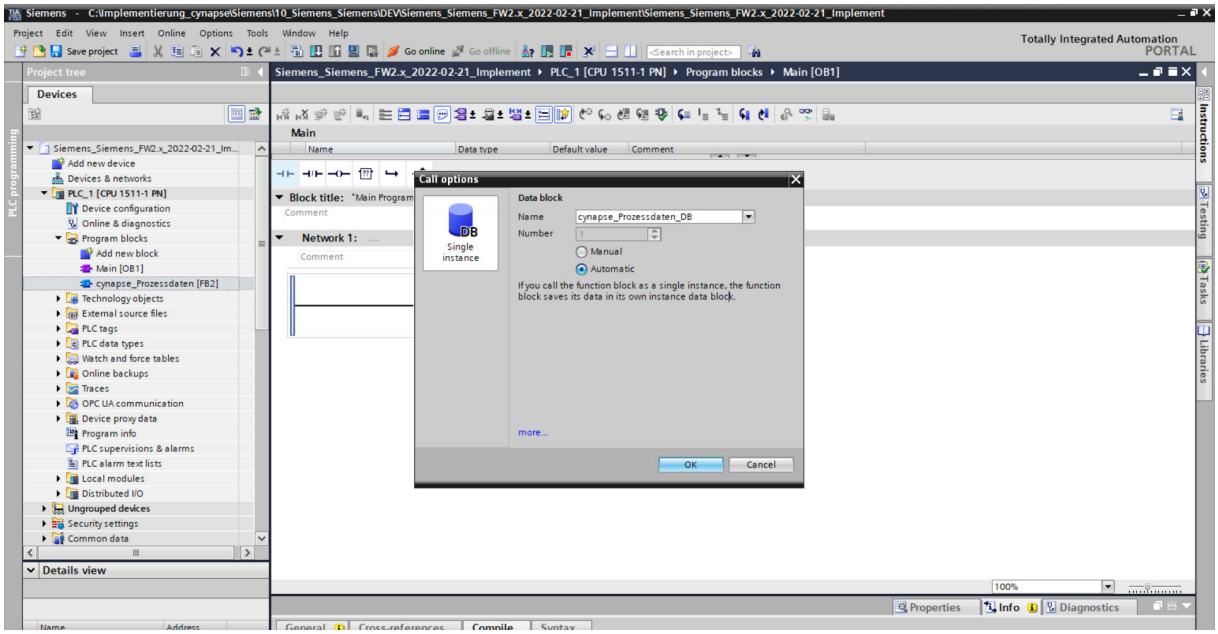
4. Drag and drop it into your project under “Program blocks”.



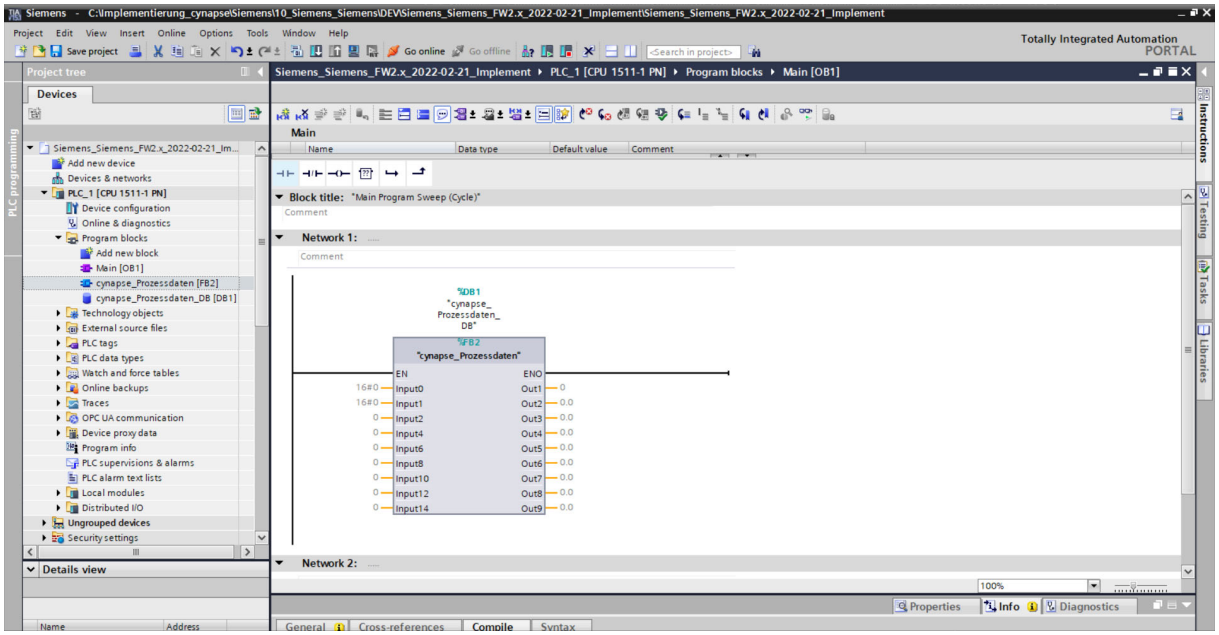
5. Drag and drop the FB into the main block.



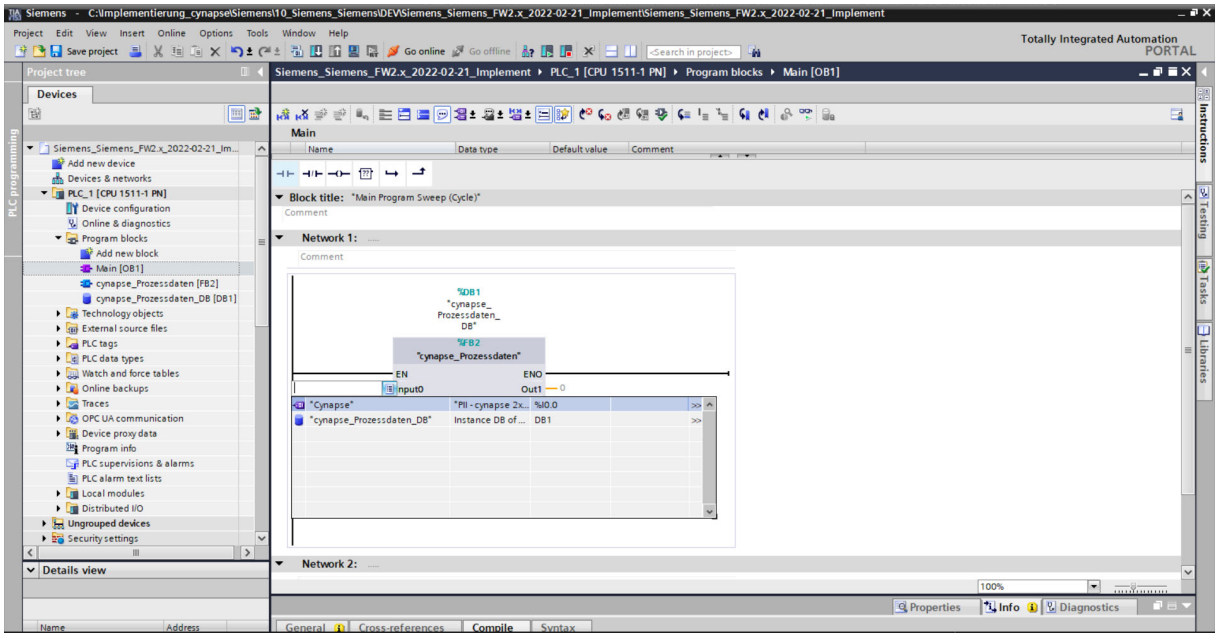
6. Create the DB by clicking on “OK”.



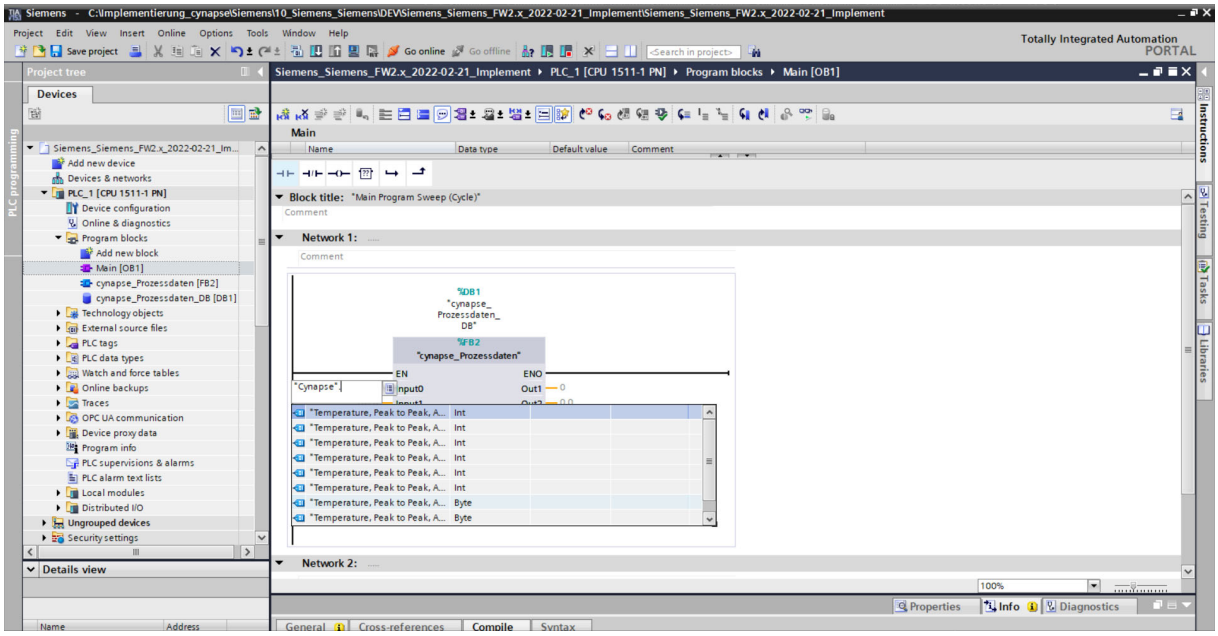
7. The block is contained in a network of the main block.



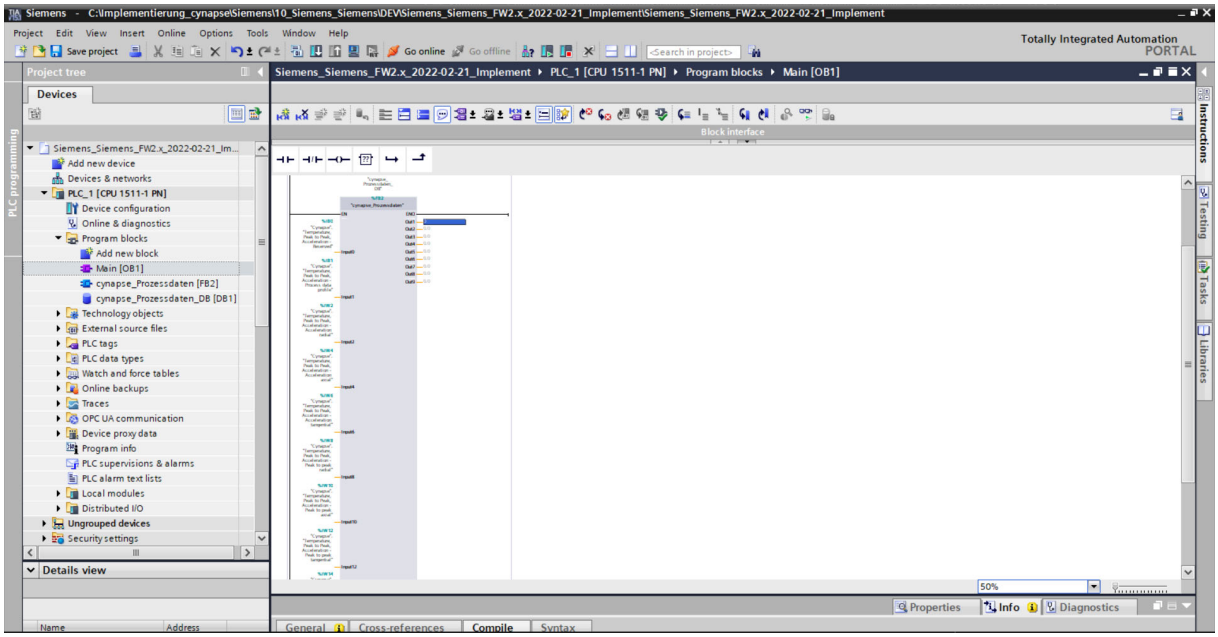
8. Double-click on the inputs to display the inputs created in variable tables (see chapter 4.2 “Providing PLC program process data section “Result”).



9. Click on the “cynapse” data type.



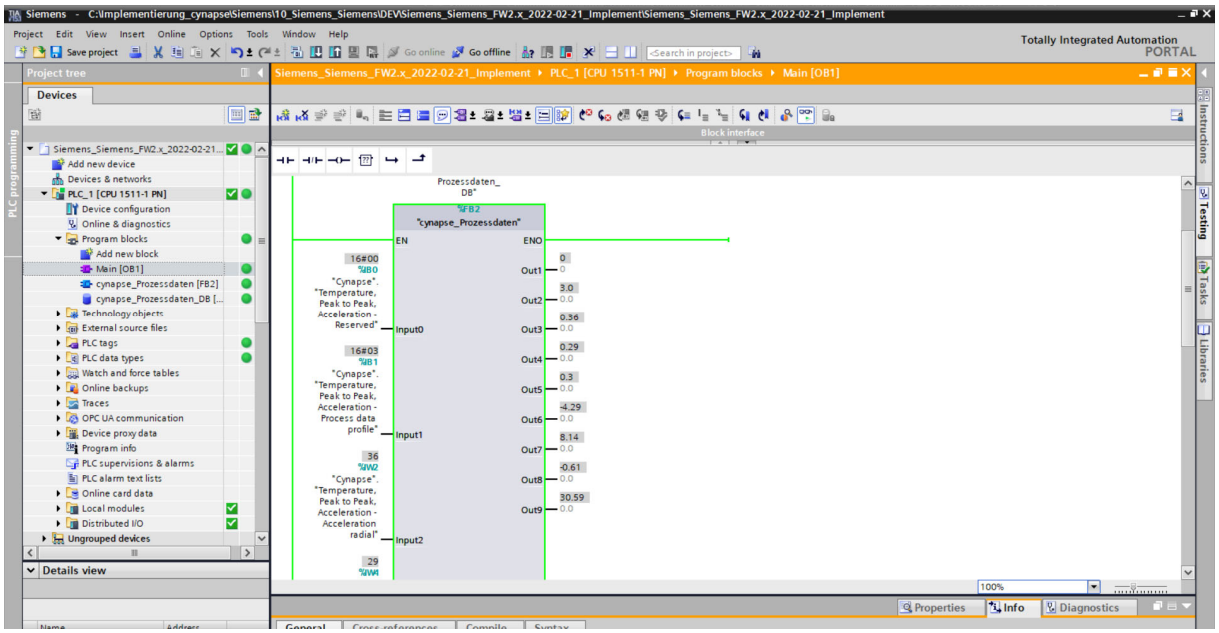
10. Link the inputs according to the information in the cynapse® operating manual.



Result

➲ The scaled process data are present at the outputs of the module according to the selected process data image and can be used further in the program.

After compiling and downloading the data can be Go online observed online.



5 Parameter

5.1 Definition

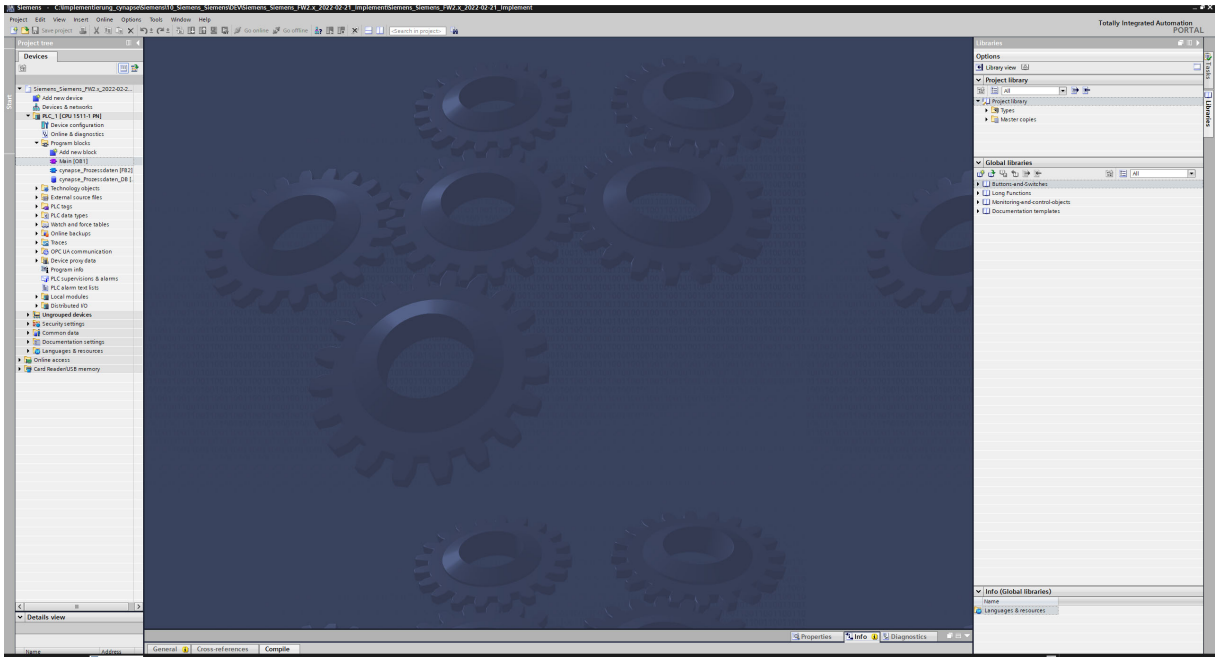
Parameters are understood to be acyclically communicated data. In this way, device parameters such as device information, threshold values or diagnostic data of an IO-Link device (e.g cynapse®) can be read or written. The data on the device is uniquely addressed with index and subindex.

For more information about the index and subindex as well as the structure of the data set, see the cynapse® operating manual.

5.2 Integrating Siemens block for parameter reading/writing into program

Requirement

- ➲ You have obtained the LIOLink library from the Siemens website and know the location.
- ➲ Your TIA project is open and the Libraries task card is open.



ⓘ **Note:** A duplicate request to a device via the IO_LINK_DEVICE module is not possible. Since parameters are read, parameters are written, events are read out and blob data are accessed on this block, these are to be locked against one another.

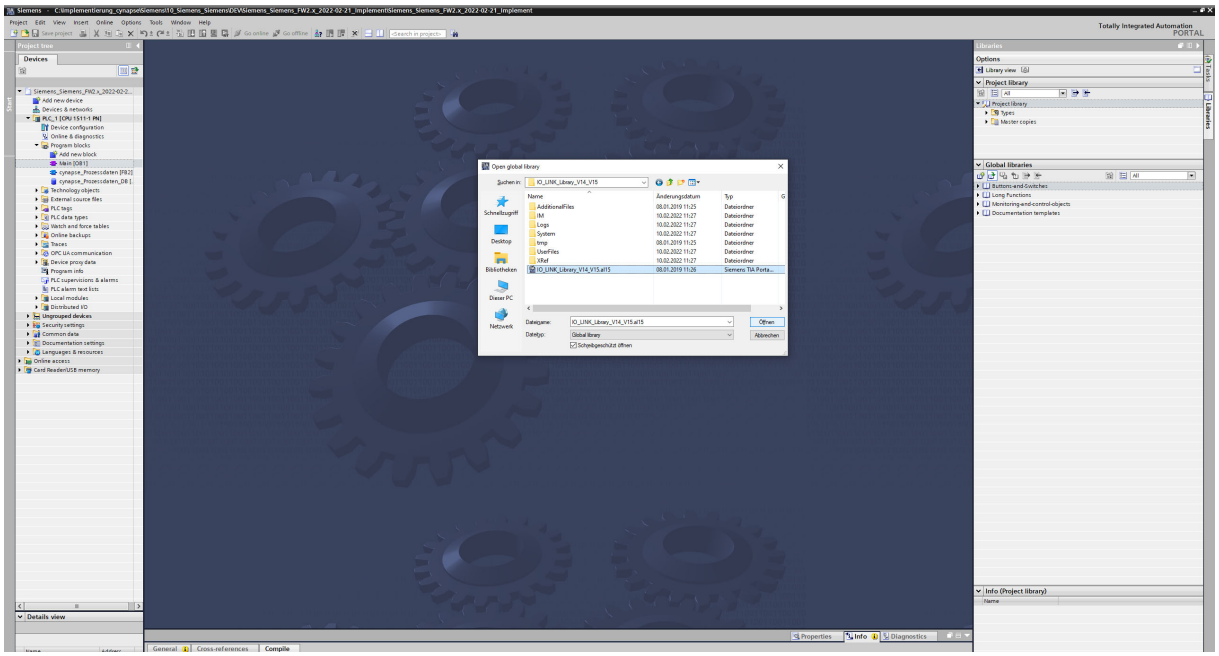
Introduction

In the following, you integrate the functional module “FBIoLinkDevice” published by Siemens into your project and determine project-specific input variables of the module.

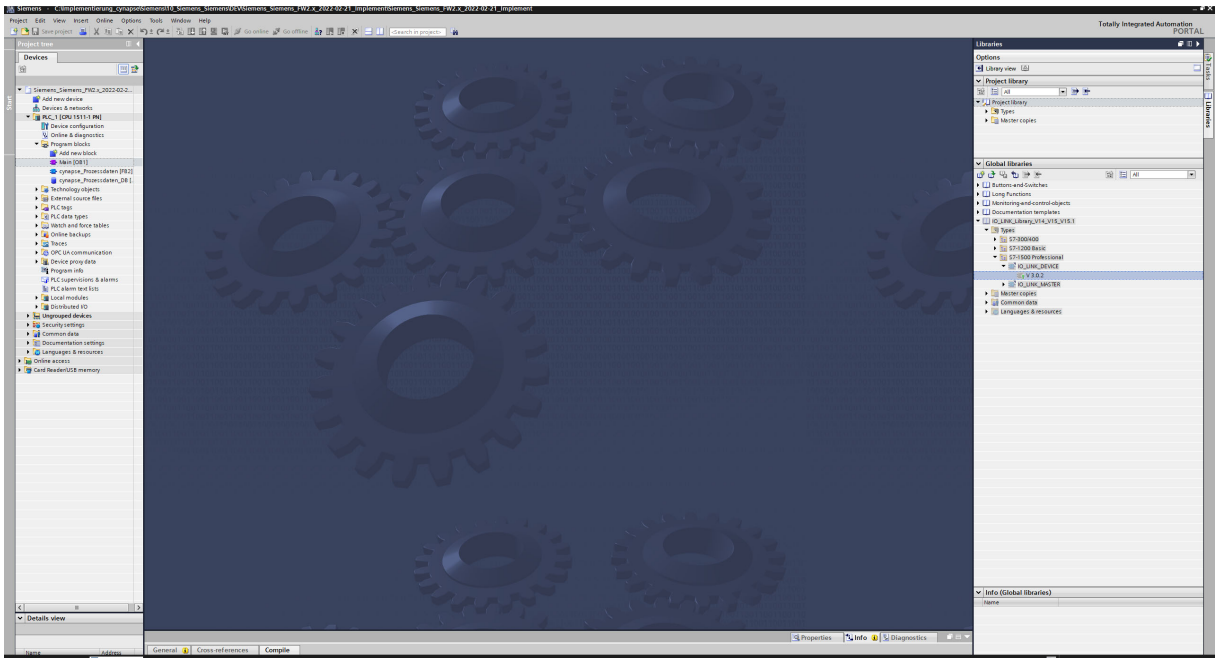
With the help of this function module, you can read program parameters, measured values and diagnostic data from an IO-Link device or write device parameters to an IO-Link device or send system commands.

Procedure

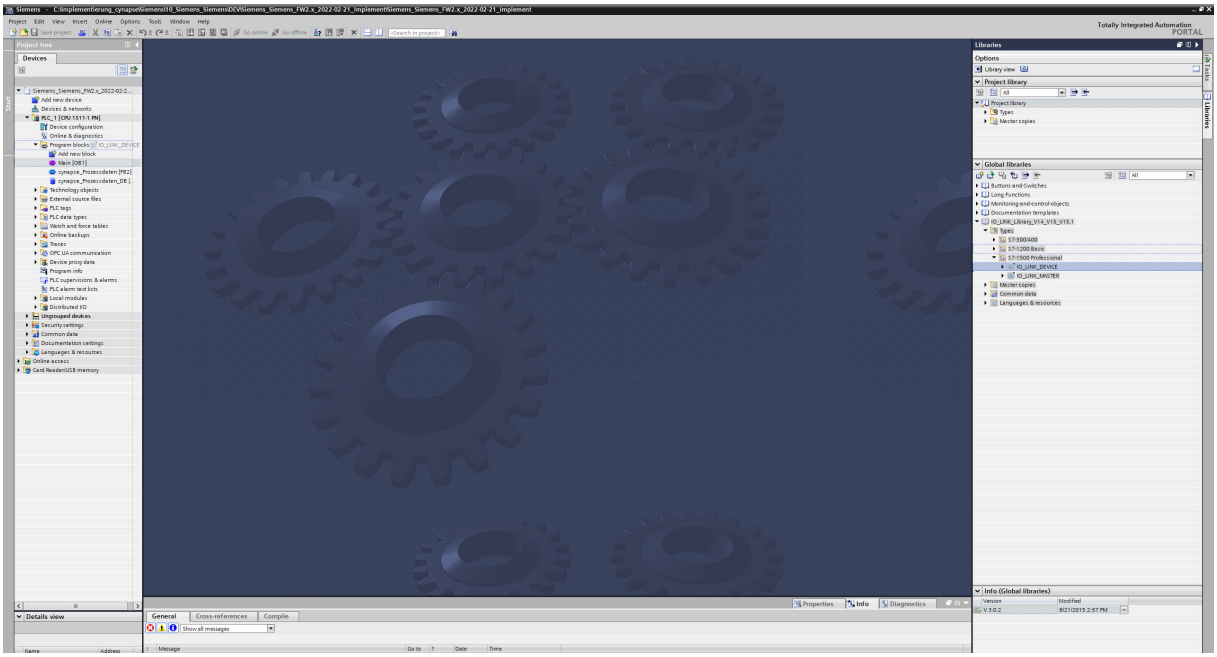
1. Click the “Open global library” button.
2. Select the library in the known location.



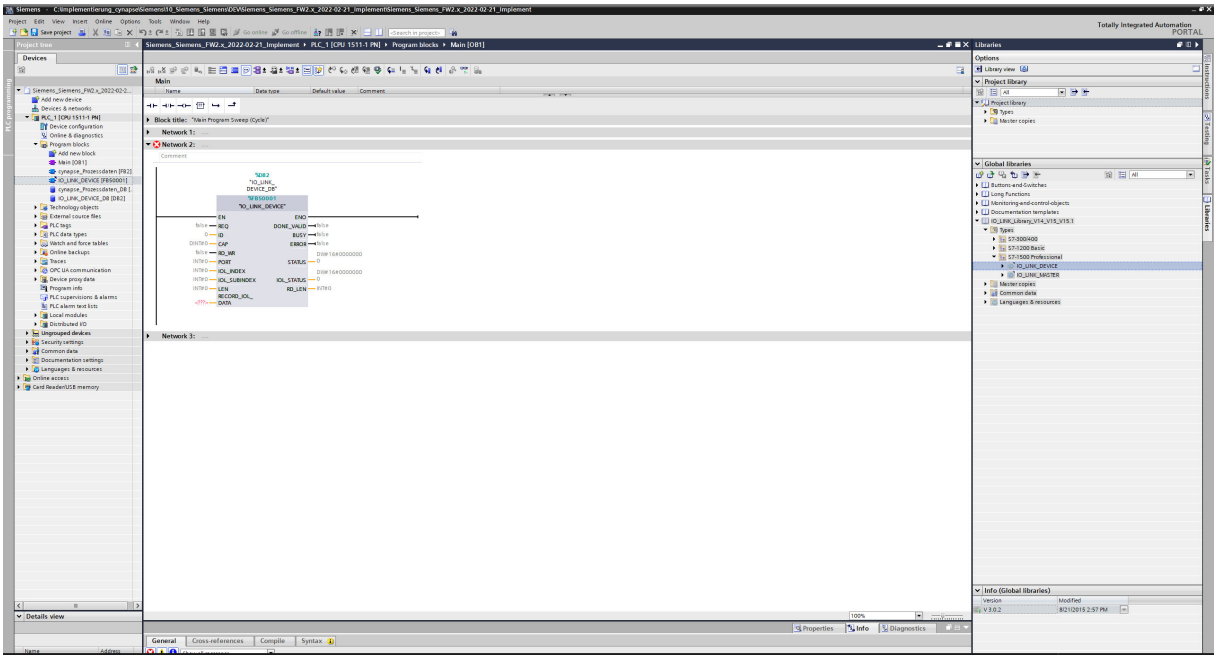
3. Click “Open”.
4. The library appears under “Global libraries” and can be opened.



5. Drag and drop the IO_LINK_DEVICE block into program blocks.

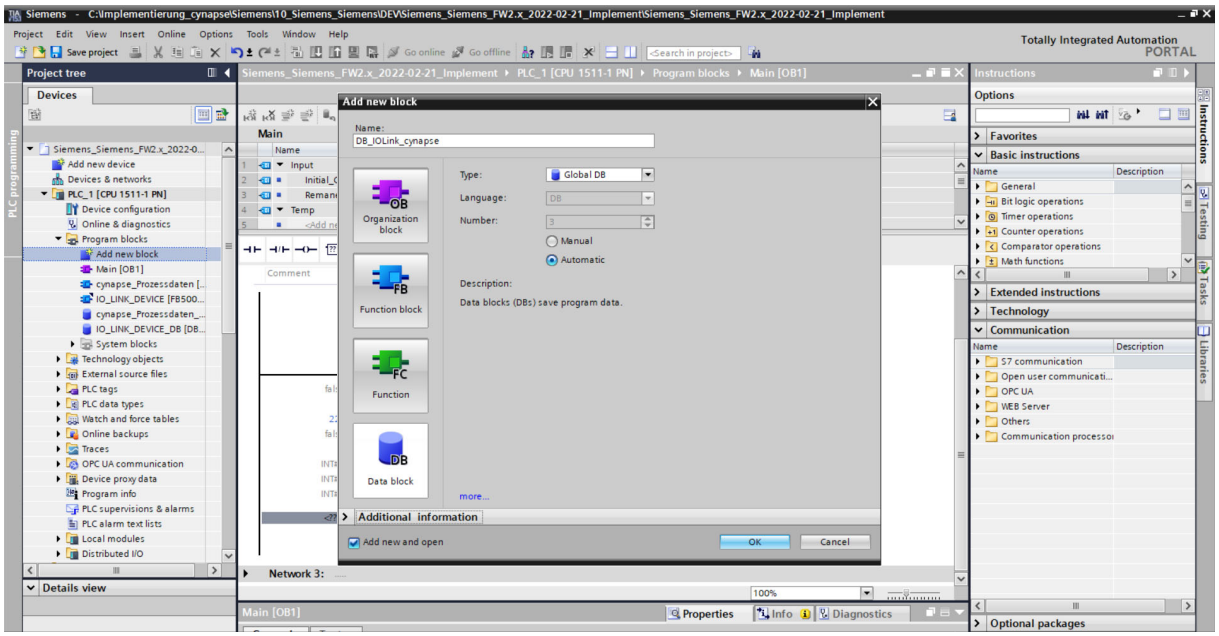


6. Open the main block with a double click.
7. Drag and drop the IO_LINK_DEVICE block into the network.
8. Have the corresponding DB created by the TIA Portal statement.

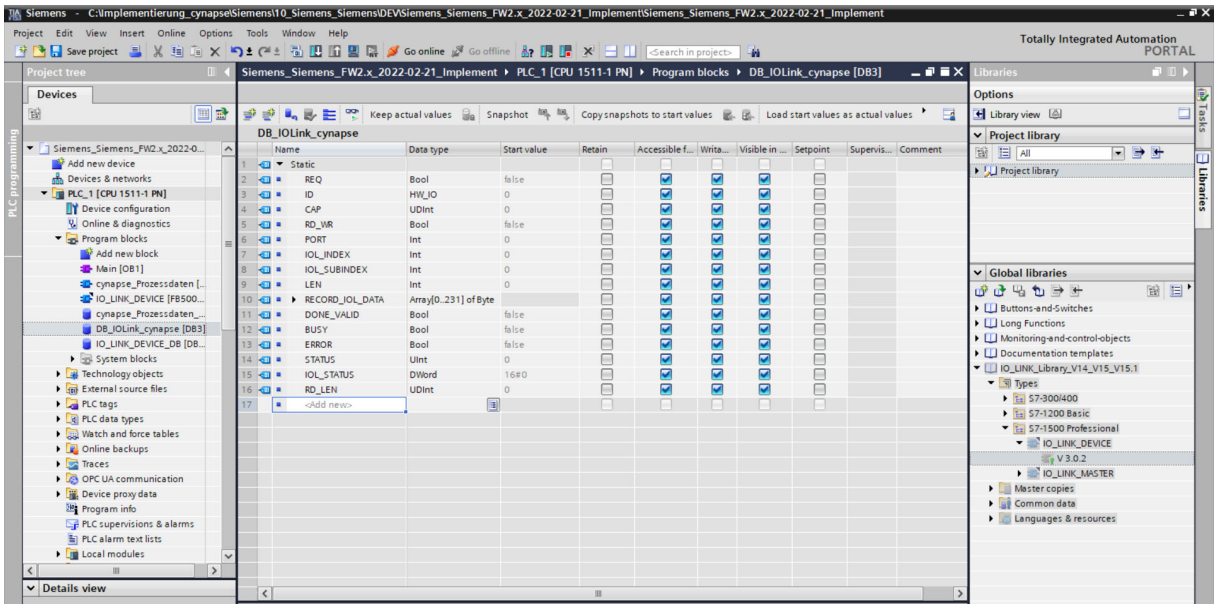


9. Create the data block DB for inputs and outputs of the function block:

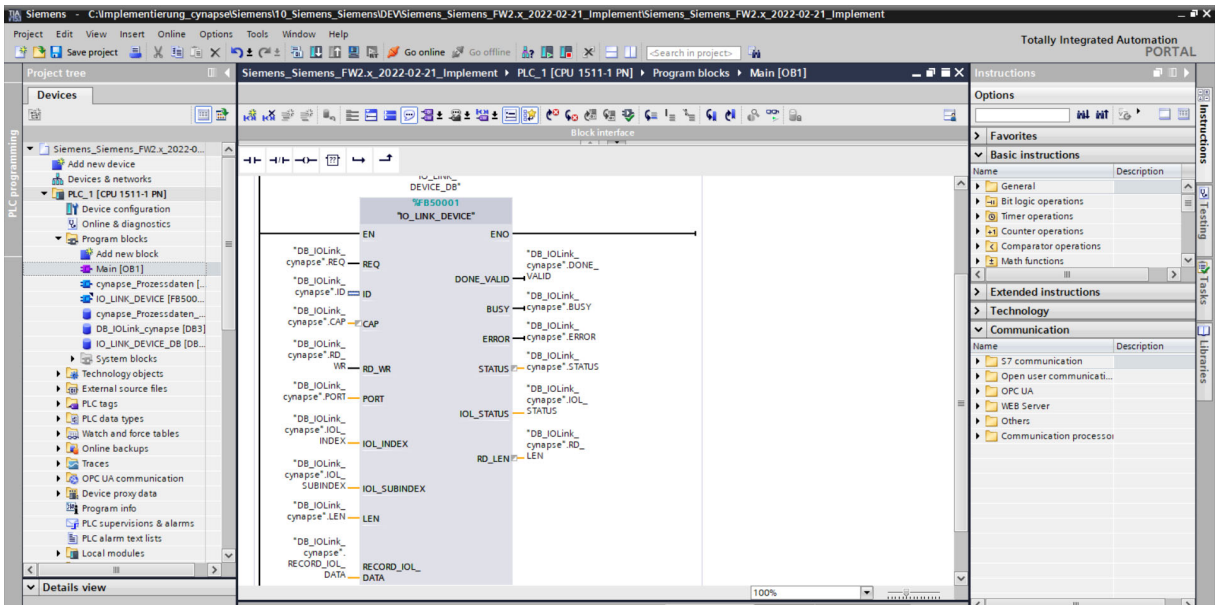
- Under “Program blocks”; click on “Add new block”.
- Select and name DB.



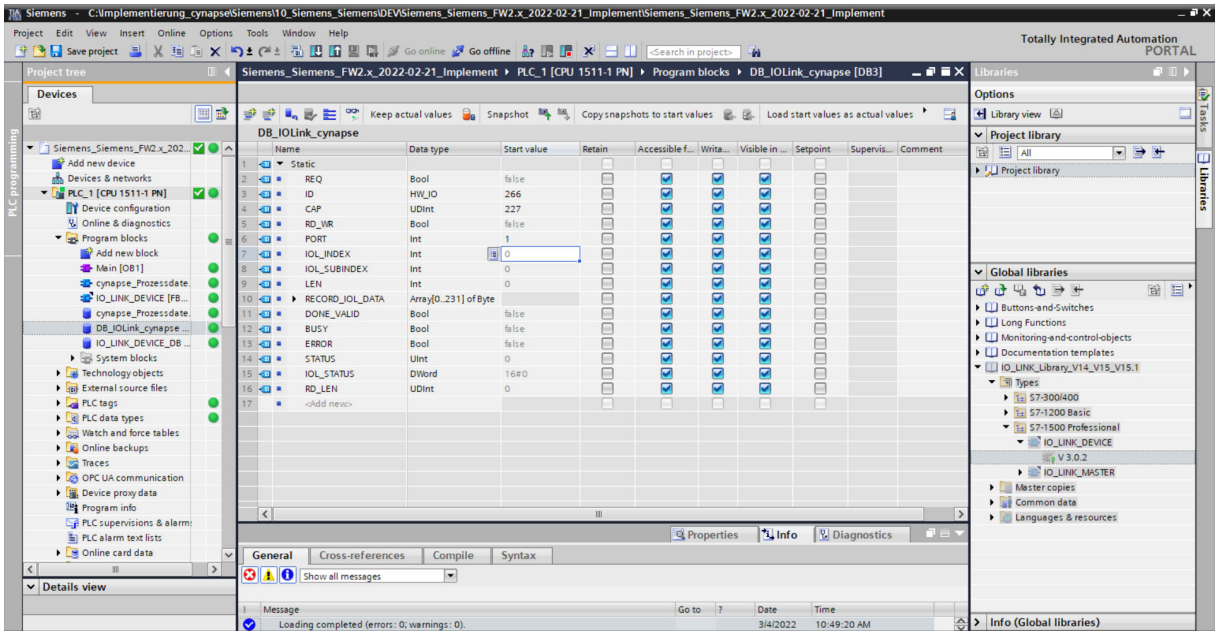
- Click “OK”.
- Open DB by double-clicking on it.
- Create variables in the required data types according to the block description of Siemens in the DB.



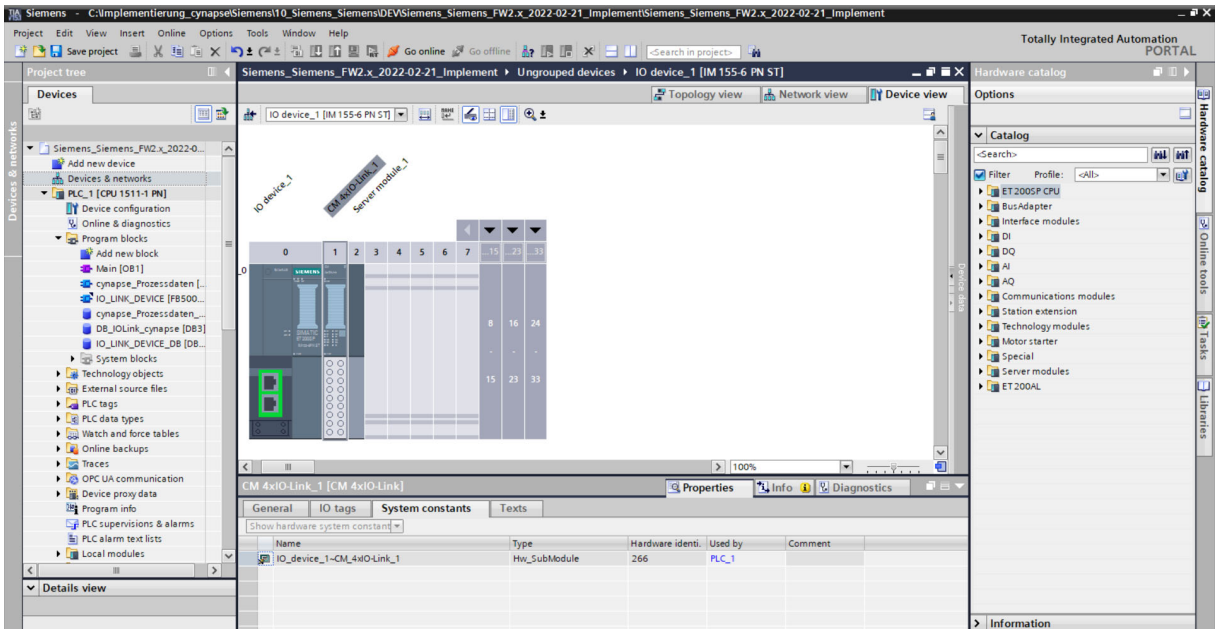
10. Link the inputs and outputs of the function block in the main block to the created variables in the data block.



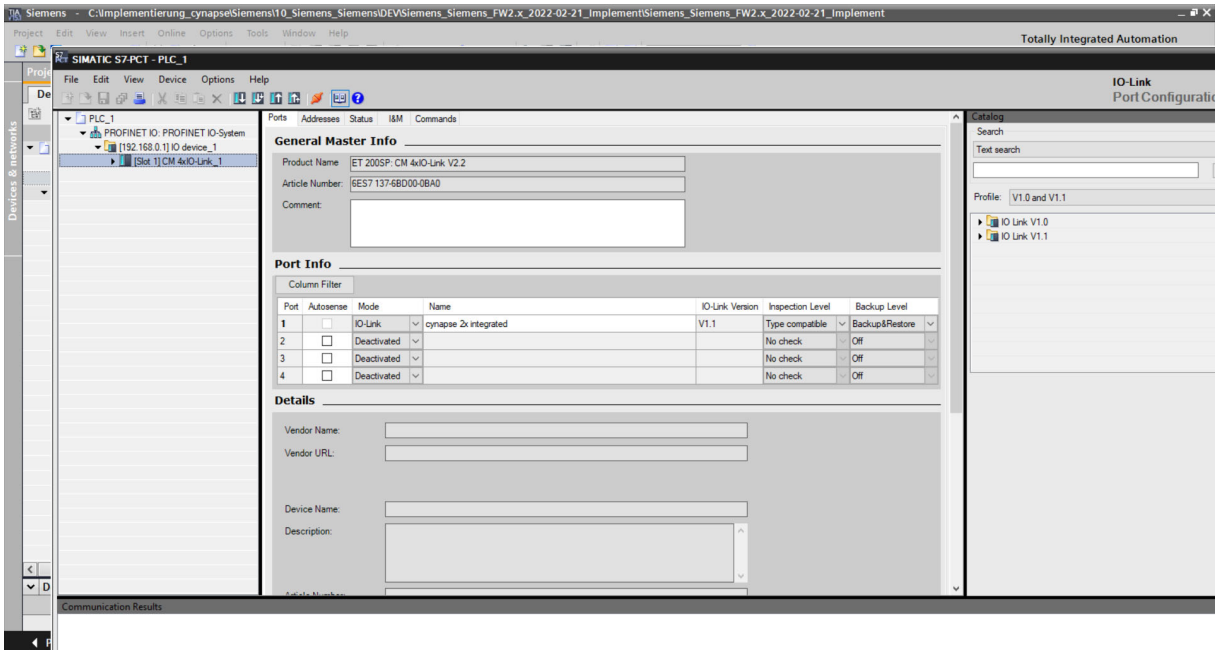
11. Open the created DB by double-clicking.
12. Set project-specific input parameters:



- ID: IO-Link communication module hardware ID: This information can be found in the system constants in the hardware view = 266

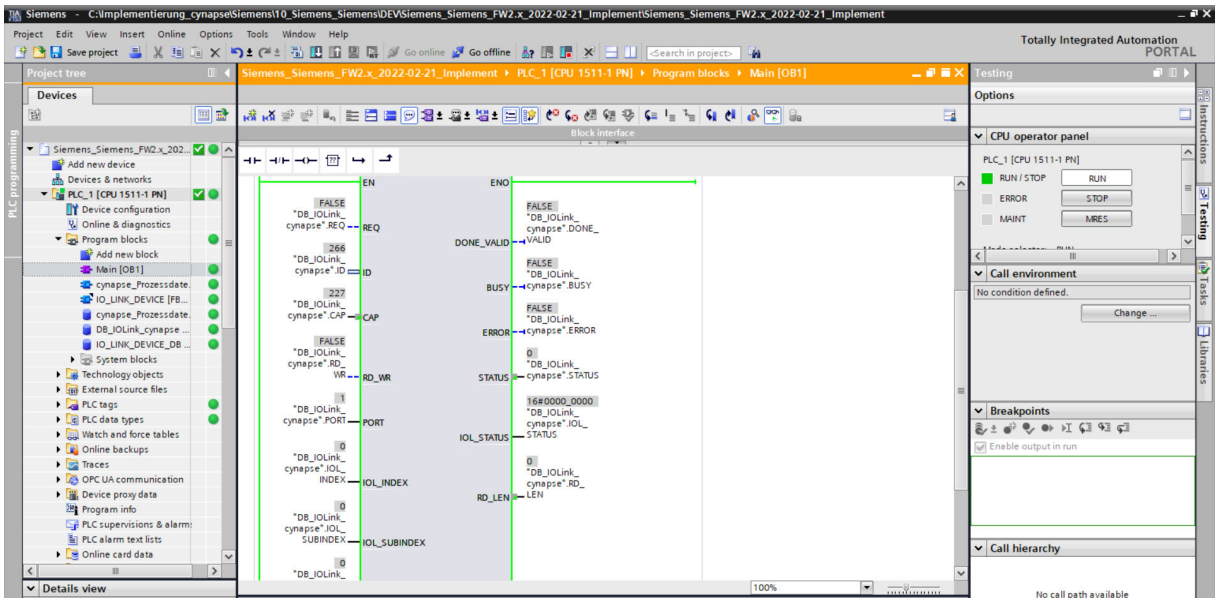


- CAP: Client Access Point: This information can be found in the Siemens documentation = 227
- PORT: Port number on which the IO-Link device is operated: This information can be found in the Port Configuration Tool = 1



Result

- The functional module is integrated into the main module.
- All inputs and outputs are linked to the created data block.
- Project-specific input quantities were determined and set as starting values in the DB.



5.3 Reading Parameters

Requirement

- ➲ The project is open and the function module IO_Link Device has been integrated into the program as described in chapter 5.2 “Integrating Siemens block for parameter reading/writing into program”.
- ➲ The project was successfully loaded into the hardware and online access is possible.
- ➲ The index and subindex information of the desired parameter were determined. The general indices are given in the IO-Link specification. The cynapse®-specific indices can be found in the operating manual.

ⓘ **Note:** A duplicate request to a device via the IO_LINK_DEVICE module is not possible. Since parameters are read, parameters are written, events are read out and blob data are accessed on this block, these are to be locked against one another.

Introduction

In the following, you will use the integrated function module “FBIoLinkDevice” to read the currently issued process data format from cynapse® using the parameter Settings.

Here is some information about the module:

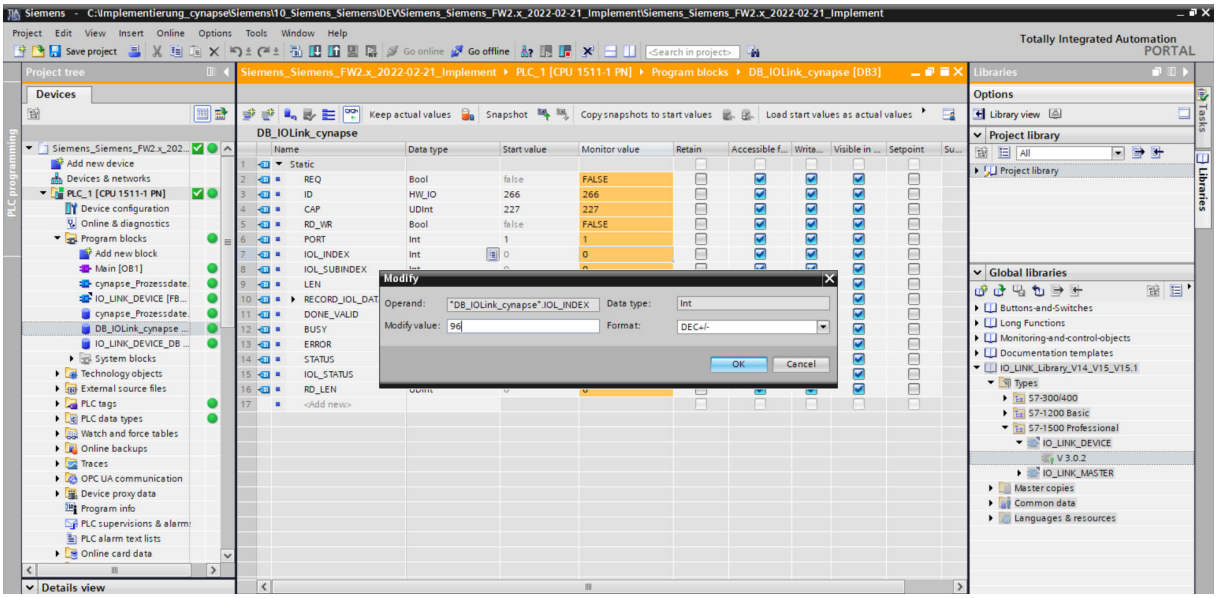
- The data transmission takes place in the form of raw data (ARRAY of byte)
- If “RD_WR” = FALSE, data is read out and output to “RECORD_IOL_DATA”.
- As long as no valid response data have been received, this is signaled via the output “BUSY” = TRUE.
- The value TRUE of the output “DONE_VALID” indicates that the transfer was successful. In the case of a read job, the data are now consistently present at the input/output “REDORD_IOL_DATA” and the output “RD_LEN” indicates the length of the received data.
- The value TRUE of the output “ERROR” indicates that an error has occurred. As long as the input REQ = TRUE, the output parameters retain their value. If the input REQ = FALSE before the processing of the FB is completed, the values of the output parameters are held for only one cycle after the processing of the order.

Procedure

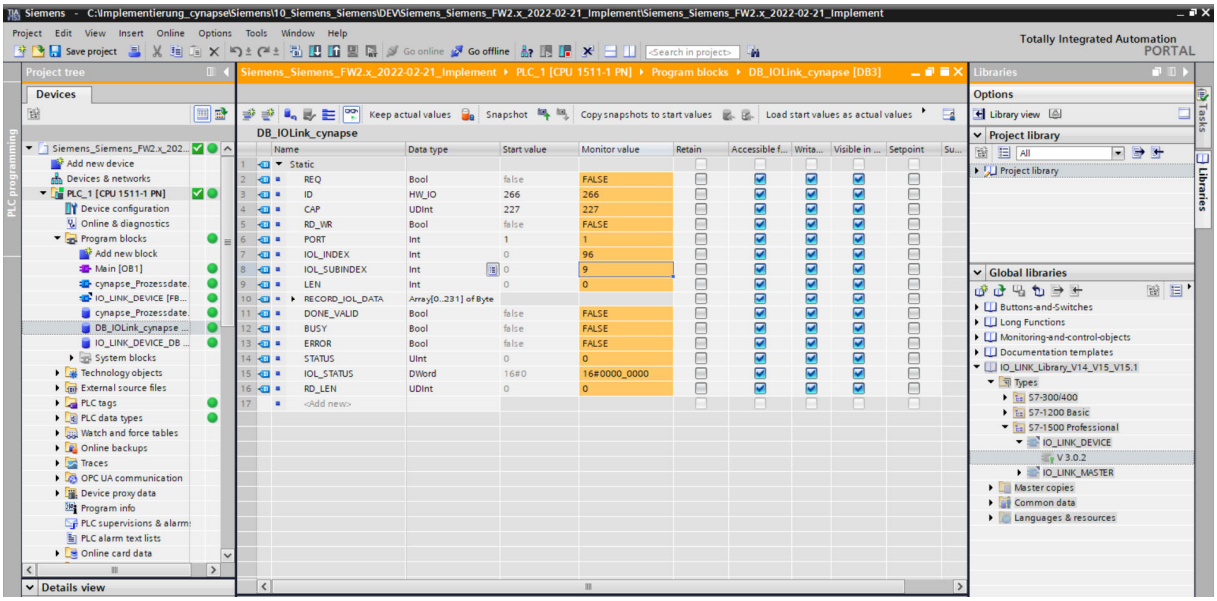
1. Open the data block with input/output variables of the block by double-clicking.
2. Press the “Go online” button. 

3. Start online monitoring. 
4. Double-click on “Monitor value” of the input you want to change.
5. Enter the index.

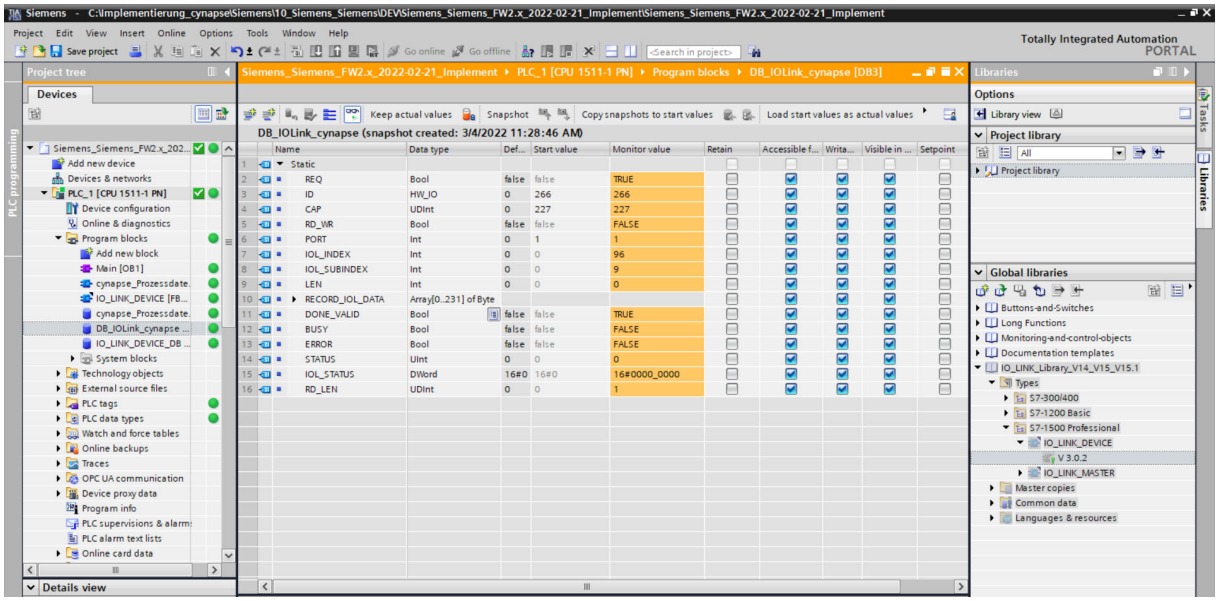
6. Confirm with “OK”.



7. Enter the subindex according to the same scheme.



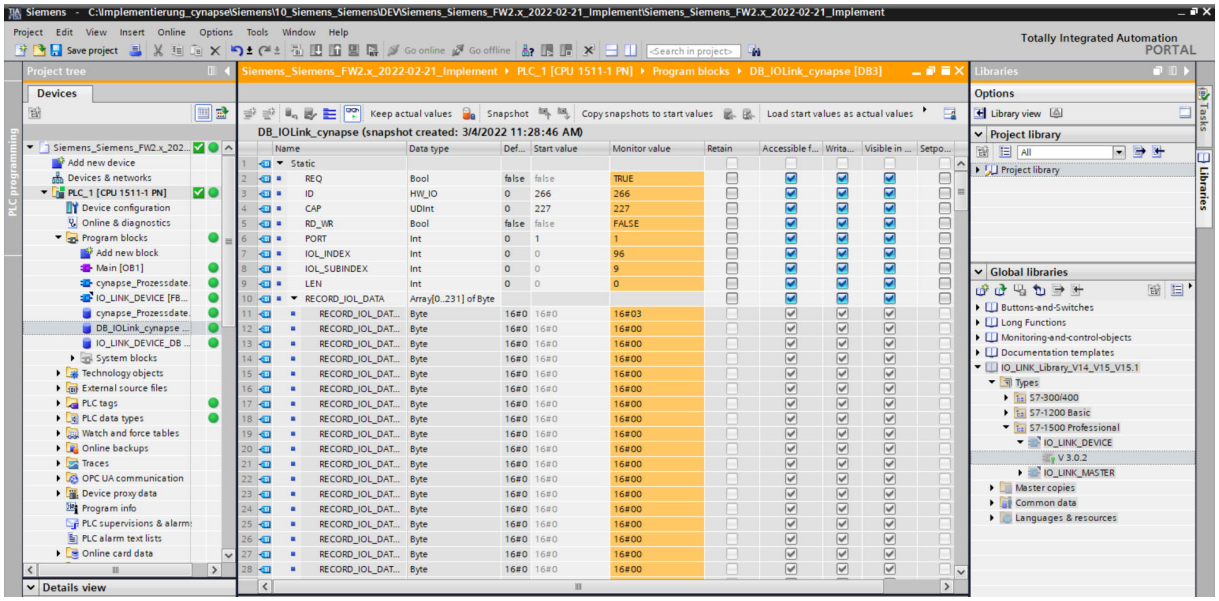
8. Set input REQ from FALSE to TRUE by double-clicking on “Monitor value”.



9. DONE_VALID TRUE indicates successful readout.
10. RD_LEN displays the length of the parameter read.
11. Expand RECORD_IOL_DATA.

Result

➲ In the RECORD_IOL_DATA array, the parameters read are displayed in coded form.



This example specifies the currently output process data format.

5.4 Writing parameters

Requirement

- ➲ The project is open and the function module IO_Link Device has been integrated into the program as described in chapter 5.2 “Integrating Siemens block for parameter reading/writing into program”.
- ➲ The project was successfully loaded into the hardware and online access is possible.
- ➲ The information about the index, subindex and length of the desired parameter was determined. The general indices are given in the IO-Link specification.
  The cynapse®-specific indices can be found in the operating manual.

ⓘ **Note:** A duplicate request to a device via the IO_LINK_DEVICE module is not possible. Since parameters are read, parameters are written, events are read out and blob data are accessed on this block, these are to be locked against one another.

ⓘ **Note:** If the length of the parameter to be written is greater than 1 byte, it must be converted into a byte array. For example, parameters of the float type must be converted into a 4-byte array using the “REAL_TO_DWORD” command.

Introduction

In the following, you change the currently output process data format using the parameter Settings of cynapse® for example by using the integrated function module “FBIoLinkDevice”. A parameter is written for this.

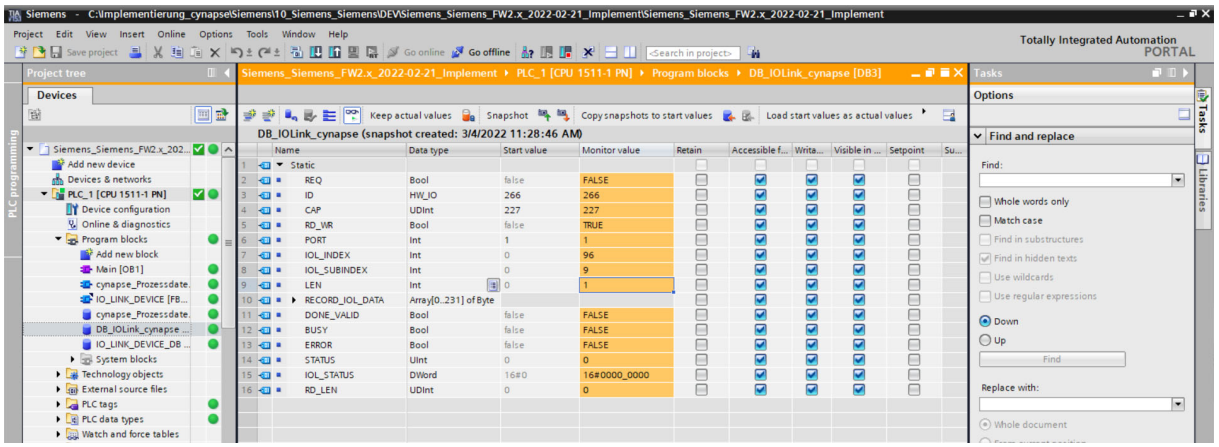
Here is some information about the module:

- The data transmission takes place in the form of raw data (ARRAY of byte)
- If “RD_WR” = TRUE, the data from “RECORD_IOL_DATA” is written in cynapse®.
- For a write job, the length of the parameter to send must be specified under LEN.
- As long as no valid response data have been received, this is signaled via the output “BUSY” = TRUE.
- The value TRUE of the output “DONE_VALID” indicates that the transfer was successful.
- The value TRUE of the output “ERROR” indicates that an error has occurred. As long as the input REQ = TRUE, the output parameters retain their value. If the input REQ = FALSE before the processing of the FB is completed, the values of the output parameters are held for only one cycle after the processing of the order.

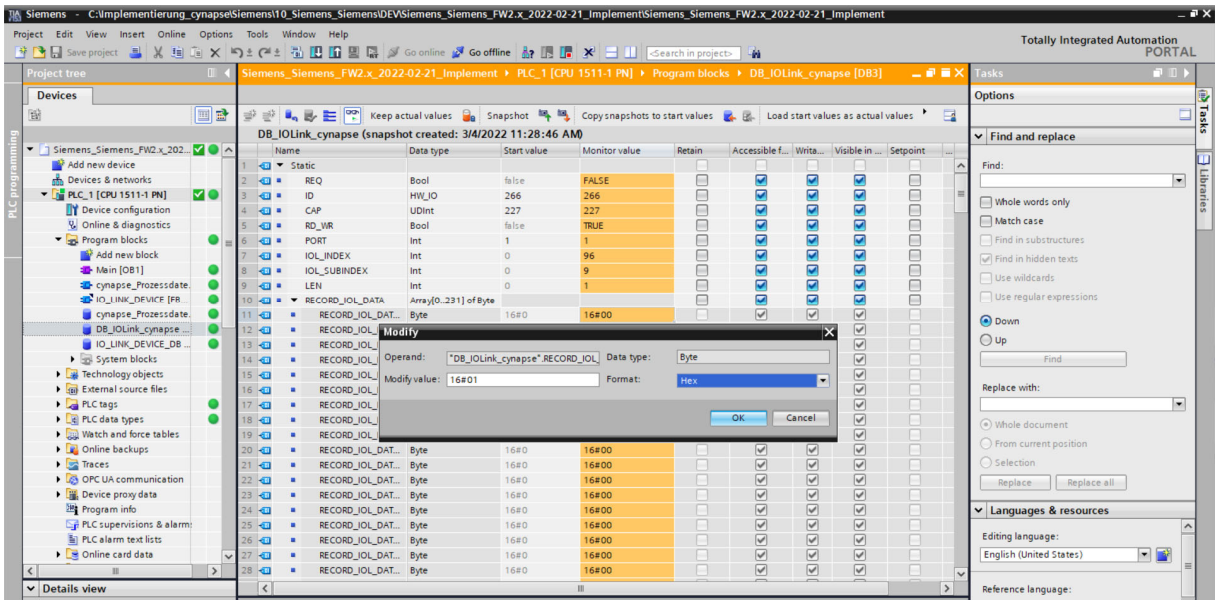
Procedure

1. Open the data block with input/output variables of the block by double-clicking.
2. Press the “Go online” button. 

3. Start online monitoring. 
4. Set the value for the input variable RD_WR by double-clicking on the variable in the orange “Monitor value” column to TRUE.
5. Enter the length of the parameter to write in LEN.



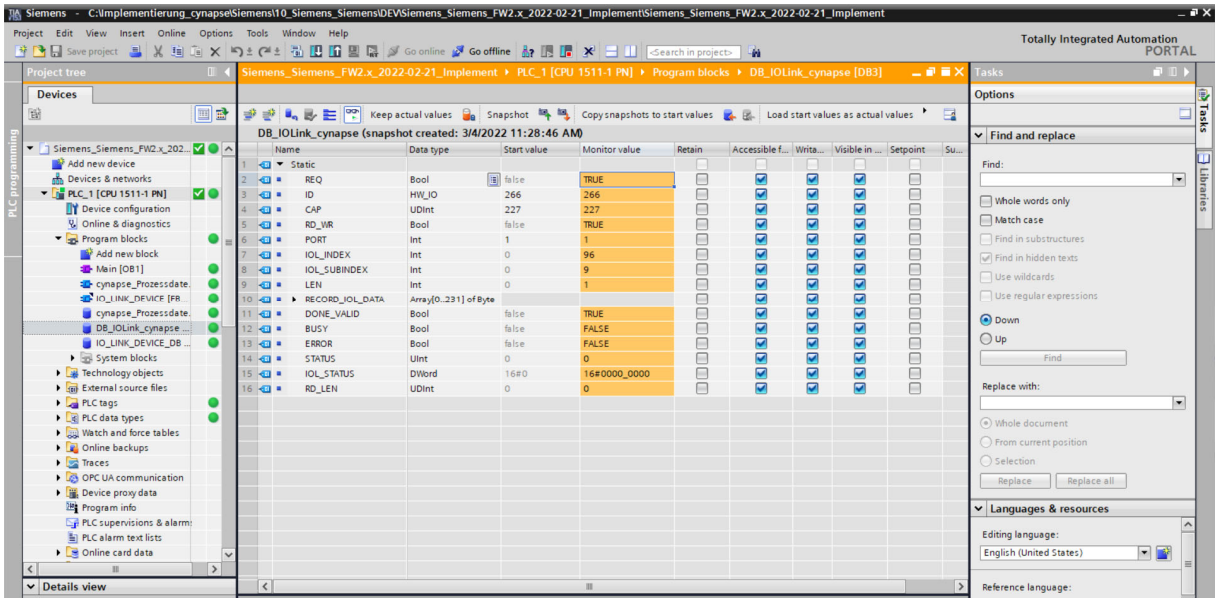
6. In Array RECORD_IOL_DATA, enter the desired process data format = 1 in the first byte.



7. Set the REQ input from FALSE to TRUE by double-clicking on “Monitor value”.

Result

➲ DONE_VALID TRUE indicates successful writing of the parameter.



The change can be checked by reading the parameter as described in chapter 5.3 “Reading Parameters”.

6 Events

6.1 Definition

Events is the notification of an incorrect operating state of the IO-Link device. Examples of this are too high an operating temperature, vibrations or a detected defect on the device. In this case, warnings or errors in the form of events only occur when threshold values are exceeded or undershot and are automatically reset. cynapse® distinguishes between limit values defined by WITTENSTEIN and user-defined limit values. The latter can be changed via Write parameters.

For details on parameters for custom thresholds and error-specific codes, see the cynapse® operating manual.

6.2 Reading events

Requirement

➲ You have obtained a sample project for reading out events from the following source: cybertronic-support@wittenstein.de

➲ The hardware build-up with up to two cynapse® is present without errors, with the order of the ports and the devices coinciding (one cynapse® on port 1, second cynapse® on port 2 of the master)

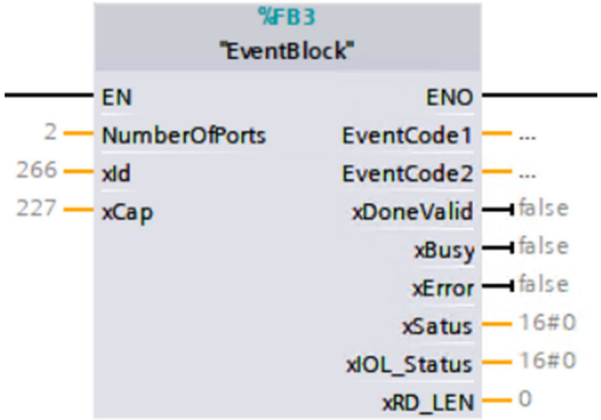
ⓘ **Note:** The function module described below is programmed for reading out events of up to two cynapse®. If you are running more than these two IO_Link devices, please contact: cybertronic-support@wittenstein.de

In order to read out events, a general event release must take place in cynapse®. This release is given in the parameter Index = 96, Subindex = 1. In addition, events are enabled for reporting via the subindexes 2, 3, 5 and 7. For more information, see the cynapse® operating manual.

ⓘ **Note:** A duplicate request to a device via the IO_LINK_DEVICE module is not possible. Since parameters are read, parameters are written, events are read out and blob data are accessed on this block, these are to be locked against one another.

Introduction

In the following, you will read upcoming events using a function module from a sample project. By shaking, several events are provoked.

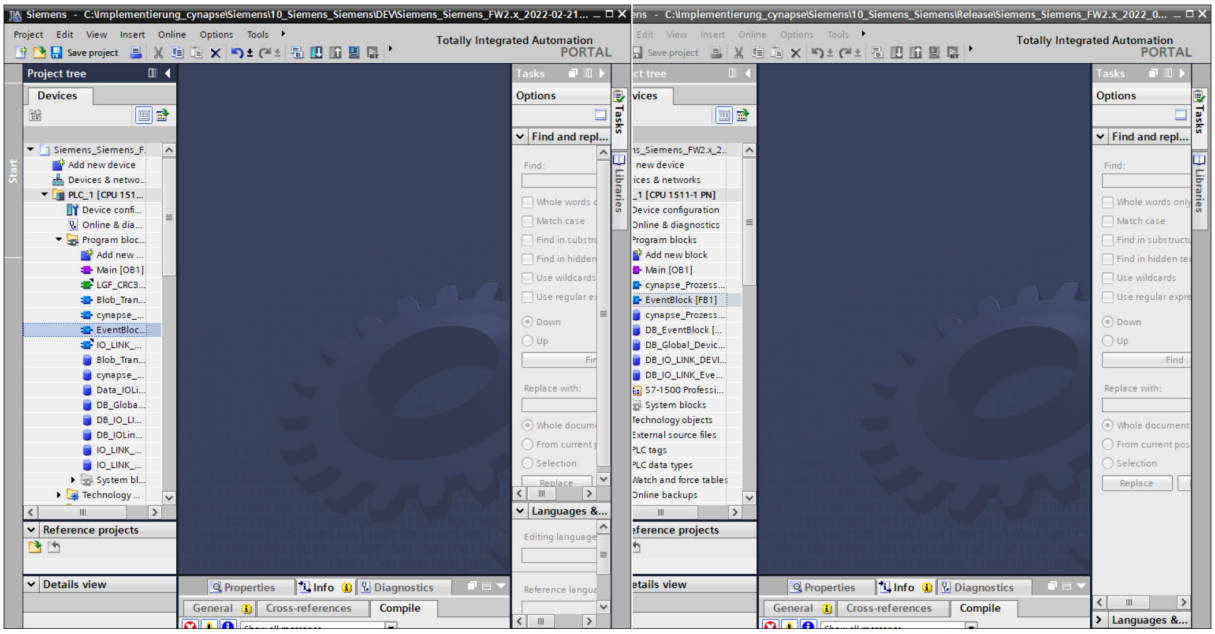


Input/output	Data type	Function
NumberOfPorts	Integer	Number of attached devices
xid	HW_IO	IO-Link Communication Module Hardware ID Executed input of the IO_LINK_DEVICE module
xCap	DInteger	Client Access Point Executed input of the IO_LINK_DEVICE module
EventCode1, EventCode2	Array of bytes	Event code of the respective port read out
xDoneValid	Bool	Executed output of the IO_LINK_DEVICE module
xBusy	Bool	Executed output of the IO_LINK_DEVICE module
xError	Bool	Executed output of the IO_LINK_DEVICE module
xStatus	DWord	Executed output of the IO_LINK_DEVICE module
xIOL_Status	DWord	Executed output of the IO_LINK_DEVICE module
xRD_LEN	Integer	Executed output of the IO_LINK_DEVICE module

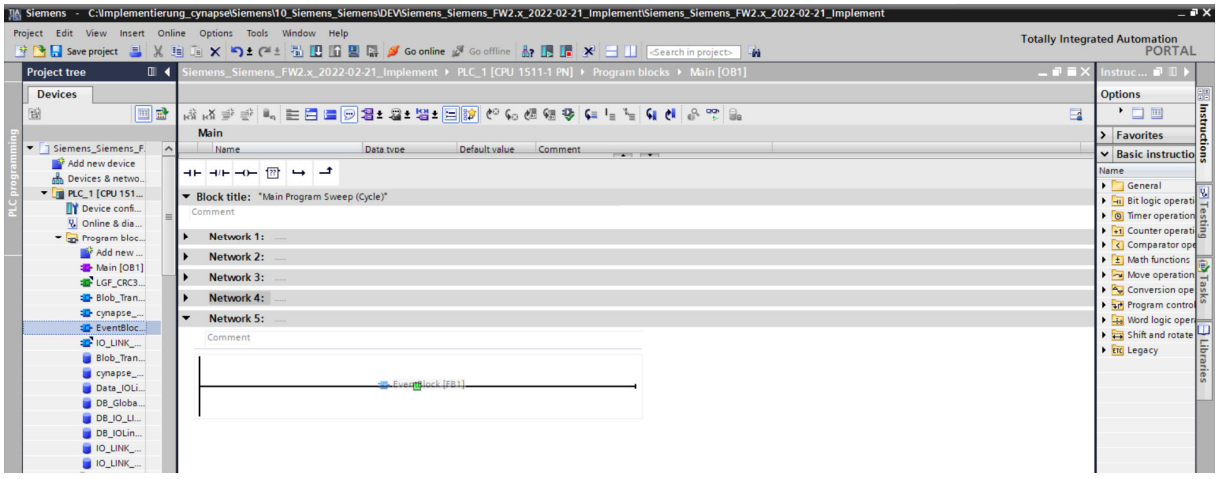
Tbl - 2

Procedure

1. Open the resulting sample project.
2. In parallel, open the project in which you want to monitor events.
3. Highlight the EventBlock function block.
4. Drag and drop this into your project under “Program blocks”.

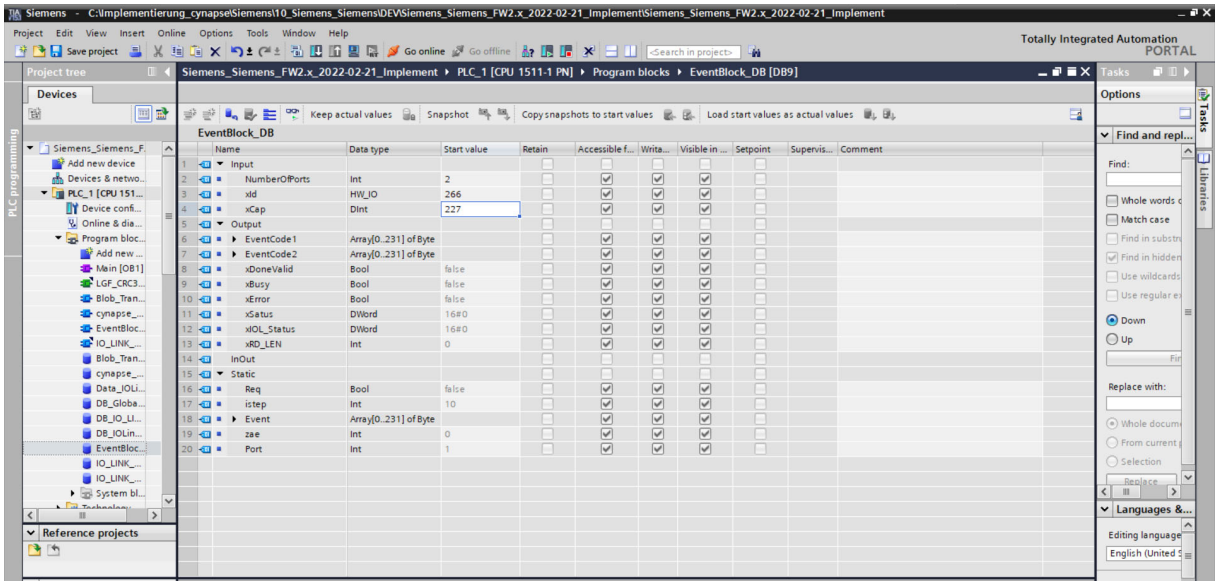


5. Drag and drop the FB into the main block.

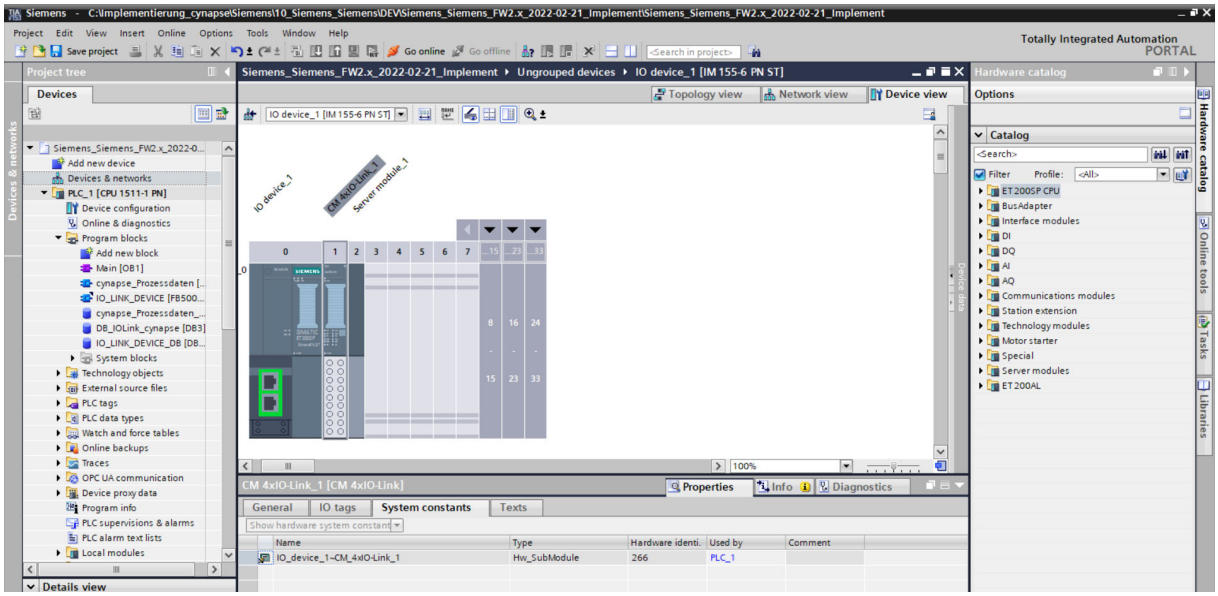


6. Create the DB by clicking on “OK”.
7. The block is contained in the network of the main block.
8. Open the data block with input/output variables of the block by double-clicking.

9. Set project-specific input parameters:



- NumberOfPorts: Number of attached devices
- ID: IO-Link communication module hardware ID: This information can be found in the system constants in the hardware view = 266

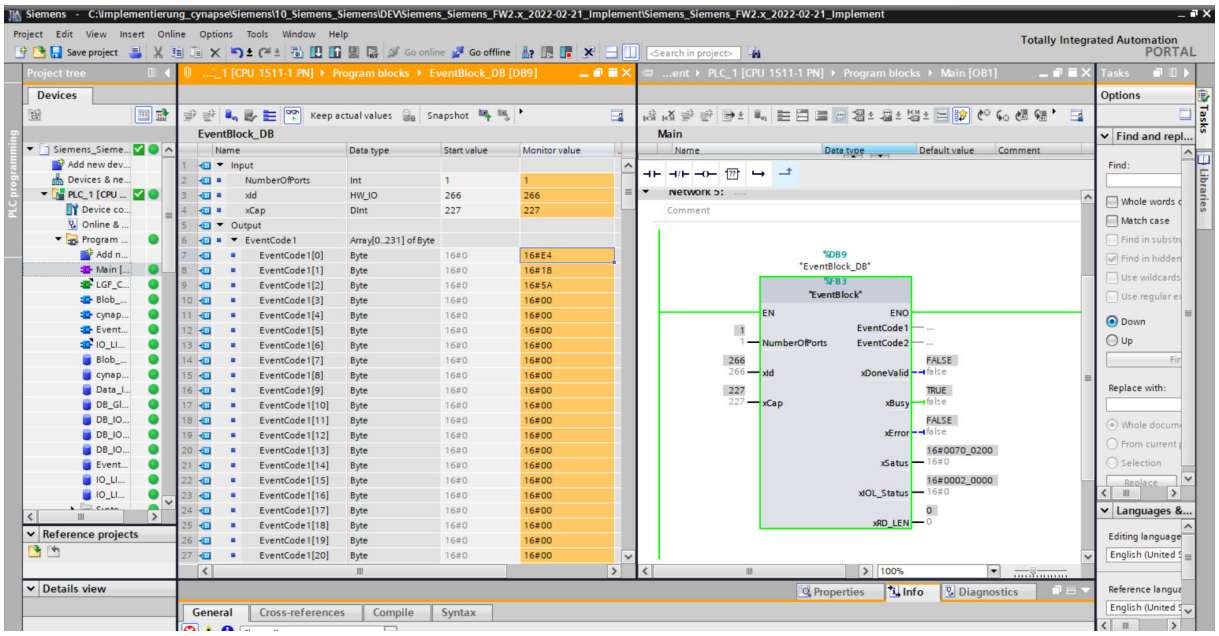


- CAP: Client Access Point: This information can be found in the Siemens documentation = 227

10. Compile the program.
11. Run the program on the hardware.
12. Press the “Go online” button. Go online
13. Start online monitoring.

Result

➲ The upcoming event codes can be found in the EventCode arrays. If several events are pending, they are displayed successively in packets of 3 bytes as a list in the RECORD_EVENT array.



A table from the IO-Link specification explains this.

Table B.14 – DetailedDeviceStatus (Index 0x0025)

Sub-index	Object name	Data Type	Comment
1	Error_Warning_1	3 octets	All octets 0x00: no Error/ Warning Octet 1: EventQualifier Octet 2,3: EventCode
2	Error_Warning_2	3 octets	
3	Error_Warning_3	3 octets	
4	Error_Warning_4	3 octets	
…			
n	Error_Warning_n	3 octets	

Byte 1 always has the value 16#E4 and initiates the event code.

The event code in byte 2 and byte 3 can now be decoded with the cynapse® operating manual and used further.

In this example, the following event is pending:

Byte 1 EventQualifier	Byte 2 EventCode	Byte 3 EventCode	Translation using the operating manual
16#E4	16#18	16#5A	The user's upper temperature threshold has been exceeded

Tbl - 3

7 Blob data

7.1 Definition

IO-Link defines the transfer of large amounts of data (**b**inary **l**arge **ob**ject) by the BLOB transfer profile. In this case, the type of the data block to be transmitted is identified via the BLOB_ID between 1 and 32767. The sign of the ID indicates the direction of the transfer; a positive sign indicates the data flow direction from master to device, a negative sign indicates data flow from device to master.

cynapse® offers various data packets transported by BLOB transfer. For more information about the BLOB_ID and the decoding of the data packets read out, see the cynapse® operating manual.

7.2 Reading blob data using the “Blob_Transfer” FB

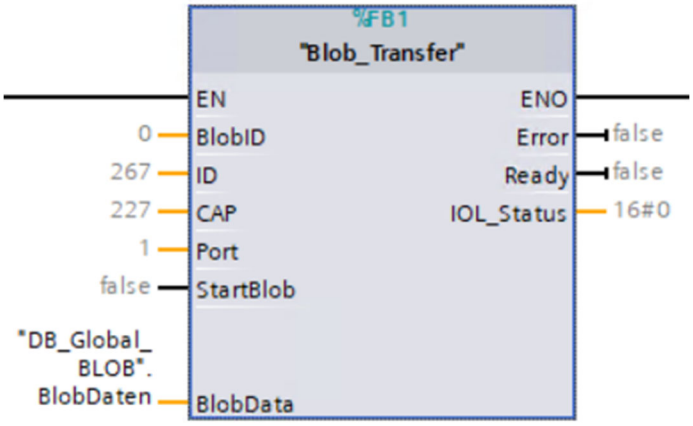
Requirement

➲ You have obtained a sample project for blob transfer from the following source: cybertronic-support@wittenstein.de

ⓘ **Note:** A duplicate request to a device via the IO_LINK_DEVICE module is not possible. Since parameters are read, parameters are written, events are read out and blob data are accessed on this block, these are to be locked against one another.

Introduction

Below, you will read blob data from a sample project using a function block. The latter takes over the specification-compliant sequence of the blob transfer. For more information, see the IO-Link specification.



Input/output	Data type	Function
BlobID	Integer	ID of the data block to be transmitted
ID	HW_IO	IO-Link Communication Module Hardware ID
CAP	Integer	Client Access Point
port	Integer	Port number where the IO-Link device is operated
StartBlob	Bool	Positive edge: Start blob transfer
BlobData	Array of bytes	Read data
Error	Bool	Error status (0: no error)
Ready	Bool	Ready status (1: successfully completed blob transfer)
IOL_Status	DWord	IO-Link error status of IO_LINK_DEVICE block

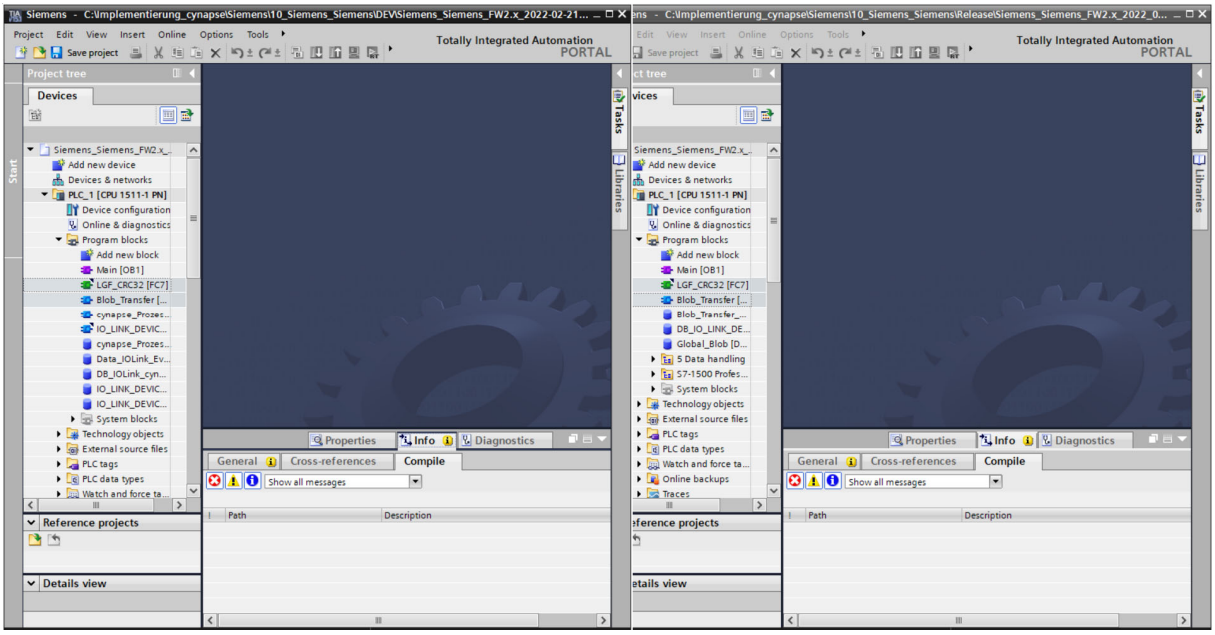
Tbl - 4

Procedure

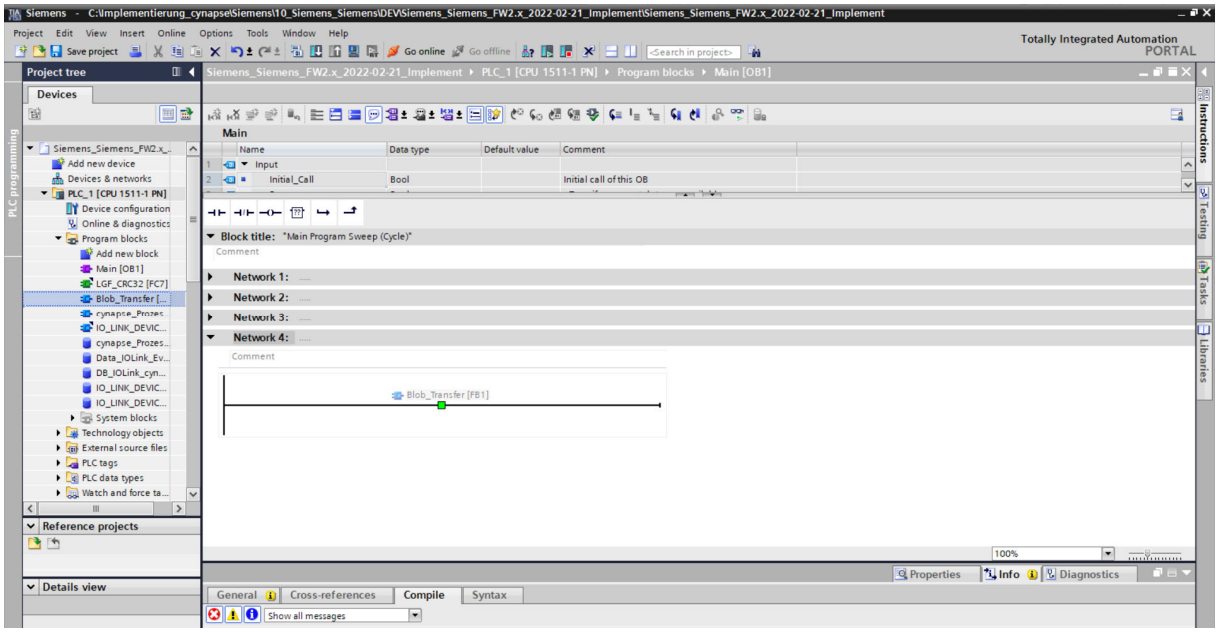
1. Create the DB by clicking on “OK”.
2. The block is contained in the network of the main block.
3. Open the resulting sample project.
4. In parallel, open the project in which you want to read blob data
5. Select the Blob-Transfer function block and the LGF_CRC32 function in the sample project.

ⓘ **Note:** Important: LGF_CRC32 from example project is slightly modified. If this is not used, the length of the array must be included in the original module.

6. Drag and drop this into your project under “Program blocks”.

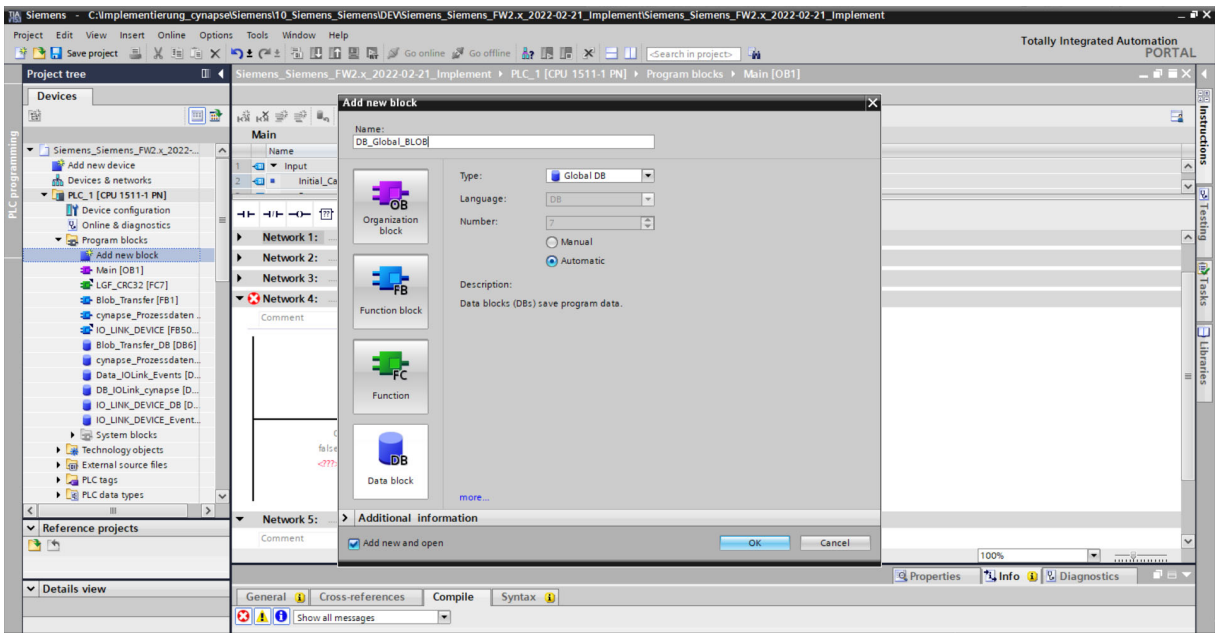


7. Drag and drop the FB into the main block.

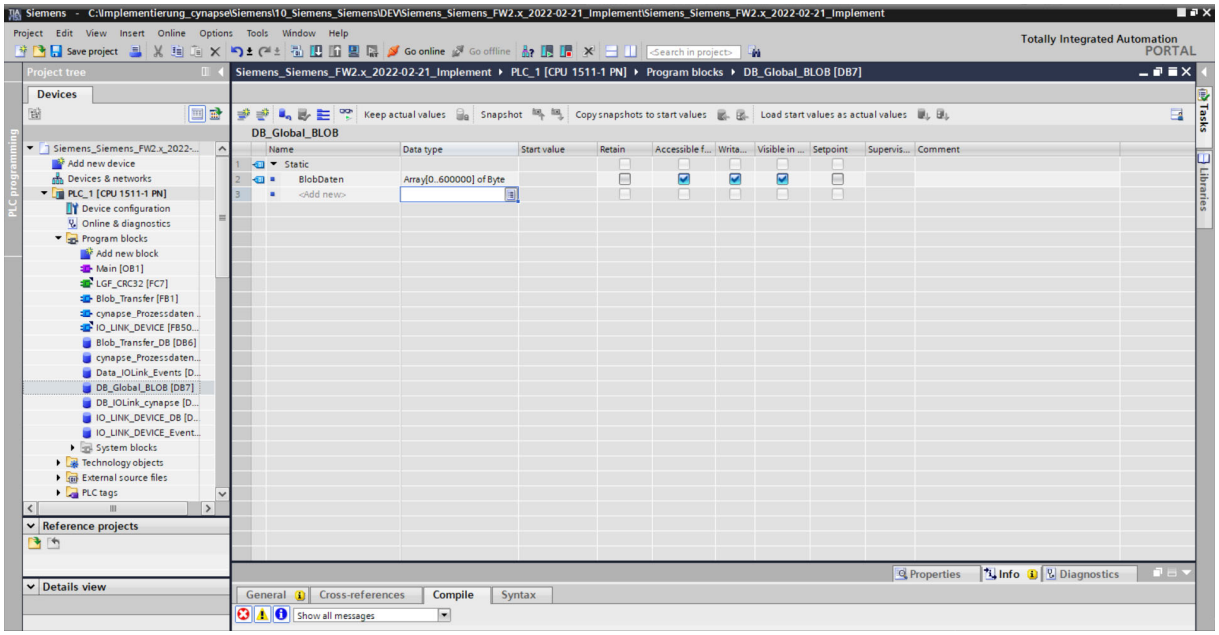


8. Create the DB by clicking on “OK”.
9. The block is contained in the network of the main block.
10. In order to be able to operate this, a range of values for the read-out BLOB data must be reserved in a global data module

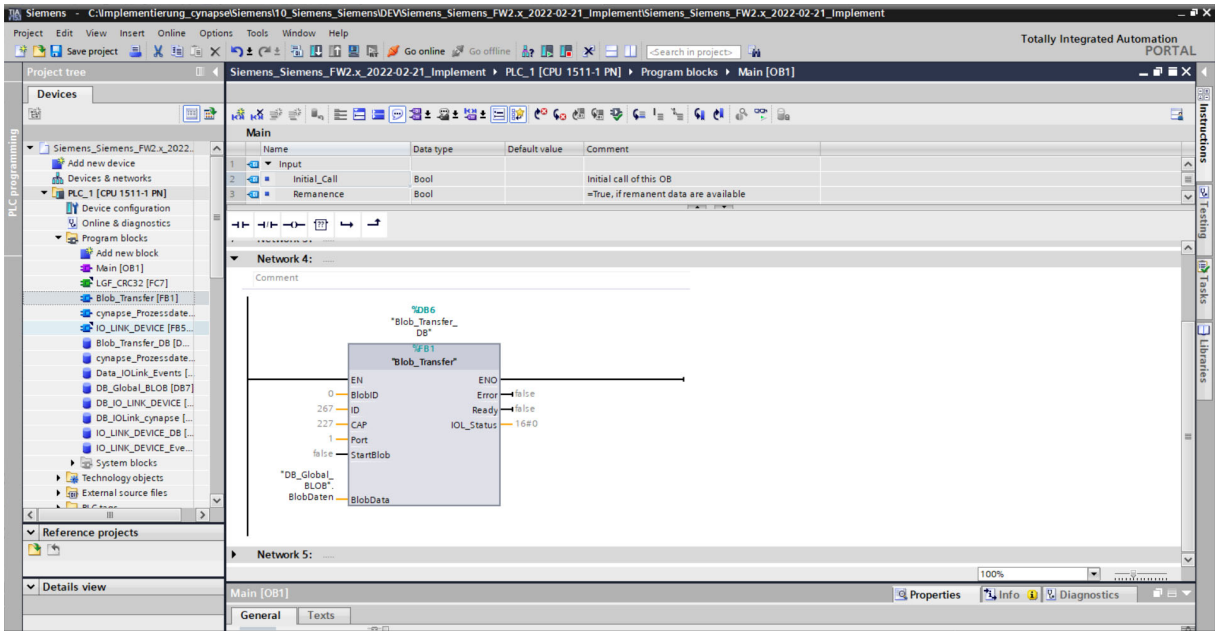
- Double click on “Add new block”.
- Create global data block.



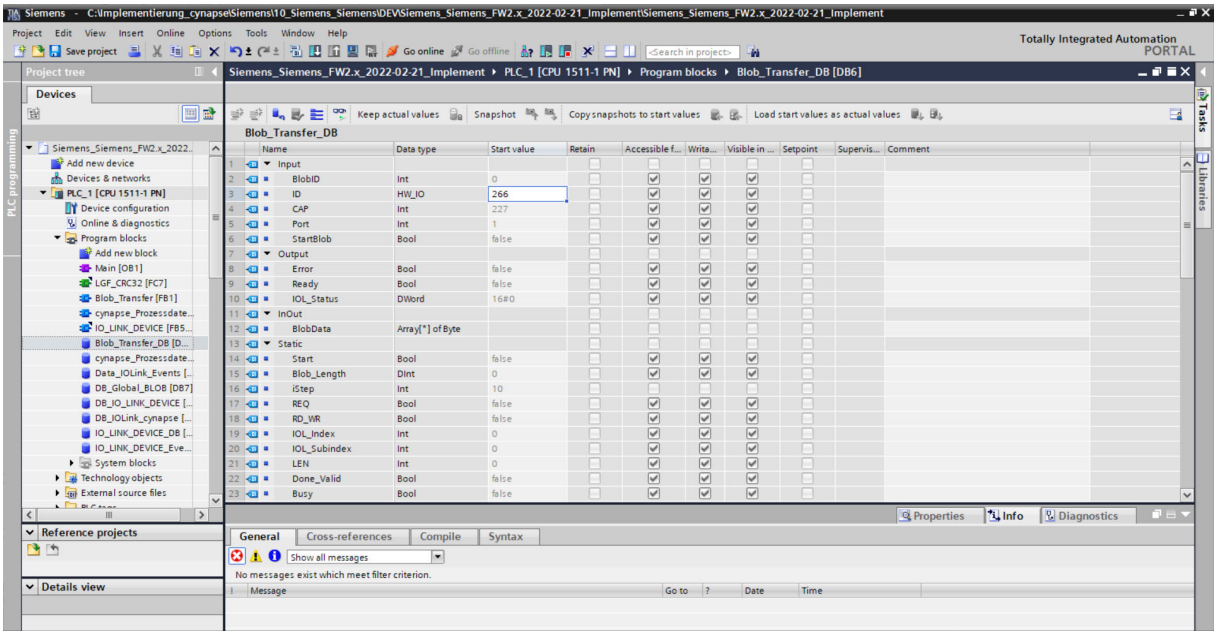
- Create an array in the size of 0-600000 bytes.



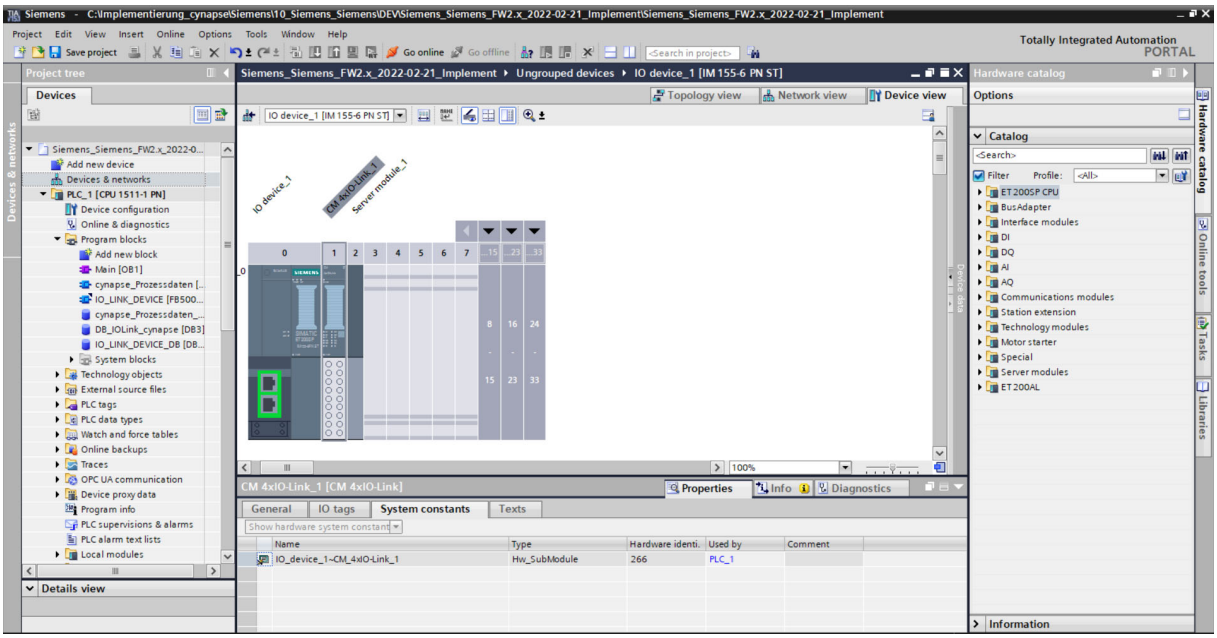
11. Link the block.



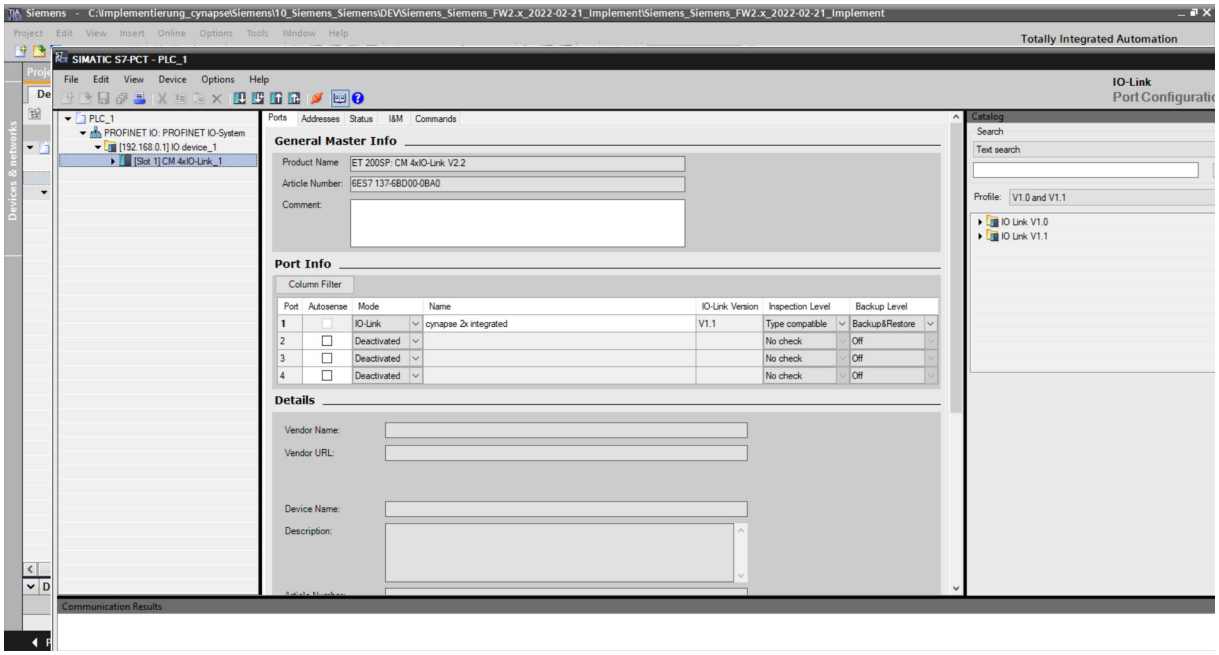
12. Set project-specific input parameters:



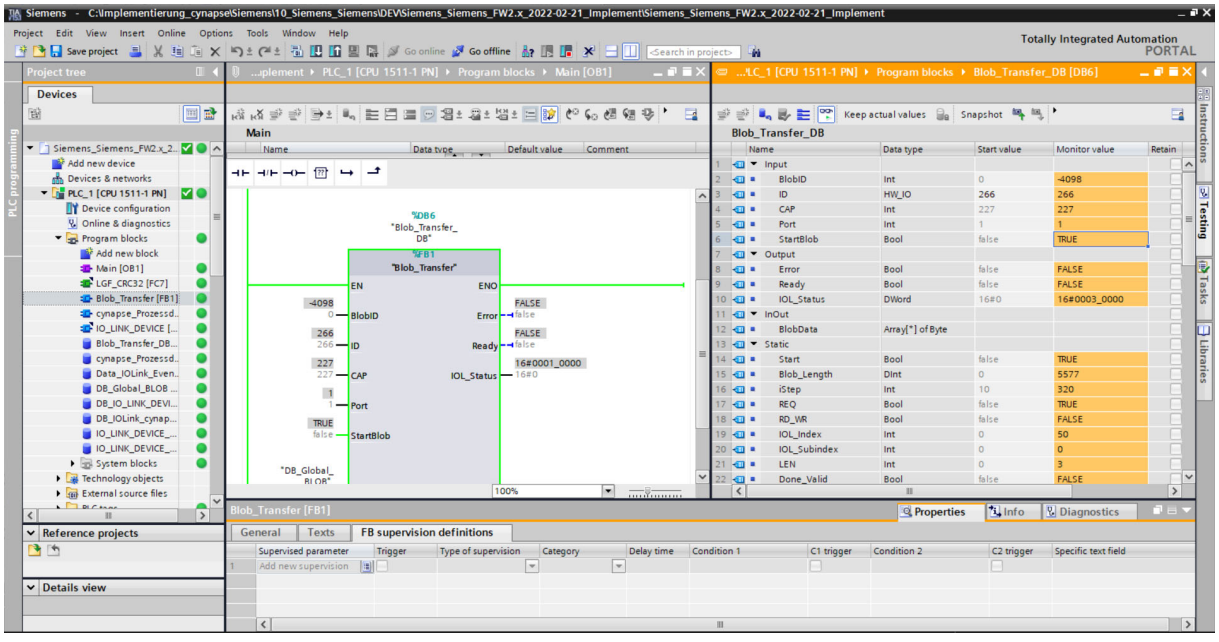
- ID: IO-Link communication module hardware ID: This information can be found in the system constants in the hardware view = 266



- CAP: Client Access Point: This information can be found in the Siemens documentation = 227
- PORT: Port number on which the IO-Link device is operated: This information can be found in the Port Configuration Tool = 1

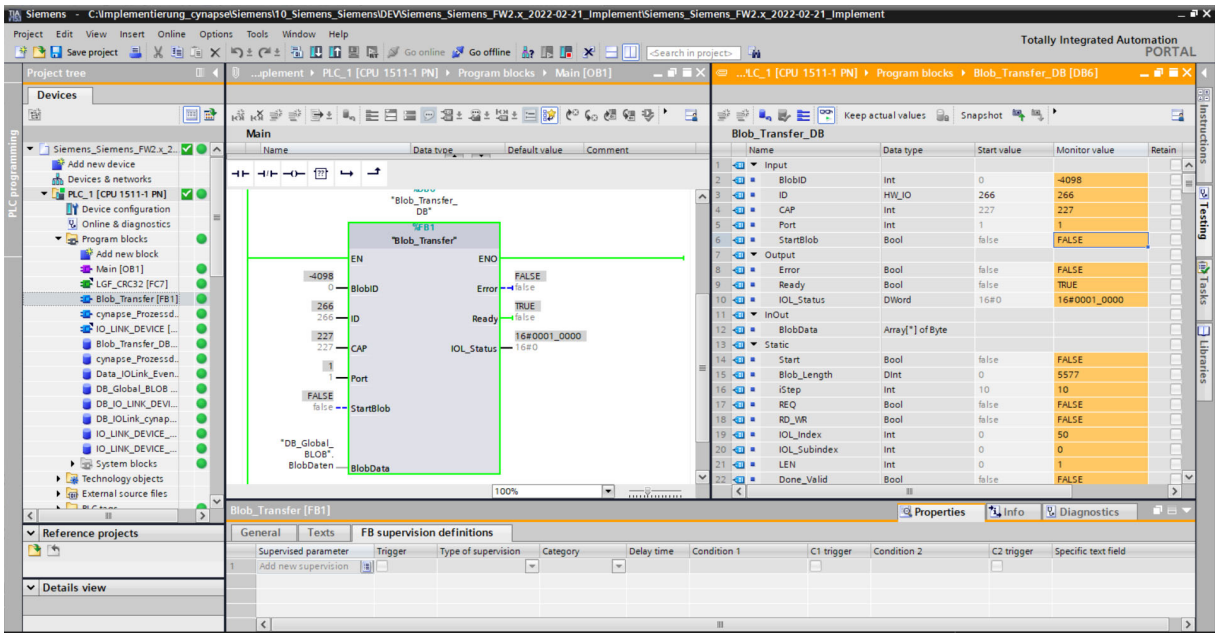


13. Download the project to load the hardware and go online.
14. Specify the BlobID.
15. Start the blob transfer with the start trigger.



Result

➲ By the output Ready = 1, the block indicates a successfully executed blob transfer. The data being read out is in the array associated with the BlobData output and can be used further.



8 Firmware update

8.1 Updating cynapse® firmware using PCT

Requirement

➲ You have obtained the latest update of cynapse® from the following source: cybertronic-support@wittenstein.de

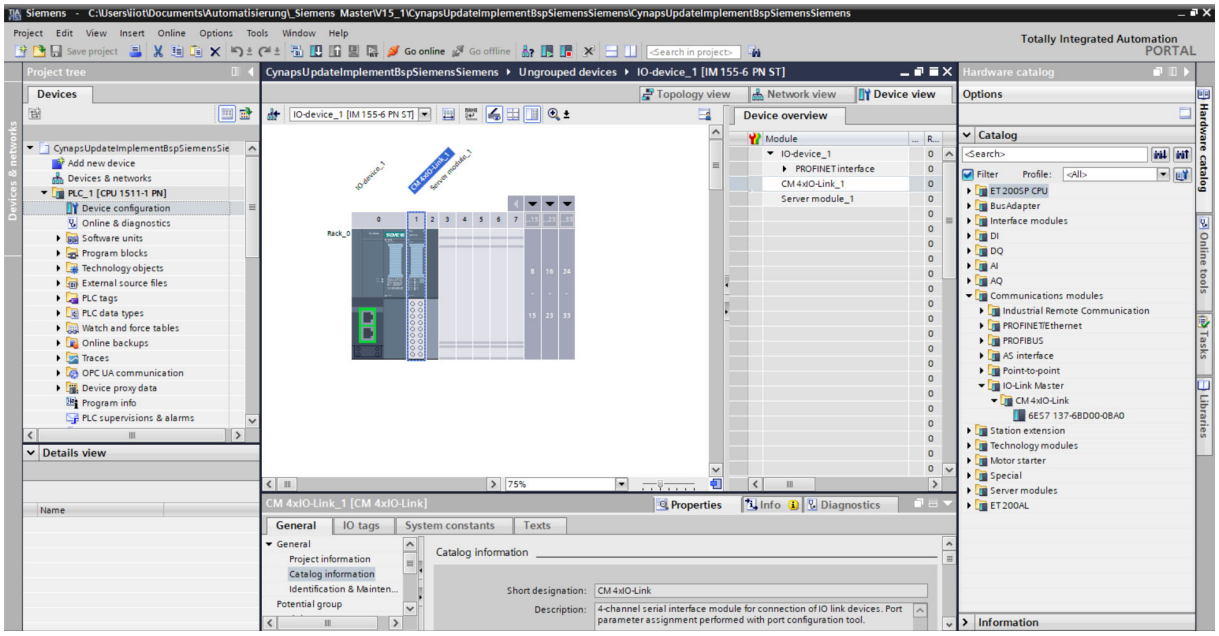
➲ You have the latest update available.
➲ The hardware configuration is error free.
➲ A communication with the hardware is possible.

Introduction

The following is a firmware update for cynapse®.

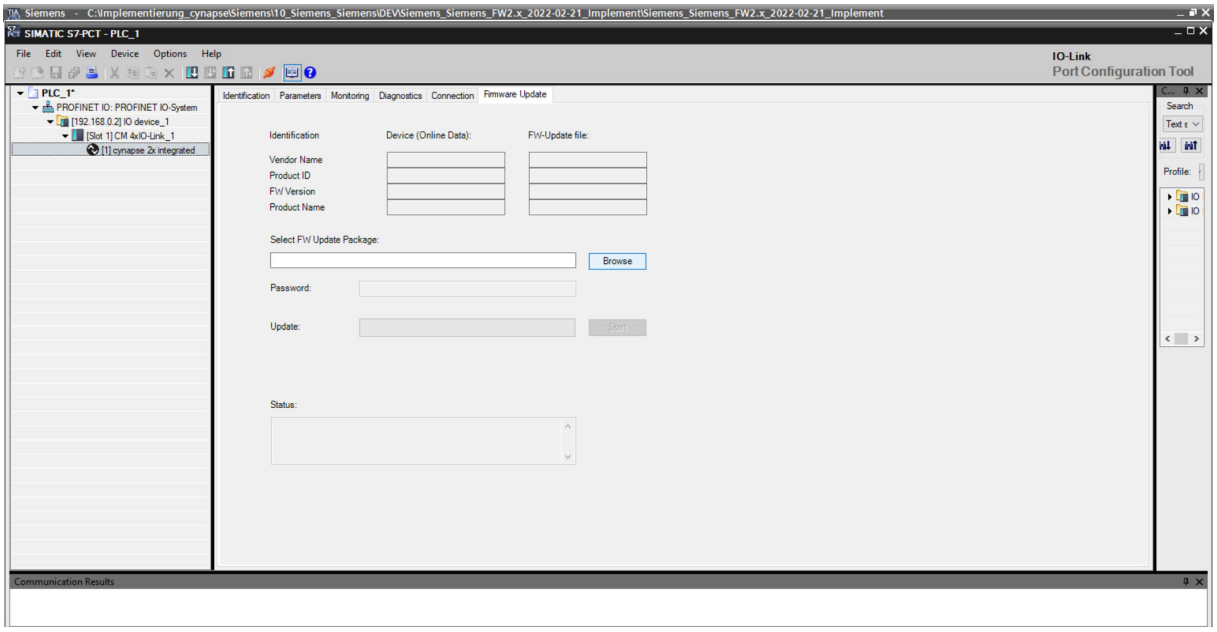
Procedure

1. Open the “Device configuration” in the TIA portal.
2. Switch to the “Device view” of the IO device.

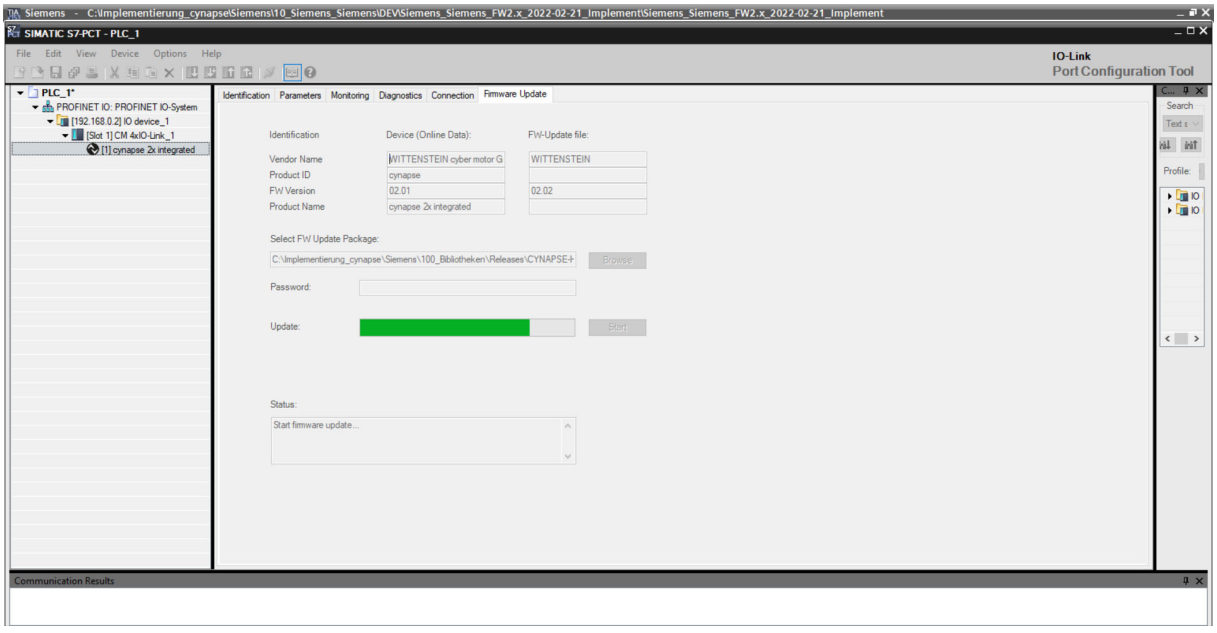


3. Move your mouse pointer over the CM 4xIO-Link module.
4. Right-click the dialog box and click “Start device tool”.
5. Select the device.
6. Open the “Firmware Update” tab.

7. Select the existing update from the Browse button.

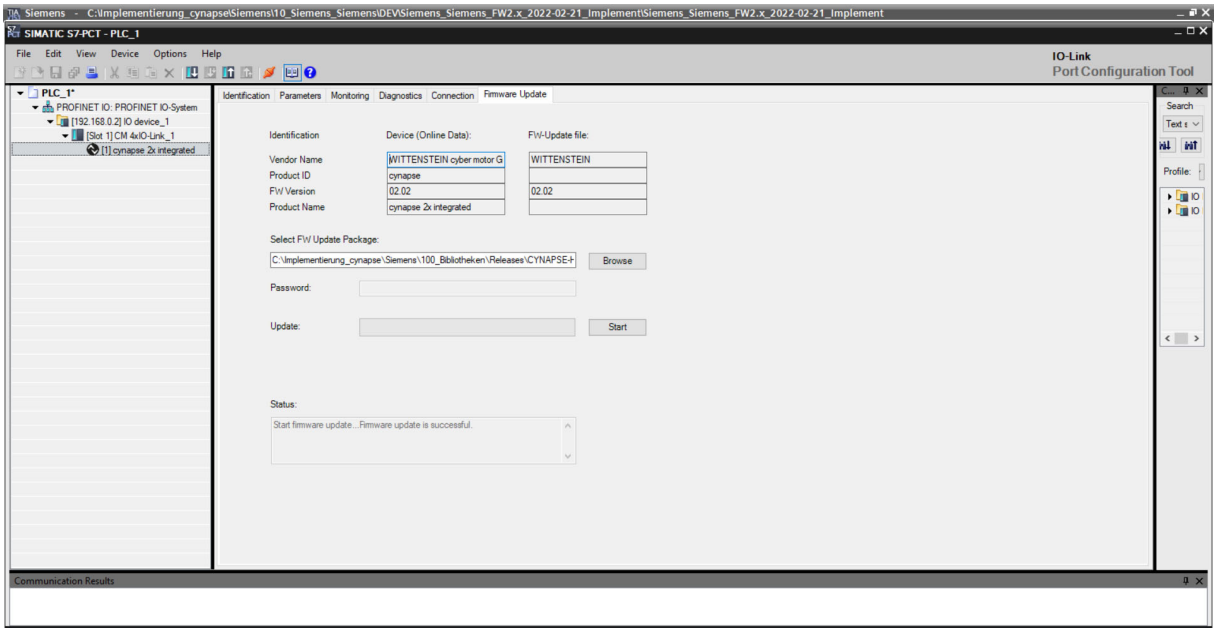


8. Click Start.



Result

➲ Status reports an update has been successfully performed.



Revision history

Revision	Date	Comment	Chapter
01	12/02/2019	New version	All
02	07/15/2022	cynapse® Trademark, Revision	All
03	06/20/2023	Translation EN	All

WITTENSTEIN alpha GmbH · Walter-Wittenstein-Straße 1 · 97999 Igersheim · Germany
Tel. +49 7931 493-0 · info@wittenstein.de

WITTENSTEIN – one with the future

www.wittenstein-alpha.de