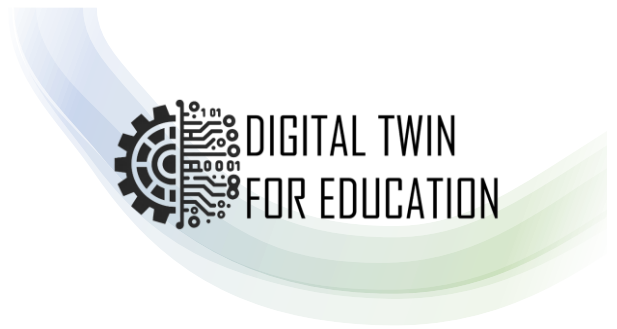


Module 4

Documentation
Key Concept Product development



DIGITAL TWIN FOR EDUCATION

Course materials

Digital twin documentation
Key concept for product development

Status: Feb 2026

Author: Dr. K. Krettenauer
C. Schmid
B. Geibel

Module 4

Documentation Key Concept Product development

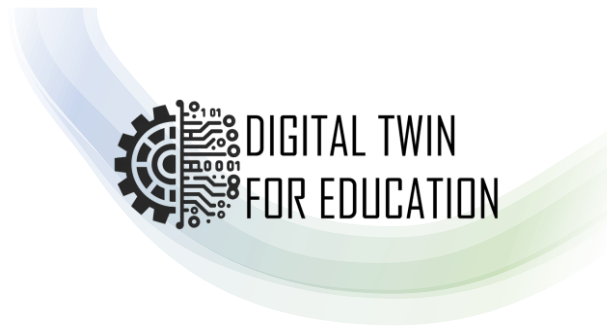


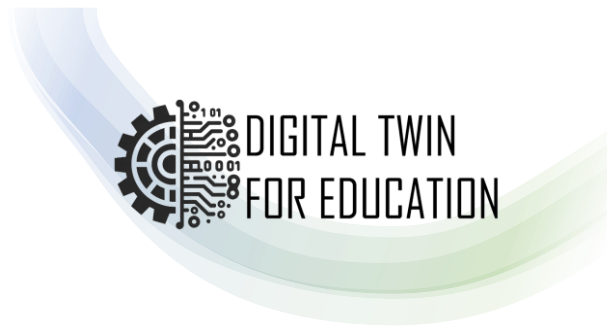
Table of Contents

| | | |
|----|---|----|
| 0 | Model analysis with task | 3 |
| 13 | Importing a symbol table for variable names..... | 6 |
| 14 | Selection and definition of parameters for the inputs in NX..... | 9 |
| 15 | Selection and definition of parameters for outputs in NX | 11 |
| 16 | Assignment of signals in the signal adapter | 12 |
| 17 | Connecting parameters and signals in the signal adapter | 15 |
| 18 | Connecting input parameters and signals in the signal adapter | 18 |
| 19 | Simulation release in the TIA Portal | 21 |
| 20 | Create PLC hardware and integrate PLC variables | 26 |
| 21 | Creating a function block in the TIA Portal..... | 29 |
| 22 | Program creation in the function block..... | 31 |
| 23 | Call the FB in Main OB1 and save the program | 33 |
| 24 | Creating a new simulation instance in PLCSim Advanced | 35 |
| 25 | Load TIA program in PLC Sim | 37 |
| 26 | Creating signal connections in MCD | 40 |
| 27 | Temporary storage and testing of the software status in NX | 45 |
| 28 | Testing the software status in the TIA Portal | 46 |
| 29 | Creation of a step chain program | 48 |
| 30 | Control of the outputs of the step chain program | 49 |
| 31 | Testing the software version with NX..... | 54 |
| 32 | Save status and close entire project..... | 55 |

Note: The numbering of the captions starts with 13 because of the continuing of the process chain. In Module 3 you already made the steps 1 to 12.

Module 4

Documentation
Key Concept Product development

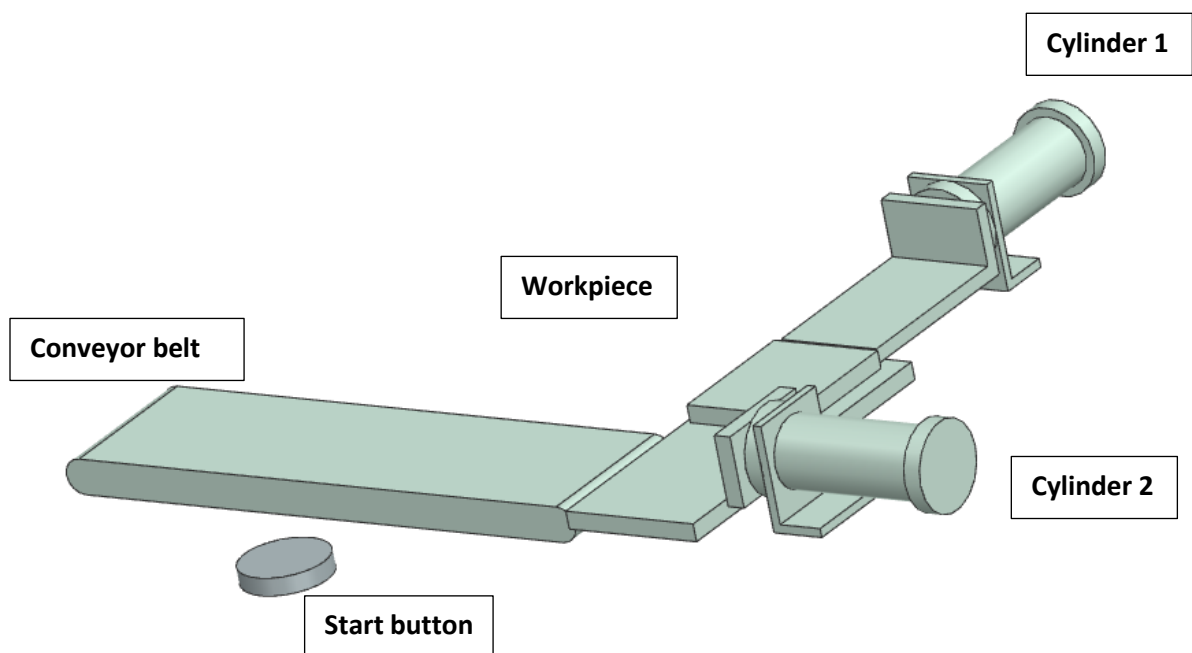


0 Model analysis with task

The NX model used for this exercise contains two pneumatic cylinders and a conveyor belt.

The task of the system is to first push a workpiece (rectangular body) laying on a plate with two pneumatic cylinders and then transport the workpiece away using the conveyor belt.

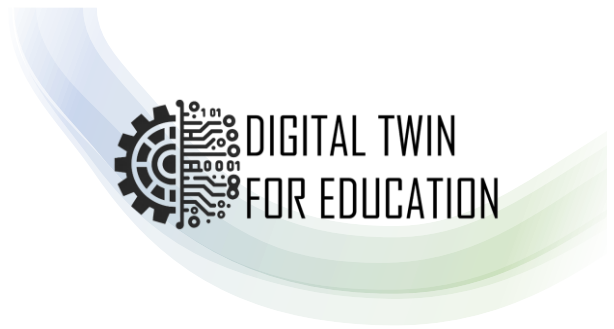
An additional start button can be used to initiate a sequence of steps for the sequence control. The double-acting pneumatic cylinders are controlled via solenoid valves and their position is monitored by limit switches.



In the following work process, the already existing CAD model (assembly in STEP file format) is imported to Siemens NX MCD .

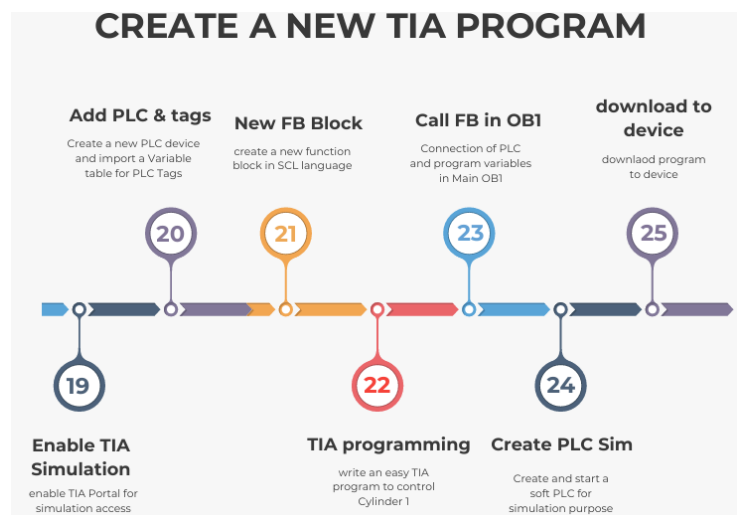
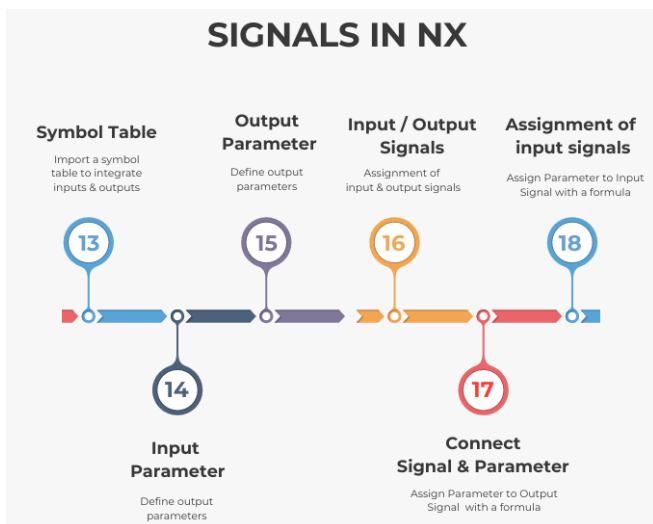
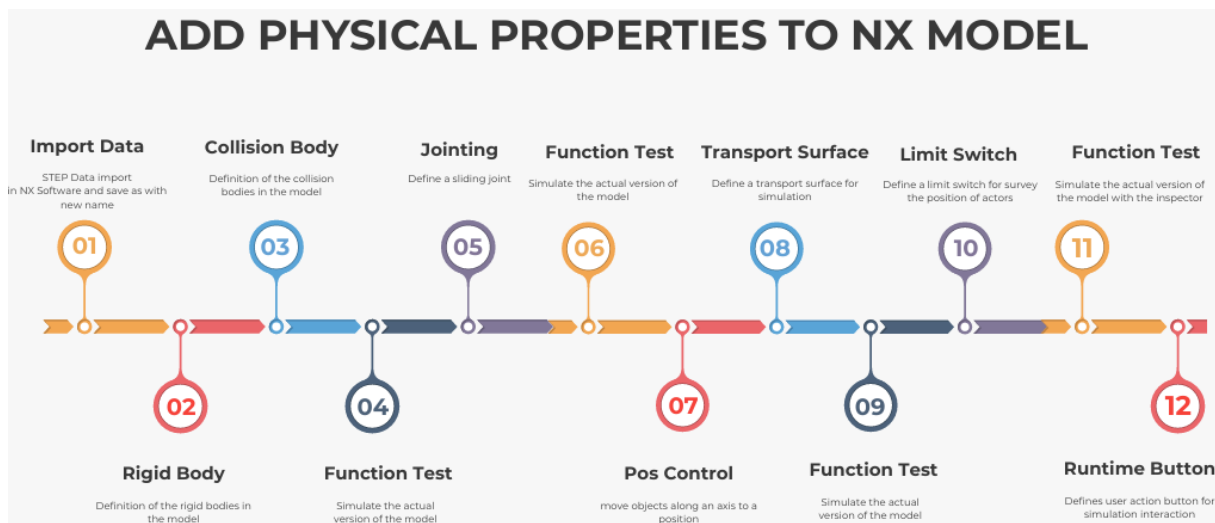
Module 4

Documentation Key Concept Product development



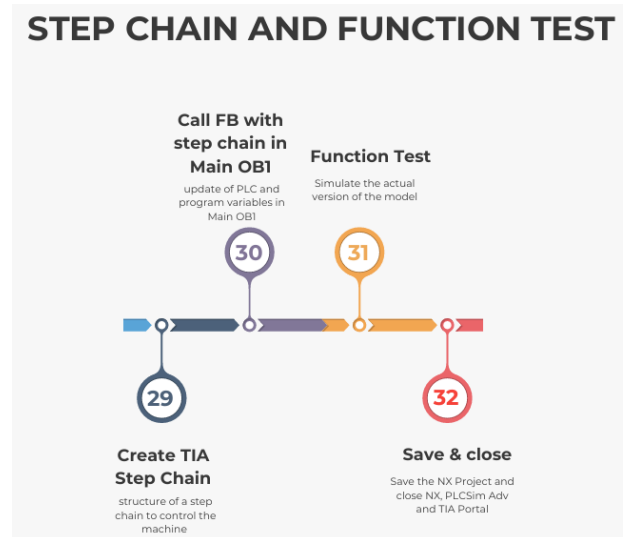
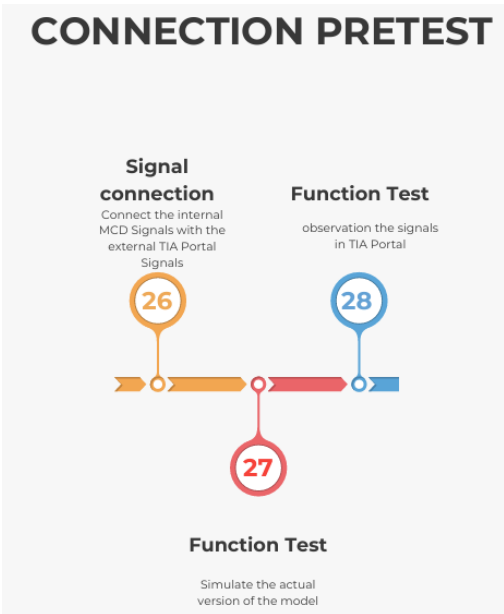
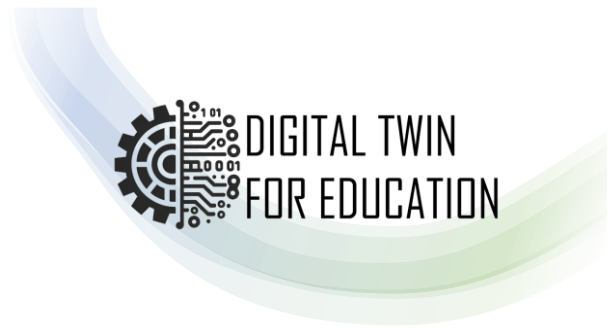
Parts of it will be made „physically“ movable so that the model can then be linked to a control system (PLC SIM advanced) and the associated programming environment (TIA-Portal) .

The whole work process is divided into five sections.



Module 4

Documentation
Key Concept Product development



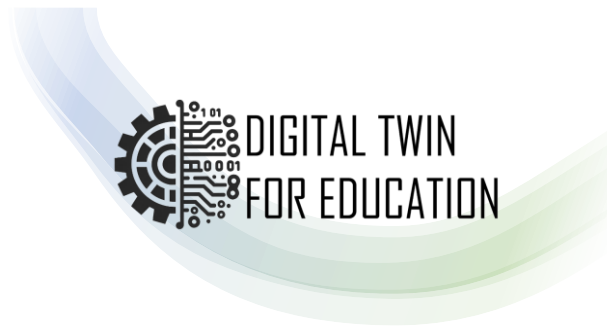
A crucial part of working with NX MCD is the creation of signals and their assignment.

Movements of actuators and sensor signals are assigned to parameters in order to be connected to PLC signals later on.

This mapping takes place in the **Signal Adapter**, which forms the core of communication.

Module 4

Documentation
Key Concept Product development

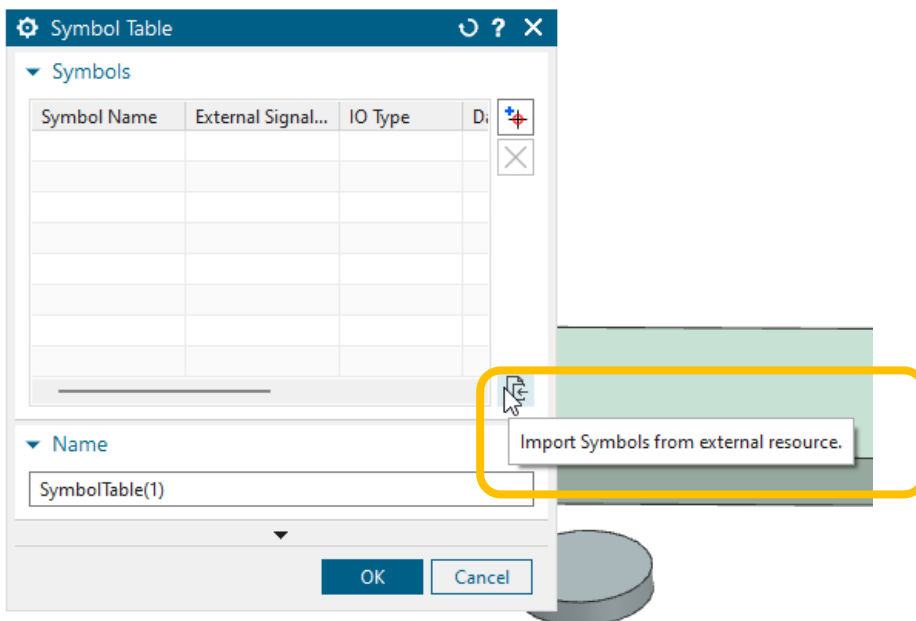
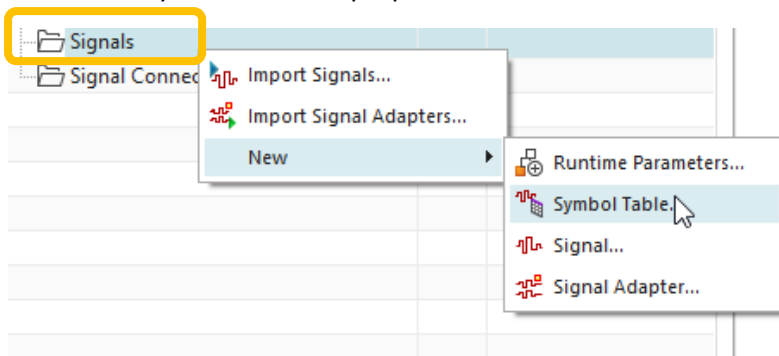


13 Importing a symbol table for variable names

Video: 13Import Symbols

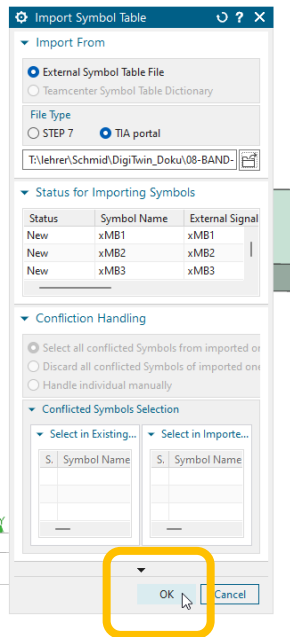
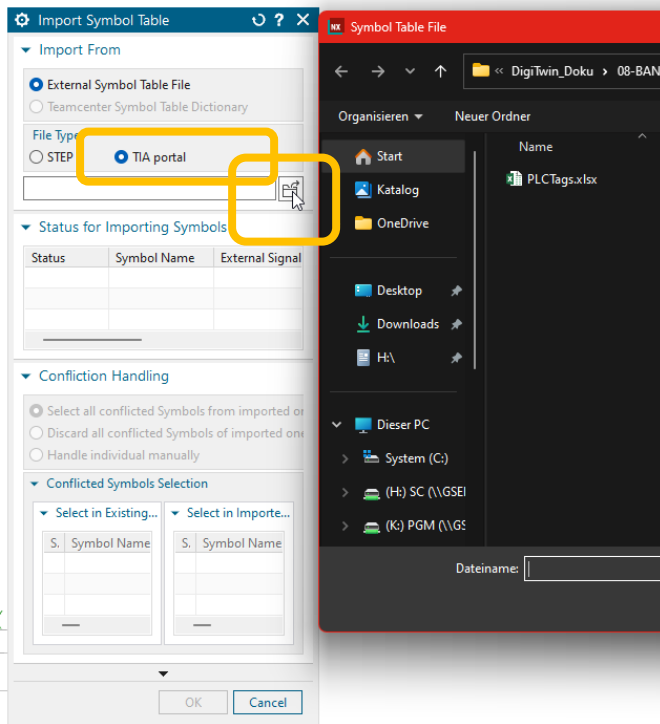
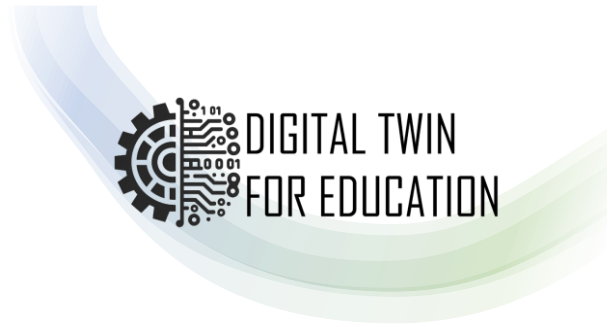
The next step is to import a symbol table containing the variable names of the PLC variables from the TIA Portal project in order to facilitate the signal mapping with the PLC program later on.

If, as in our example, the naming structure is identical, the variable assignment can be done automatically for simulation purposes.



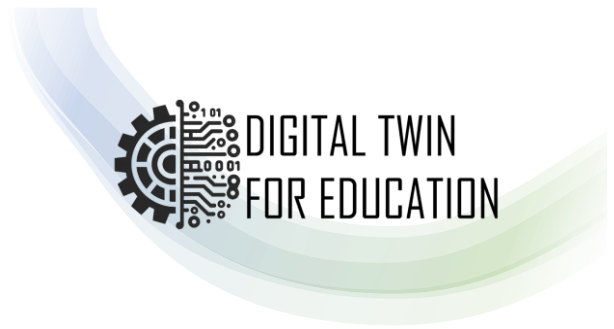
Module 4

Documentation Key Concept Product development



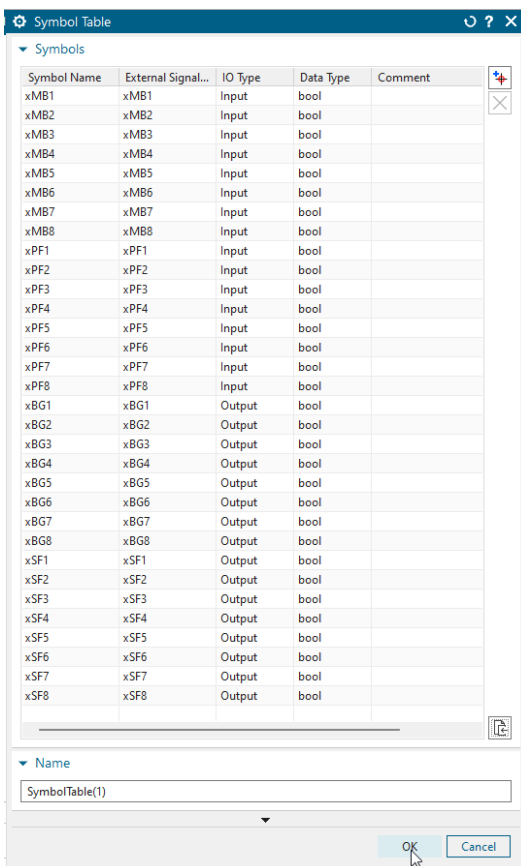
Module 4

Documentation Key Concept Product development

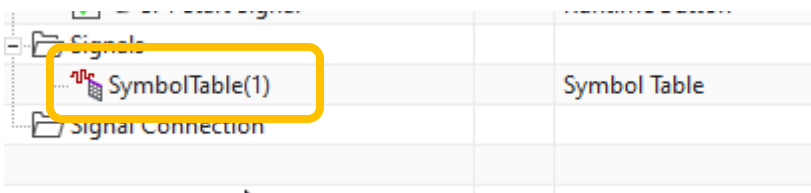


In the symbol table in the NX model, you can see that the assignment for MB1 (solenoid valve), for example, is declared as an input, and for sensors such as BG1 (end position monitoring), it is declared as an output.

The reason for this is that the variables are viewed from the perspective of the NX model, and in this case, a sensor is an output because it delivers a signal to the controller, and the solenoid valve is an input because the signal comes into the NX model as an input signal from the PLC.

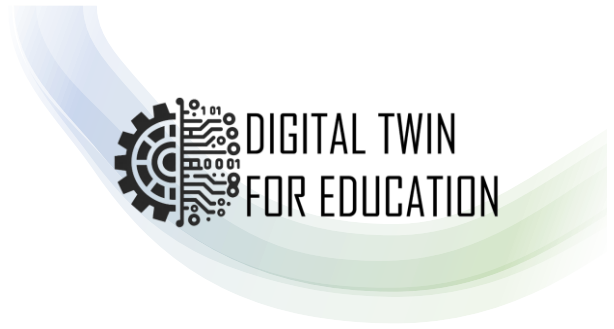


The symbol table is then generated automatically.



Module 4

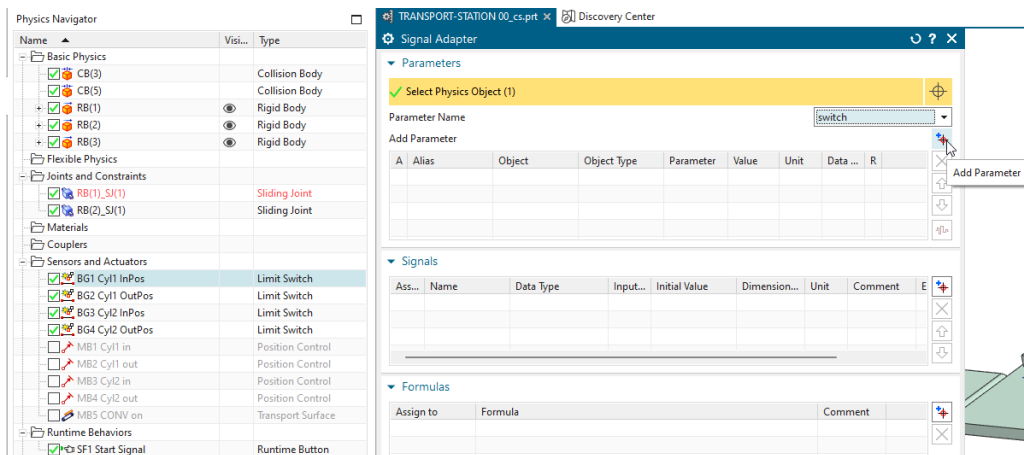
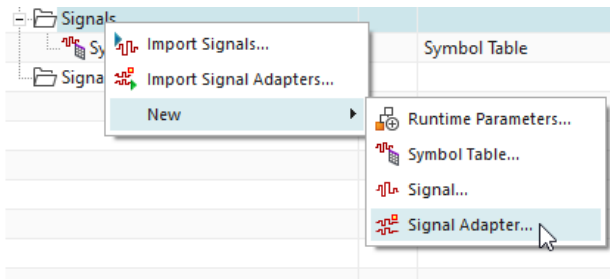
Documentation Key Concept Product development



14 Selection and definition of parameters for the inputs in NX

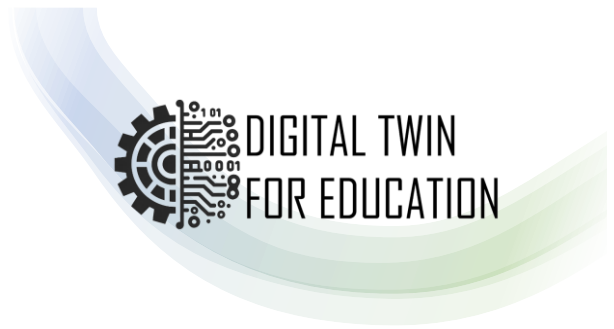
Video: 14Signals-Ad-Inputs

In this step, parameters for individual sensors are defined specifically according to the respective application.



Module 4

Documentation Key Concept Product development



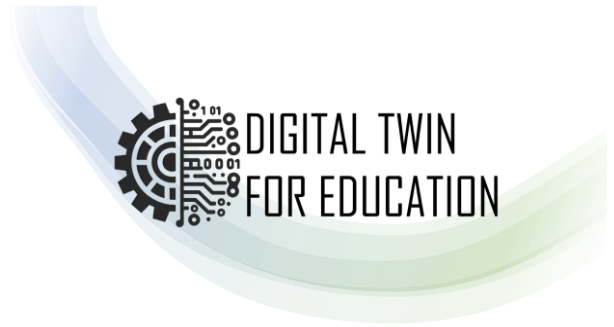
The limit position sensors are defined with the parameter type "switch" and the start button with the parameter type "triggered."

The screenshot shows the 'Signal Adapter' software window. It has a dark blue header with a gear icon, the title 'Signal Adapter', and standard window controls. The main area is divided into several sections: 'Parameters' with a yellow header and a 'Select Physics Object (1)' button; a 'Parameter Name' dropdown set to 'triggered'; an 'Add Parameter' button; a table of parameters; 'Signals' section with an empty table; 'Formulas' section with an empty table and a formula input field; and a 'Name' field containing 'SignalAdapter(1)'. At the bottom, there are 'OK', 'Apply', and 'Cancel' buttons.

| A | Alias | Object | Object Type | Parameter | Value | Unit | Data ... | R |
|---|-------------|------------------|----------------|-----------|-------|------|----------|---|
| | Parameter_1 | BG1 Cyl1 InPos | Limit Switch | switch | false | | bool | R |
| | Parameter_2 | BG2 Cyl1 OutPos | Limit Switch | switch | false | | bool | R |
| | Parameter_3 | BG3 Cyl2 InPos | Limit Switch | switch | false | | bool | R |
| | Parameter_4 | BG4 Cyl2 OutPos | Limit Switch | switch | false | | bool | R |
| | Parameter_5 | SF1 Start Signal | Runtime Button | triggered | false | | bool | R |

Module 4

Documentation Key Concept Product development

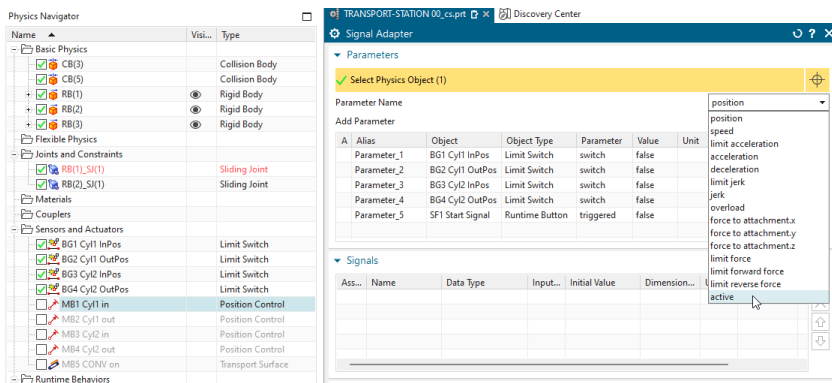


15 Selection and definition of parameters for outputs in NX

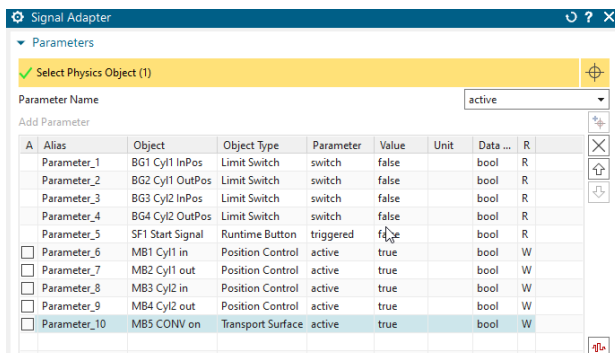
Video: 15Signals-Ad-Outputs

In this step, parameters for individual actuators are defined specifically for the respective application.

| | | |
|-------------------|----------------|--|
| Signals | | |
| SignalAdapter(1) | Signal Adapter | |
| SymbolTable(1) | Symbol Table | |
| Signal Connection | | |

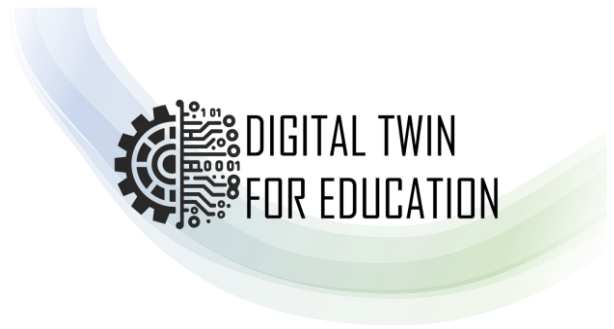


When defining the parameters, a definition with the parameter type "active" is used.



Module 4

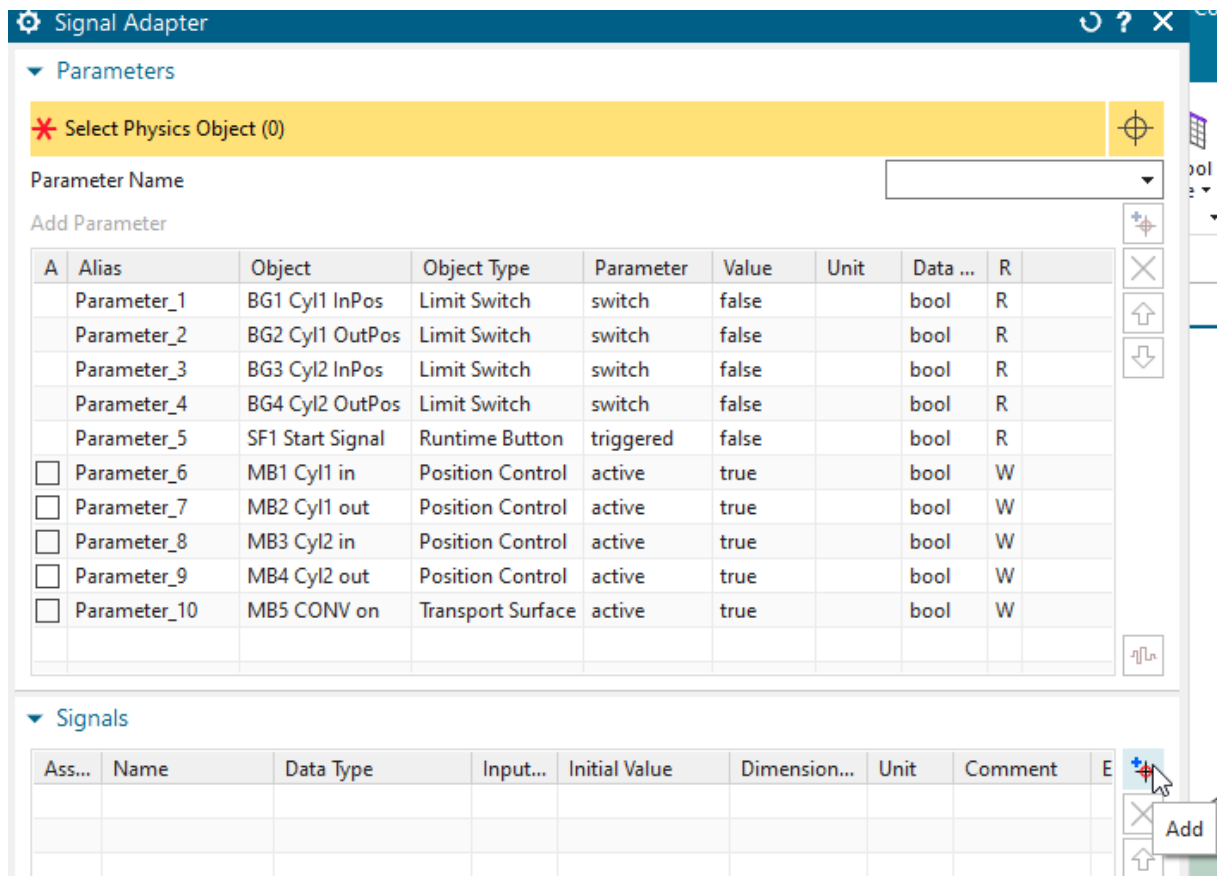
Documentation
Key Concept Product development



16 Assignment of signals in the signal adapter

Video: 16Add_Signals

All input and output signals are then created in the signal adapter.



The screenshot shows the 'Signal Adapter' window with two main sections: 'Parameters' and 'Signals'.

Parameters Section:

- A yellow banner at the top says '* Select Physics Object (0)'.
- A dropdown menu for 'Parameter Name' is empty.
- A table lists 10 parameters with columns: A, Alias, Object, Object Type, Parameter, Value, Unit, Data ..., and R.

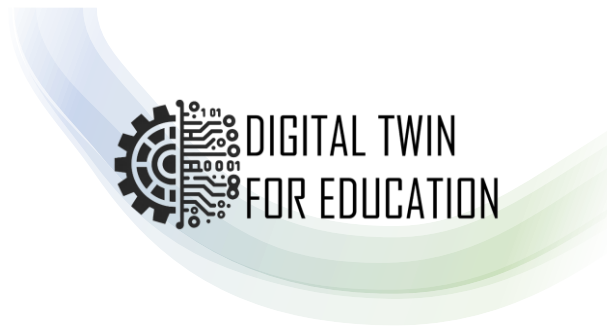
| A | Alias | Object | Object Type | Parameter | Value | Unit | Data ... | R |
|--------------------------|--------------|------------------|-------------------|-----------|-------|------|----------|---|
| | Parameter_1 | BG1 Cyl1 InPos | Limit Switch | switch | false | | bool | R |
| | Parameter_2 | BG2 Cyl1 OutPos | Limit Switch | switch | false | | bool | R |
| | Parameter_3 | BG3 Cyl2 InPos | Limit Switch | switch | false | | bool | R |
| | Parameter_4 | BG4 Cyl2 OutPos | Limit Switch | switch | false | | bool | R |
| | Parameter_5 | SF1 Start Signal | Runtime Button | triggered | false | | bool | R |
| <input type="checkbox"/> | Parameter_6 | MB1 Cyl1 in | Position Control | active | true | | bool | W |
| <input type="checkbox"/> | Parameter_7 | MB2 Cyl1 out | Position Control | active | true | | bool | W |
| <input type="checkbox"/> | Parameter_8 | MB3 Cyl2 in | Position Control | active | true | | bool | W |
| <input type="checkbox"/> | Parameter_9 | MB4 Cyl2 out | Position Control | active | true | | bool | W |
| <input type="checkbox"/> | Parameter_10 | MB5 CONV on | Transport Surface | active | true | | bool | W |

Signals Section:

- A table with columns: Ass..., Name, Data Type, Input..., Initial Value, Dimension..., Unit, Comment, and E.
- The table is currently empty.
- An 'Add' button is visible in the bottom right corner of the signals table.

Module 4

Documentation
Key Concept Product development



▼ Signals

| Ass... | Name | Data Type | Input... | Initial Value | Dimension... | Unit | Comment | E |
|--------------------------|------|-----------|----------|---------------|--------------|------|---------|---|
| <input type="checkbox"/> | xBG1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG9 | bool | Output | false | | | | |
| | xBG9 | | | | | | | |
| | xBG2 | | | | | | | |
| | xBG3 | | | | | | | |
| | xBG4 | | | | | | | |
| | xBG5 | | | | | | | |
| | xBG6 | | | | | | | |
| | xBG7 | | | | | | | |
| | xBG8 | | | | | | | |
| | xSF1 | | | | | | | |
| | xSF2 | | | | | | | |

▼ Formulas

| Assign to | Formula | Comment |
|-----------|---------|---------|
|-----------|---------|---------|

▼ Signals

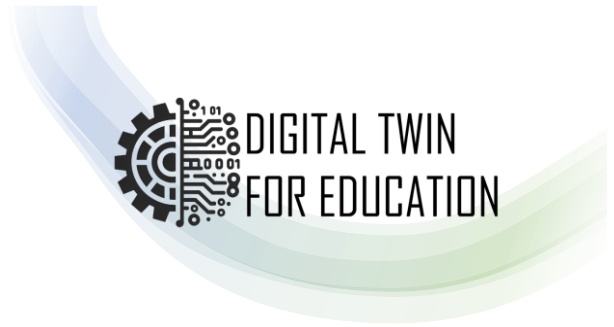
| Ass... | Name | Data Type | Input... | Initial Value | Dimension... | Unit | Comment | E |
|--------------------------|------|-----------|----------|---------------|--------------|------|---------|---|
| <input type="checkbox"/> | xBG1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG2 | bool | Output | false | | | | |

▼ Formulas

| | | | | |
|--------------------------|------|------|--------|-------|
| <input type="checkbox"/> | xBG5 | bool | Output | false |
| | | | Input | |
| | | | Output | |

Module 4

Documentation
Key Concept Product development



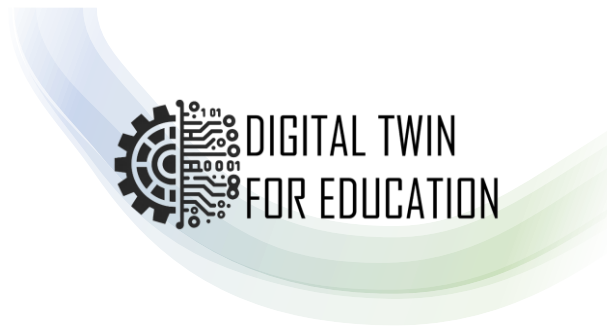
The assignment of whether it is an INPUT or OUTPUT is made from the perspective of the NX model. This means that an actuator is defined as an INPUT and a sensor as an OUTPUT.

▼ Signals

| Ass... | Name | Data Type | Input... | Initial Value | Dimension... | Unit | Comment | E |
|--------------------------|------|-----------|----------|---------------|--------------|------|---------|---|
| <input type="checkbox"/> | xBG1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG2 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG3 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG4 | bool | Output | false | | | | |
| <input type="checkbox"/> | xF1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xMB1 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB2 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB3 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB4 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB5 | bool | Input | false | | | | |

Module 4

Documentation Key Concept Product development

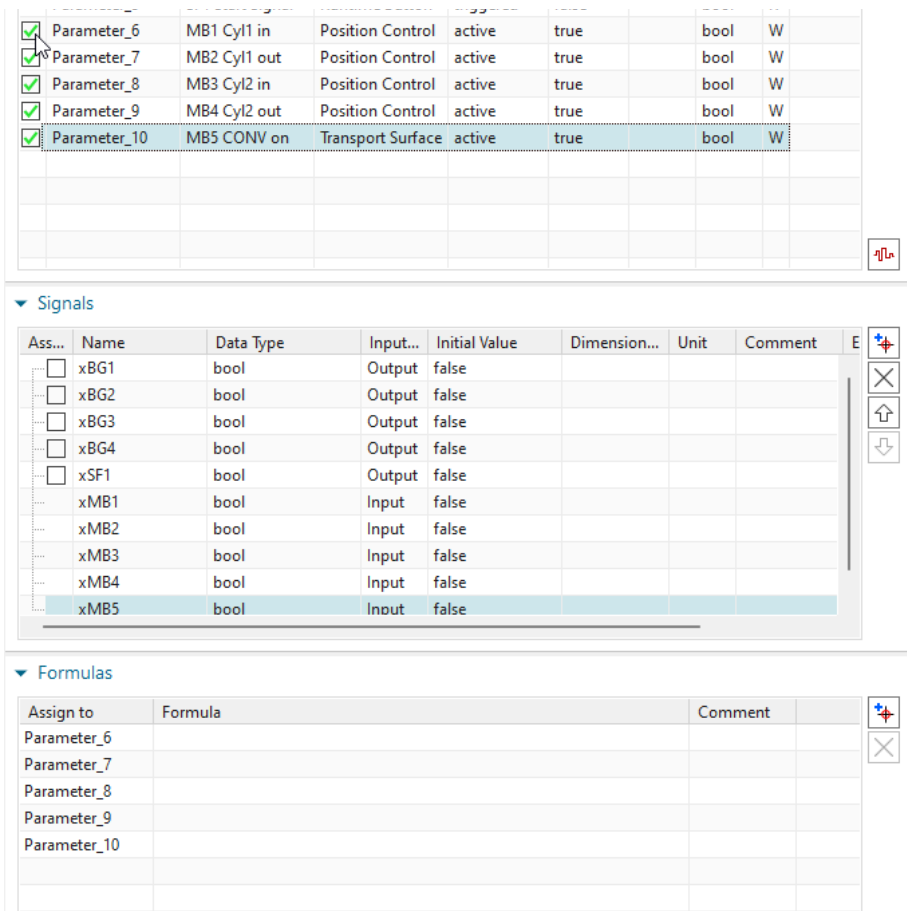


17 Connecting parameters and signals in the signal adapter

Video: 17Connect_MB

In this step, the individual signals are manually merged with the respective matching parameter.

In the first step, the output parameters are checked and then assigned to the corresponding signal via the Formulas area.



The screenshot displays a software interface with three main sections:

- Parameters Table:** A table with columns for parameter name, description, control type, status, and data type. The first five rows are checked.
- Signals Table:** A table with columns for name, data type, input/output status, and initial value. The row for xMB5 is highlighted.
- Formulas Table:** A table with columns for 'Assign to' and 'Formula'. The 'Assign to' column lists Parameter_6 through Parameter_10.

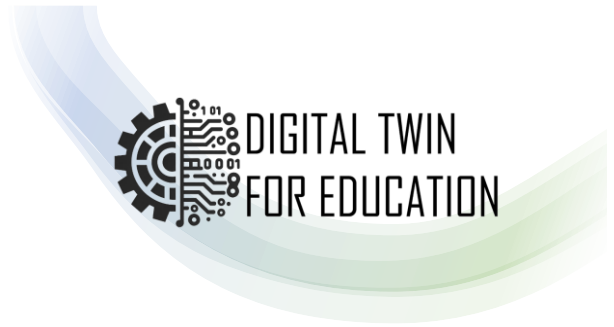
| Parameter | Description | Control | Status | Data Type |
|--|--------------|-------------------|--------|-----------|
| <input checked="" type="checkbox"/> Parameter_6 | MB1 Cyl1 in | Position Control | active | bool |
| <input checked="" type="checkbox"/> Parameter_7 | MB2 Cyl1 out | Position Control | active | bool |
| <input checked="" type="checkbox"/> Parameter_8 | MB3 Cyl2 in | Position Control | active | bool |
| <input checked="" type="checkbox"/> Parameter_9 | MB4 Cyl2 out | Position Control | active | bool |
| <input checked="" type="checkbox"/> Parameter_10 | MB5 CONV on | Transport Surface | active | bool |

| Ass... | Name | Data Type | Input... | Initial Value | Dimension... | Unit | Comment |
|--------------------------|------|-----------|----------|---------------|--------------|------|---------|
| <input type="checkbox"/> | xBG1 | bool | Output | false | | | |
| <input type="checkbox"/> | xBG2 | bool | Output | false | | | |
| <input type="checkbox"/> | xBG3 | bool | Output | false | | | |
| <input type="checkbox"/> | xBG4 | bool | Output | false | | | |
| <input type="checkbox"/> | xF1 | bool | Output | false | | | |
| <input type="checkbox"/> | xMB1 | bool | Input | false | | | |
| <input type="checkbox"/> | xMB2 | bool | Input | false | | | |
| <input type="checkbox"/> | xMB3 | bool | Input | false | | | |
| <input type="checkbox"/> | xMB4 | bool | Input | false | | | |
| <input type="checkbox"/> | xMB5 | bool | Input | false | | | |

| Assign to | Formula | Comment |
|--------------|---------|---------|
| Parameter_6 | | |
| Parameter_7 | | |
| Parameter_8 | | |
| Parameter_9 | | |
| Parameter_10 | | |

Module 4

Documentation
Key Concept Product development



▼ Formulas

| Assign to | Formula | Comment | |
|--------------|---------|---------|--|
| Parameter_6 | xMB1 | | |
| Parameter_7 | | | |
| Parameter_8 | | | |
| Parameter_9 | | | |
| Parameter_10 | | | |
| | | | |
| | | | |
| | | | |

Formula

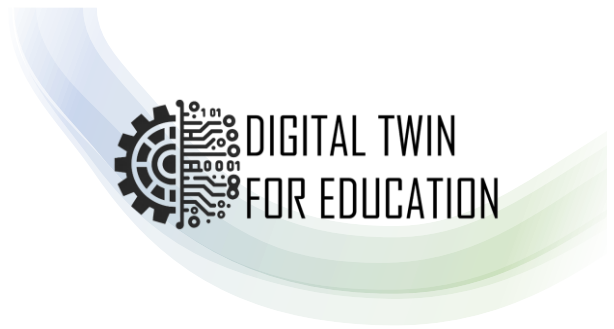
x|

- xBG1
- xBG2
- xBG3
- xBG4
- xMB1
- xMB2
- xMB3

Cancel

Module 4

Documentation
Key Concept Product development



▼ Signals

| Ass... | Name | Data Type | Input... | Initial Value | Dimension... | Unit | Comment | E |
|--------------------------|------|-----------|----------|---------------|--------------|------|---------|---|
| <input type="checkbox"/> | xBG1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG2 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG3 | bool | Output | false | | | | |
| <input type="checkbox"/> | xBG4 | bool | Output | false | | | | |
| <input type="checkbox"/> | xSF1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xMB1 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB2 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB3 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB4 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB5 | bool | Input | false | | | | |

▼ Formulas

| Assign to | Formula | Comment |
|--------------|---------|---------|
| Parameter_6 | xMB1 | |
| Parameter_7 | xMB2 | |
| Parameter_8 | xMB3 | |
| Parameter_9 | xMB4 | |
| Parameter_10 | xMB5 | |

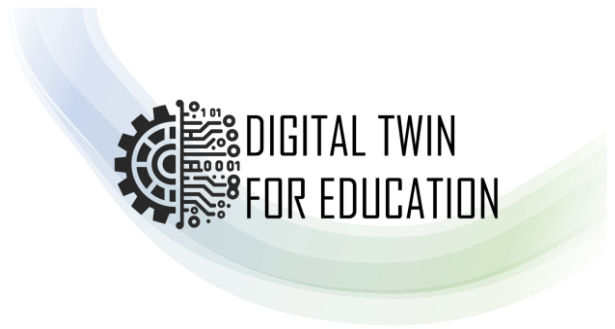
Formula

▼ Name

OK Cancel

Module 4

Documentation Key Concept Product development



18 Connecting input parameters and signals in the signal adapter

Video: 17Connect_BG

In this step, the individual signals are manually merged with the respective matching parameters.

In this step, the input signals are checked and then assigned to the corresponding parameter via the Formulas area.

This corresponds to a reverse assignment as in the previous point.

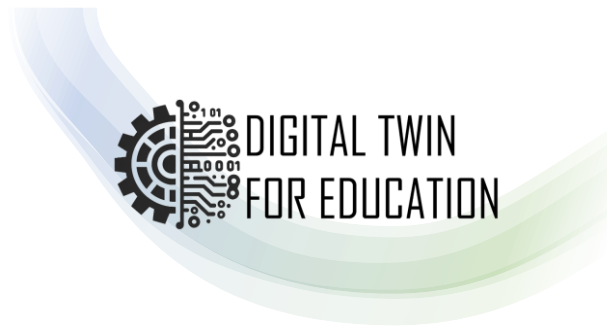
The screenshot shows a software interface with two main panels. The top panel, titled 'Signals', contains a table with columns: Ass..., Name, Data Type, Input..., Initial Value, Dimension..., Unit, Comment, and E. The table lists several signals, with 'xSF1' selected. A context menu is open over 'xSF1', showing options: Edit, Delete, Rename, and Add to Inspector. The bottom panel, titled 'Formulas', contains a table with columns: Assign to, Formula, and Comment. It shows assignments for parameters xMB1 through xMB5 to signals xBG1 through xBG4 and xSF1.

| Ass... | Name | Data Type | Input... | Initial Value | Dimension... | Unit | Comment | E |
|-------------------------------------|------|-----------|----------|---------------|--------------|------|---------|---|
| <input checked="" type="checkbox"/> | xBG1 | bool | Output | false | | | | |
| <input checked="" type="checkbox"/> | xBG2 | bool | Output | false | | | | |
| <input checked="" type="checkbox"/> | xBG3 | bool | Output | false | | | | |
| <input checked="" type="checkbox"/> | xBG4 | bool | Output | false | | | | |
| <input checked="" type="checkbox"/> | xSF1 | bool | Output | false | | | | |
| <input type="checkbox"/> | xMB1 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB2 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB3 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB4 | bool | Input | false | | | | |
| <input type="checkbox"/> | xMB5 | bool | Input | false | | | | |

| Assign to | Formula | Comment |
|--------------|---------|---------|
| Parameter_6 | xMB1 | |
| Parameter_7 | xMB2 | |
| Parameter_8 | xMB3 | |
| Parameter_9 | xMB4 | |
| Parameter_10 | xMB5 | |
| xBG1 | | |
| xBG2 | | |
| xBG3 | | |
| xBG4 | | |
| xSF1 | | |

Module 4

Documentation
Key Concept Product development



▼ Formulas

| Assign to | Formula | Comment | |
|--------------|-------------|---------|---|
| Parameter_6 | xMB1 | | + |
| Parameter_7 | xMB2 | | |
| Parameter_8 | xMB3 | | X |
| Parameter_9 | xMB4 | | |
| Parameter_10 | xMB5 | | |
| xBG2 | Parameter_2 | | |
| xBG3 | Parameter_3 | | |
| xBG4 | Parameter_4 | | |
| xSF1 | | | |
| xBG1 | Parameter_1 | | |
| | | | |

Formula

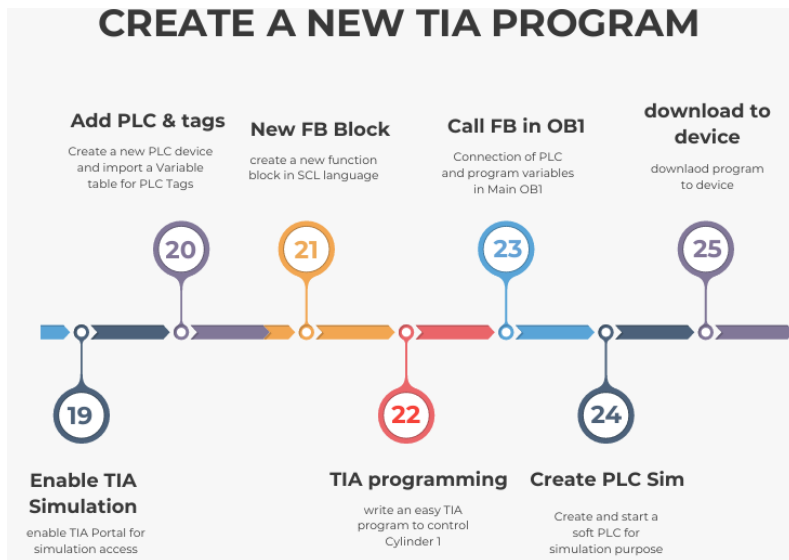
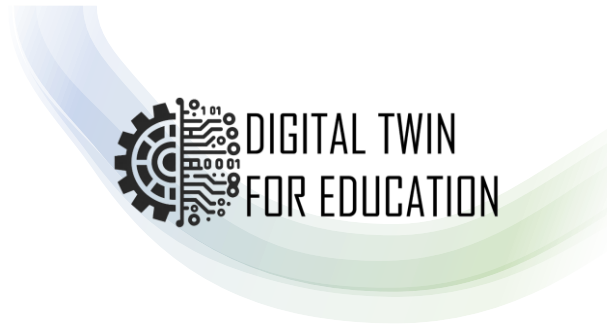
Parameter_5 f(x) ↕ ab

▼ Name

SignalAdapter(1)

Module 4

Documentation
Key Concept Product development



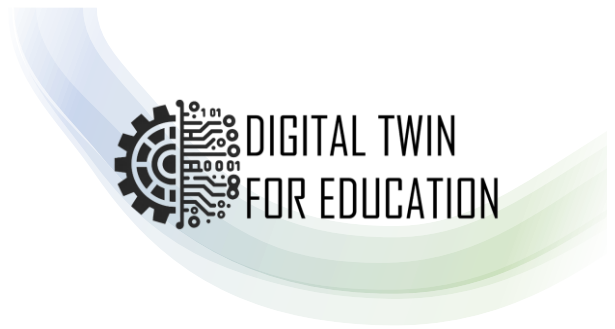
In this section we create the TIA Portal Project for the programming part of the digital twin.

We start with a new function block FB and add the code to control one cylinder.

PLC Sim advanced takes the role of a soft PLC and we download our program to an instance of it.

Module 4

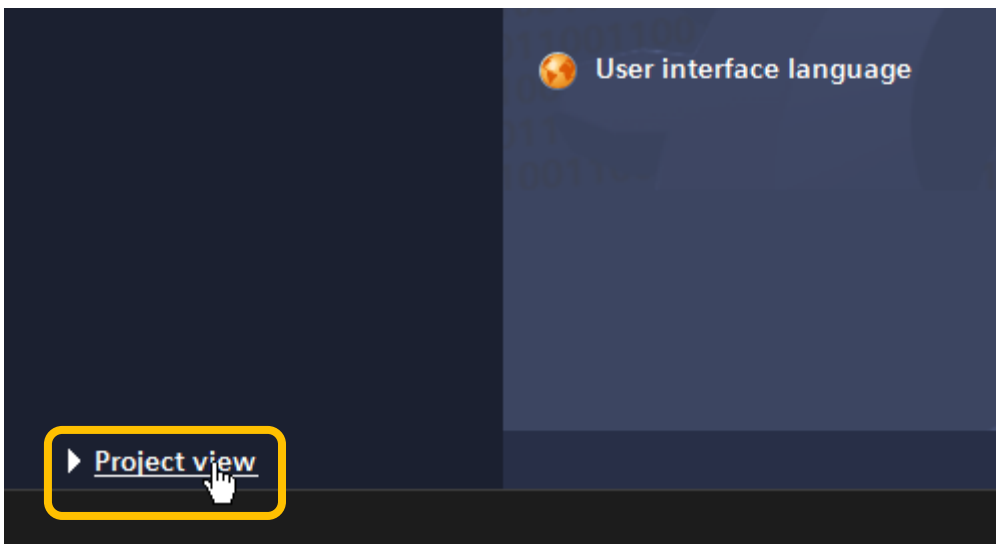
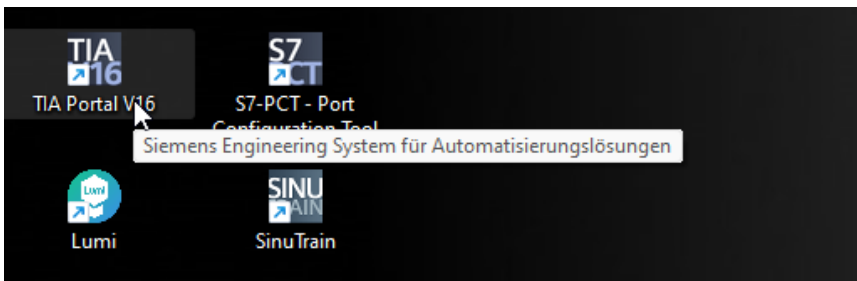
Documentation
Key Concept Product development



19 Simulation release in the TIA Portal

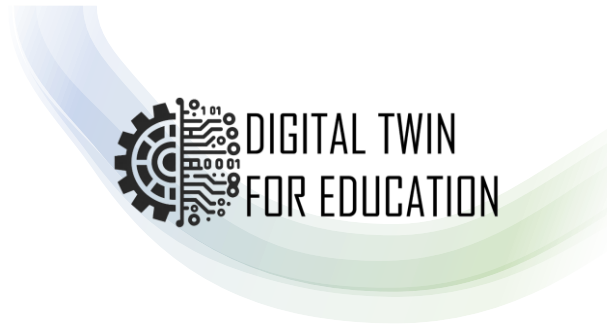
Video: 19TIA_new_enable_Simulation

In order to work with a simulation in the TIA Portal, this simulation option must first be activated in an empty project.

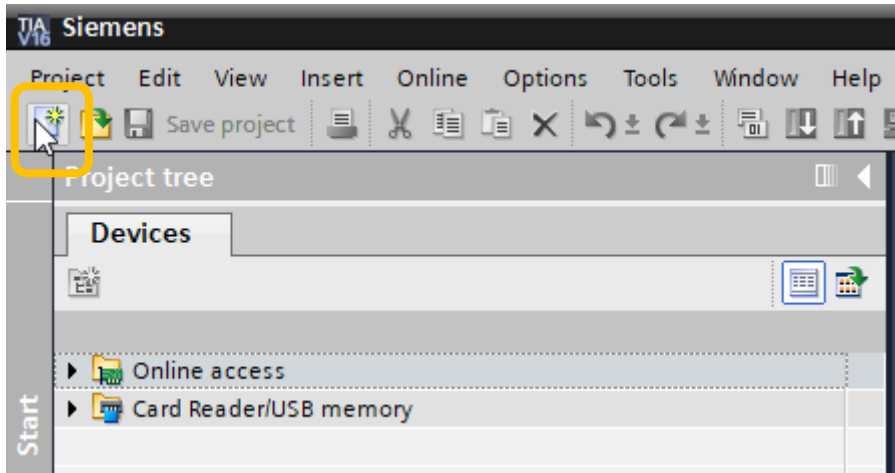


Module 4

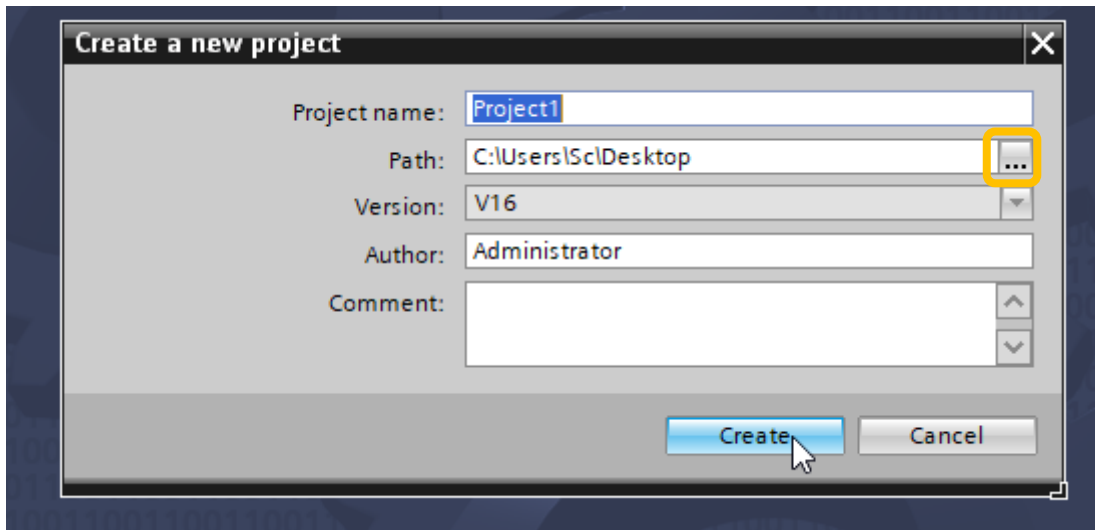
Documentation
Key Concept Product development



Create empty/new project

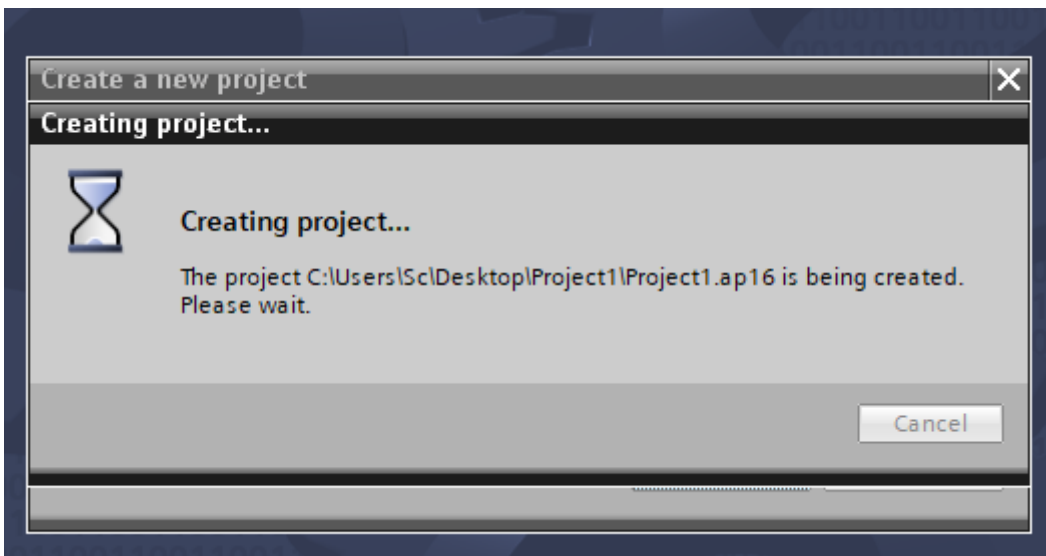
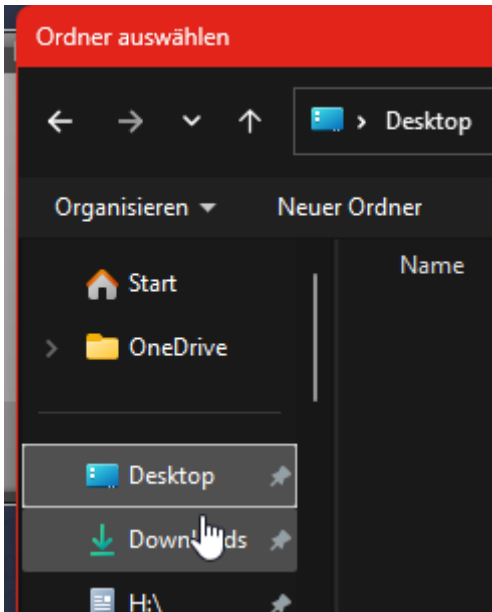
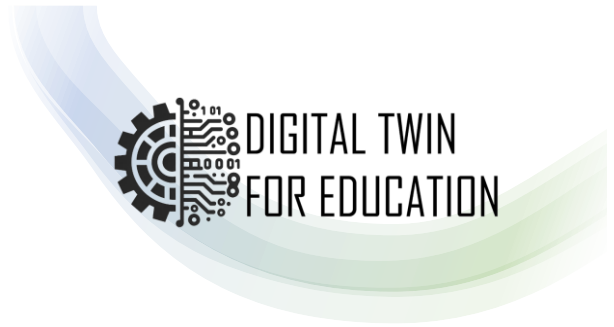


Set the path to the C:// drive (this is different in the video, please use your desktop)



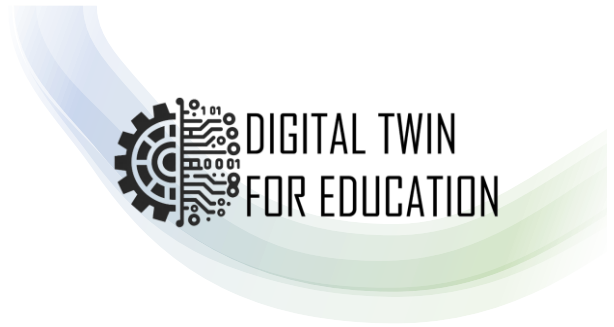
Module 4

Documentation
Key Concept Product development

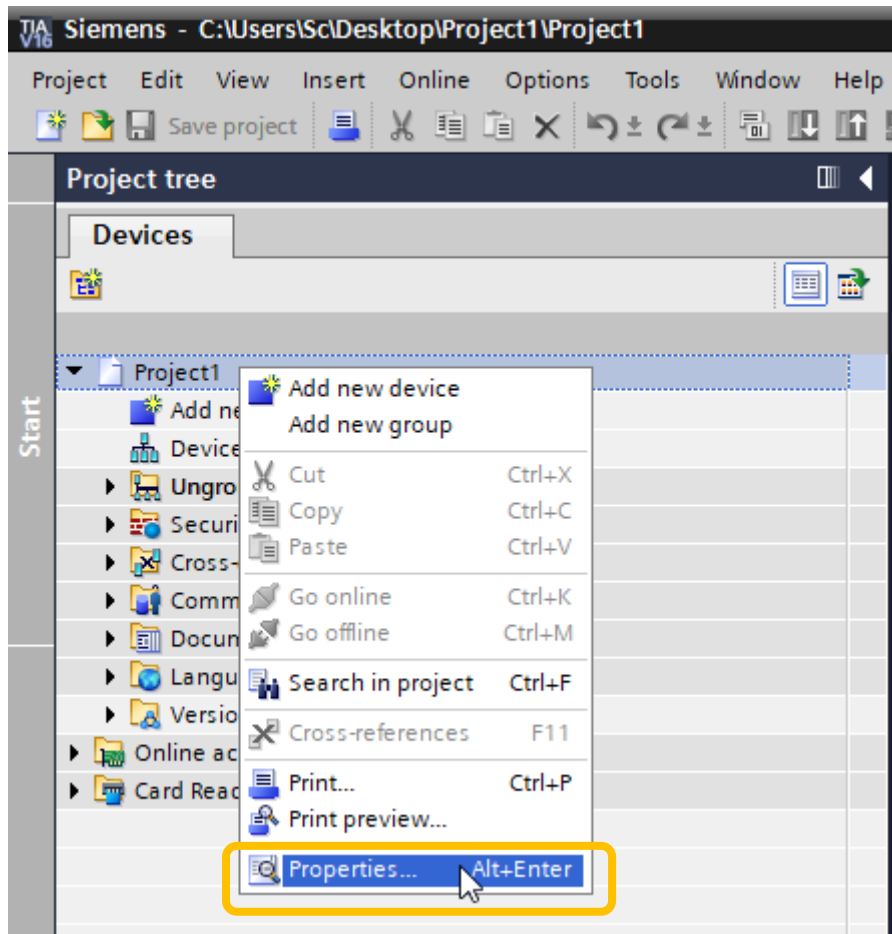


Module 4

Documentation
Key Concept Product development

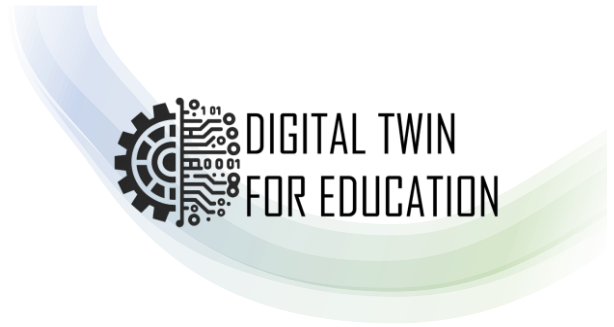


Right-click on the project and then select Properties

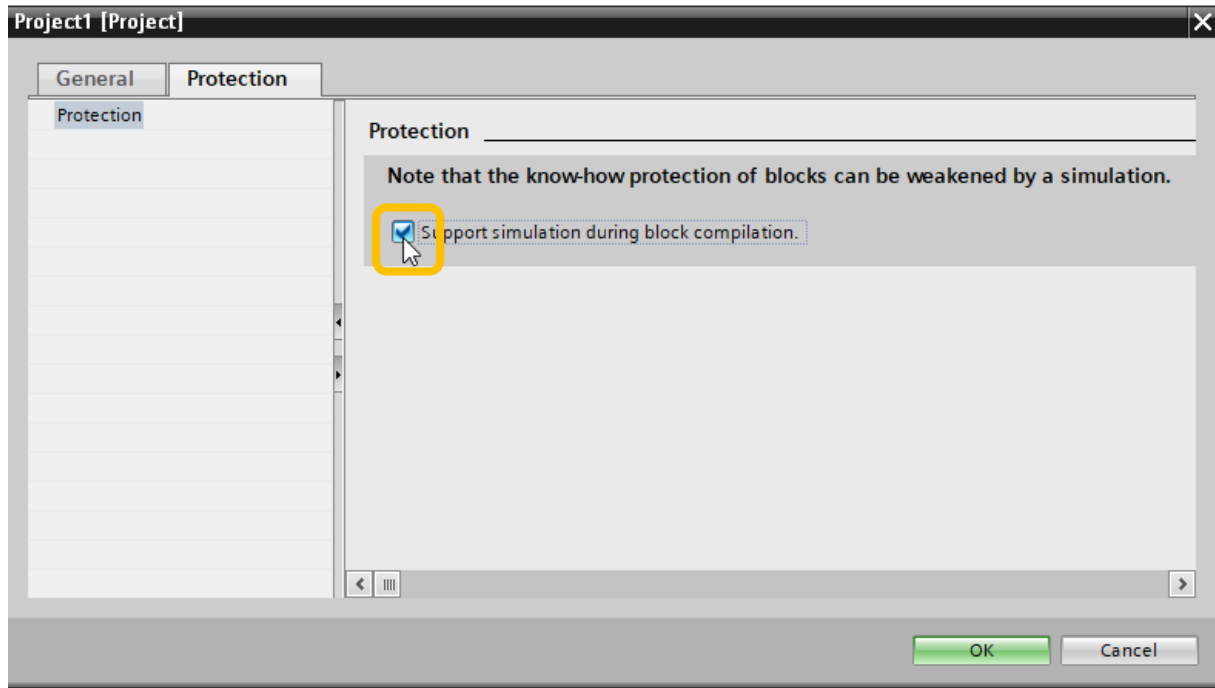


Module 4

Documentation
Key Concept Product development

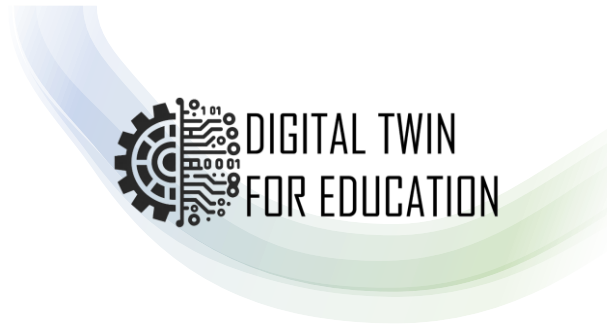


Simulation must be activated



Module 4

Documentation
Key Concept Product development

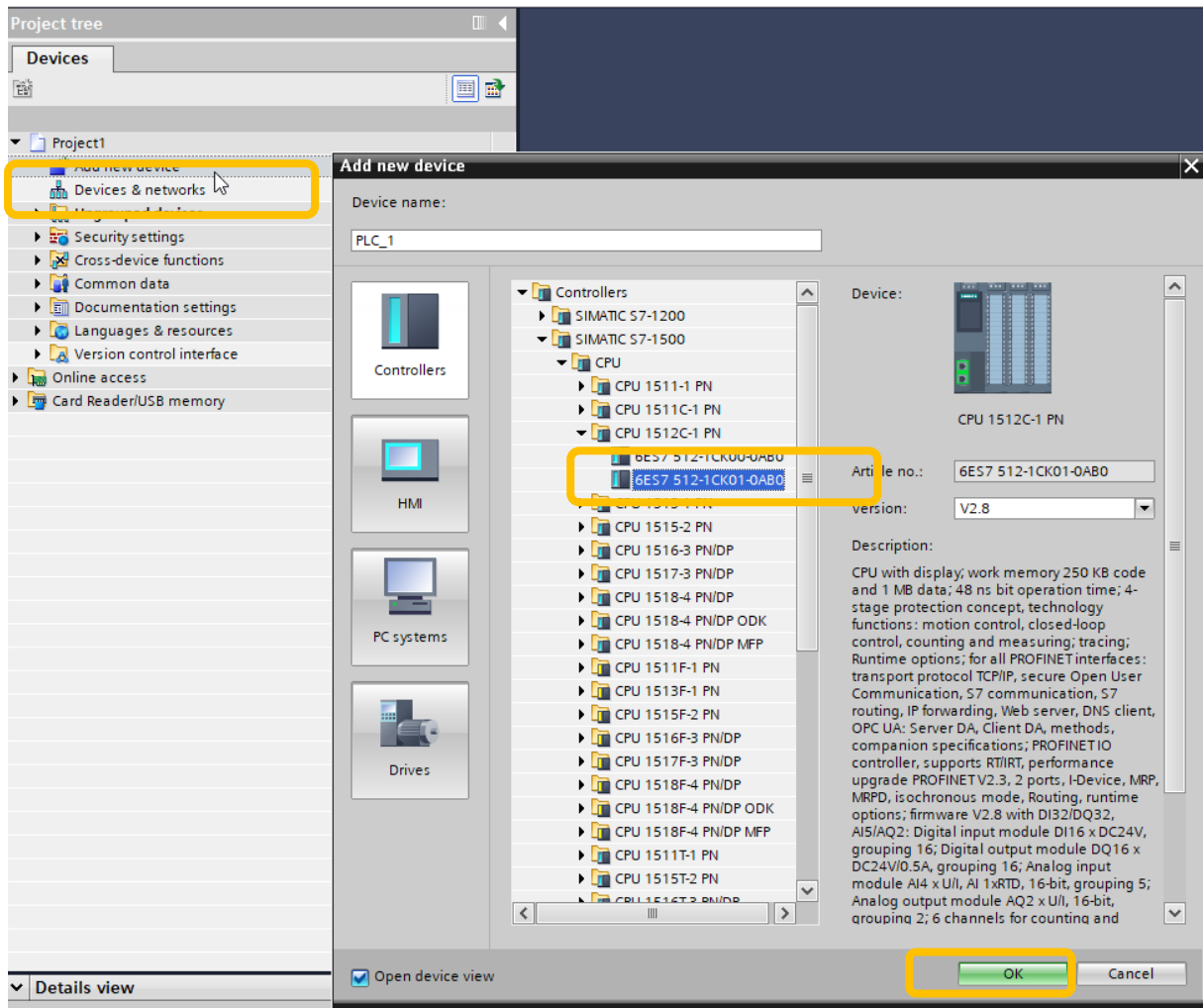


20 Create PLC hardware and integrate PLC variables

Video: 20TIA_new_PLC_Import_Variables

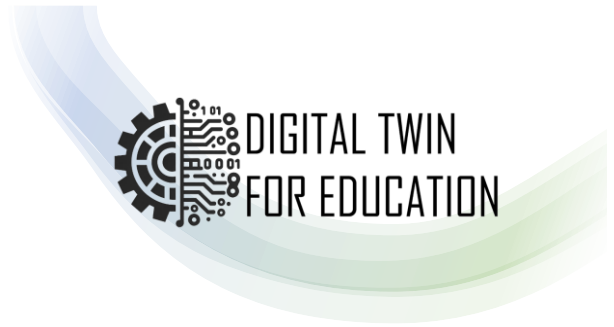
A hardware configuration for a PLC must be defined.

In this case, a 1500 PLC is used, as this corresponds to the reality on site.

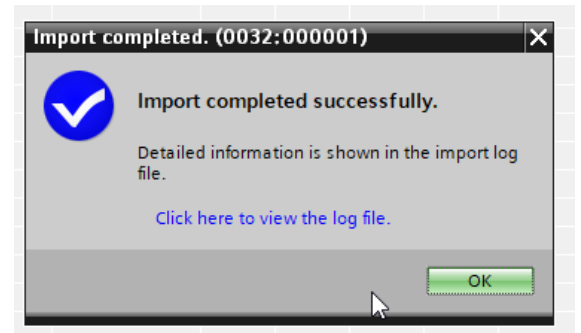
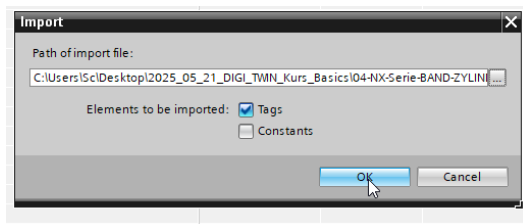
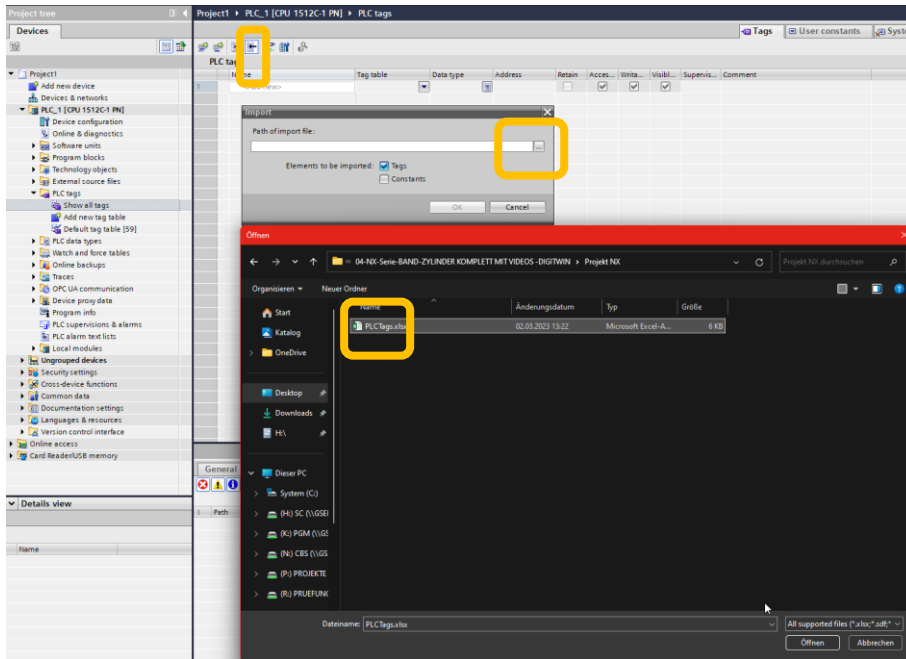


Module 4

Documentation Key Concept Product development



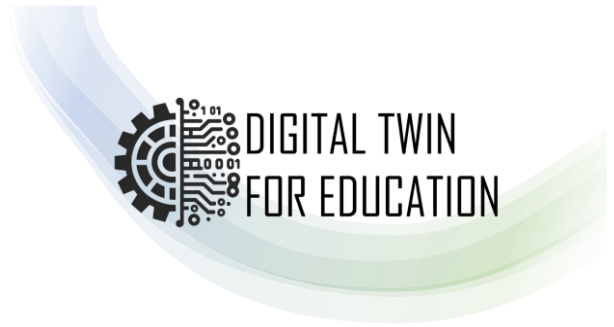
Once the PLC has been configured, the PLC variables (PLC tags) are imported using an Excel file. The file is located in the folder – Appendix → PLC_Tags.xlsx.



Overview of the imported PLC variables.

Module 4

Documentation Key Concept Product development

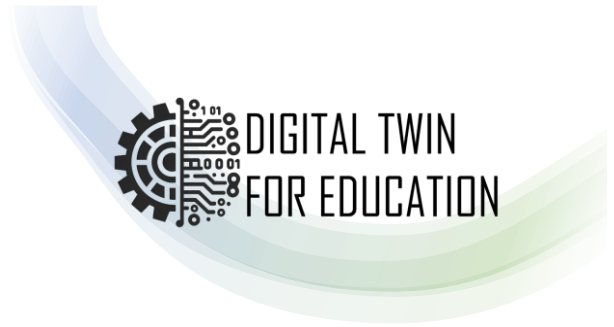


Project tree | Project1 | PLC_1 [CPU 1512C-1 PN] | PLC tags

| | Name | Tag table | Data type | Address | Retain | Access... | Write... |
|----|------|-------------------|-----------|---------|--------------------------|-------------------------------------|-------------------------------------|
| 1 | xMB1 | Variablen-tabelle | Bool | %Q4.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 2 | xMB2 | Variablen-tabelle | Bool | %Q4.1 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 3 | xMB3 | Variablen-tabelle | Bool | %Q4.2 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 4 | xMB4 | Variablen-tabelle | Bool | %Q4.3 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 5 | xMB5 | Variablen-tabelle | Bool | %Q4.4 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 6 | xMB6 | Variablen-tabelle | Bool | %Q4.5 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 7 | xMB7 | Variablen-tabelle | Bool | %Q4.6 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 8 | xMB8 | Variablen-tabelle | Bool | %Q4.7 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 9 | xPF1 | Variablen-tabelle | Bool | %Q5.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 10 | xPF2 | Variablen-tabelle | Bool | %Q5.1 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 11 | xPF3 | Variablen-tabelle | Bool | %Q5.2 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 12 | xPF4 | Variablen-tabelle | Bool | %Q5.3 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 13 | xPF5 | Variablen-tabelle | Bool | %Q5.4 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 14 | xPF6 | Variablen-tabelle | Bool | %Q5.5 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 15 | xPF7 | Variablen-tabelle | Bool | %Q5.6 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 16 | xPF8 | Variablen-tabelle | Bool | %Q5.7 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 17 | xBG1 | Variablen-tabelle | Bool | %I10.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 18 | xBG2 | Variablen-tabelle | Bool | %I10.1 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 19 | xBG3 | Variablen-tabelle | Bool | %I10.2 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 20 | xBG4 | Variablen-tabelle | Bool | %I10.3 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 21 | xBG5 | Variablen-tabelle | Bool | %I10.4 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 22 | xBG6 | Variablen-tabelle | Bool | %I10.5 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 23 | xBG7 | Variablen-tabelle | Bool | %I10.6 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 24 | xBG8 | Variablen-tabelle | Bool | %I10.7 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 25 | xSF1 | Variablen-tabelle | Bool | %I11.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 26 | xSF2 | Variablen-tabelle | Bool | %I11.1 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 27 | xSF3 | Variablen-tabelle | Bool | %I11.2 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 28 | xSF4 | Variablen-tabelle | Bool | %I11.3 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 29 | xSF5 | Variablen-tabelle | Bool | %I11.4 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 30 | xSF6 | Variablen-tabelle | Bool | %I11.5 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

Module 4

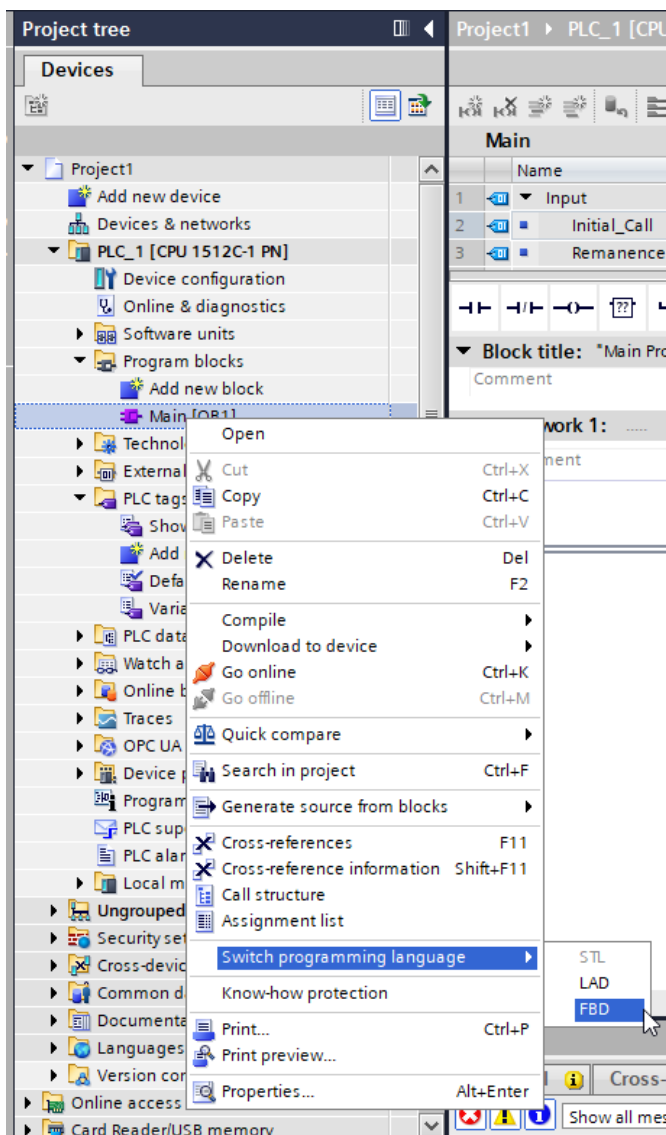
Documentation
Key Concept Product development



21 Creating a function block in the TIA Portal

Video: 21TIA_New_FB

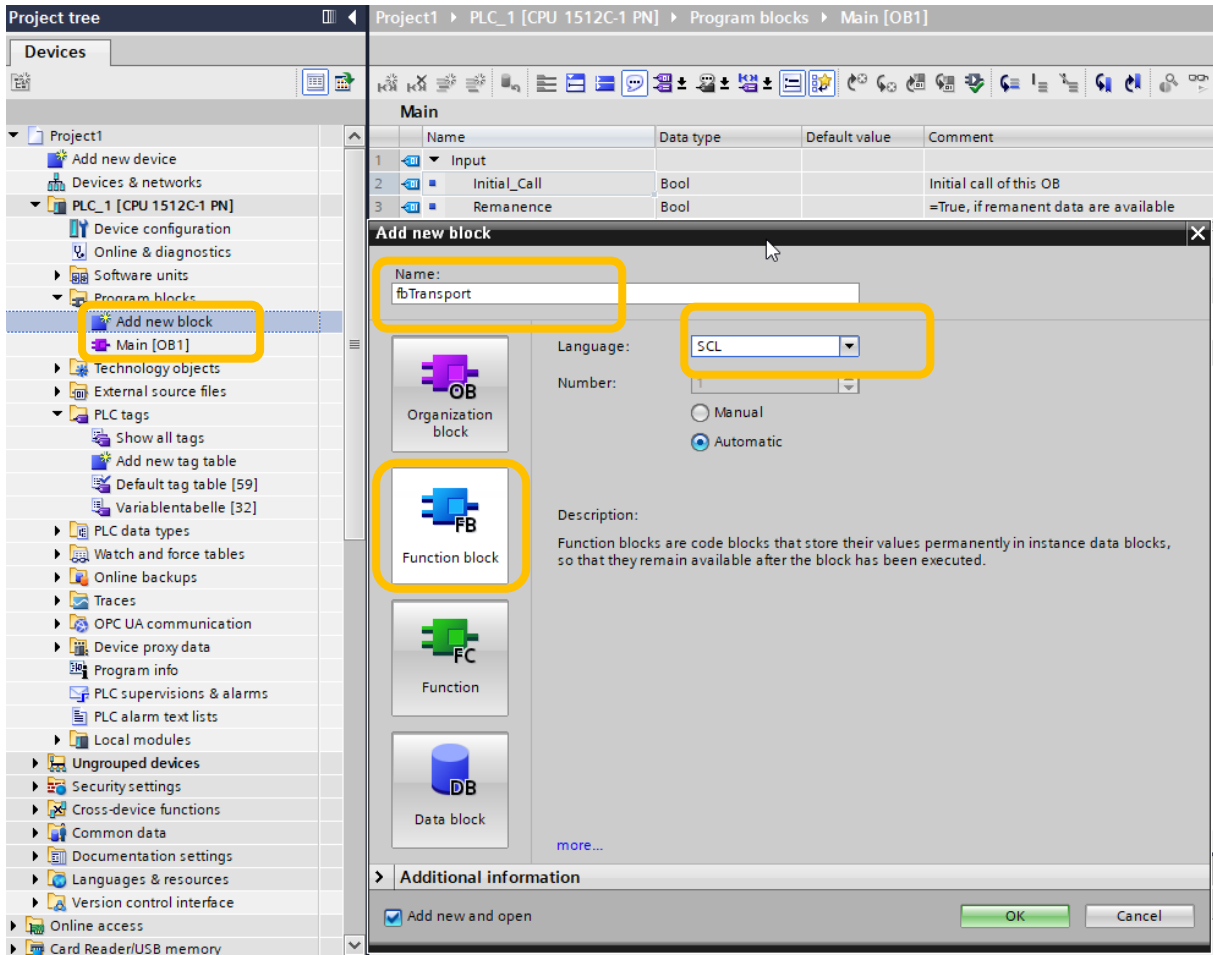
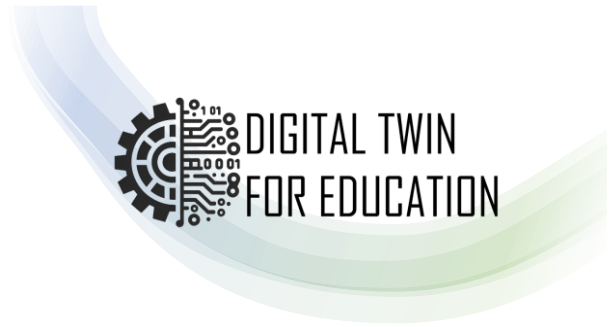
Changing the programming language in the MAIN OB1 block



A new function block is then created using the SCL programming language

Module 4

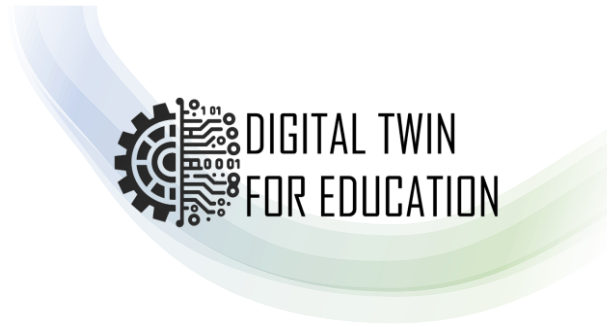
Documentation Key Concept Product development



Please select Function block!

Module 4

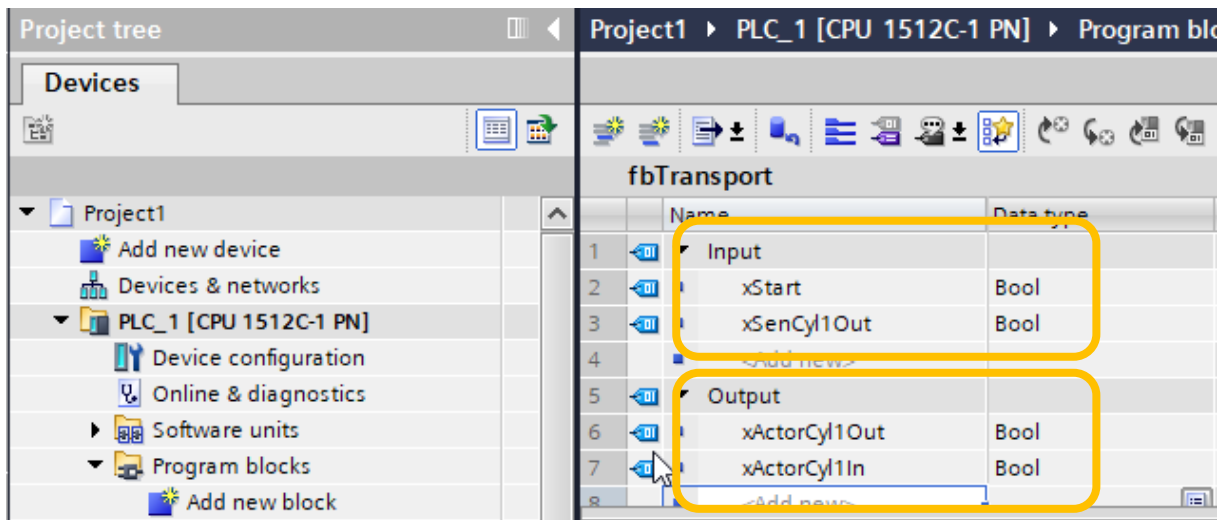
Documentation
Key Concept Product development



22 Program creation in the function block

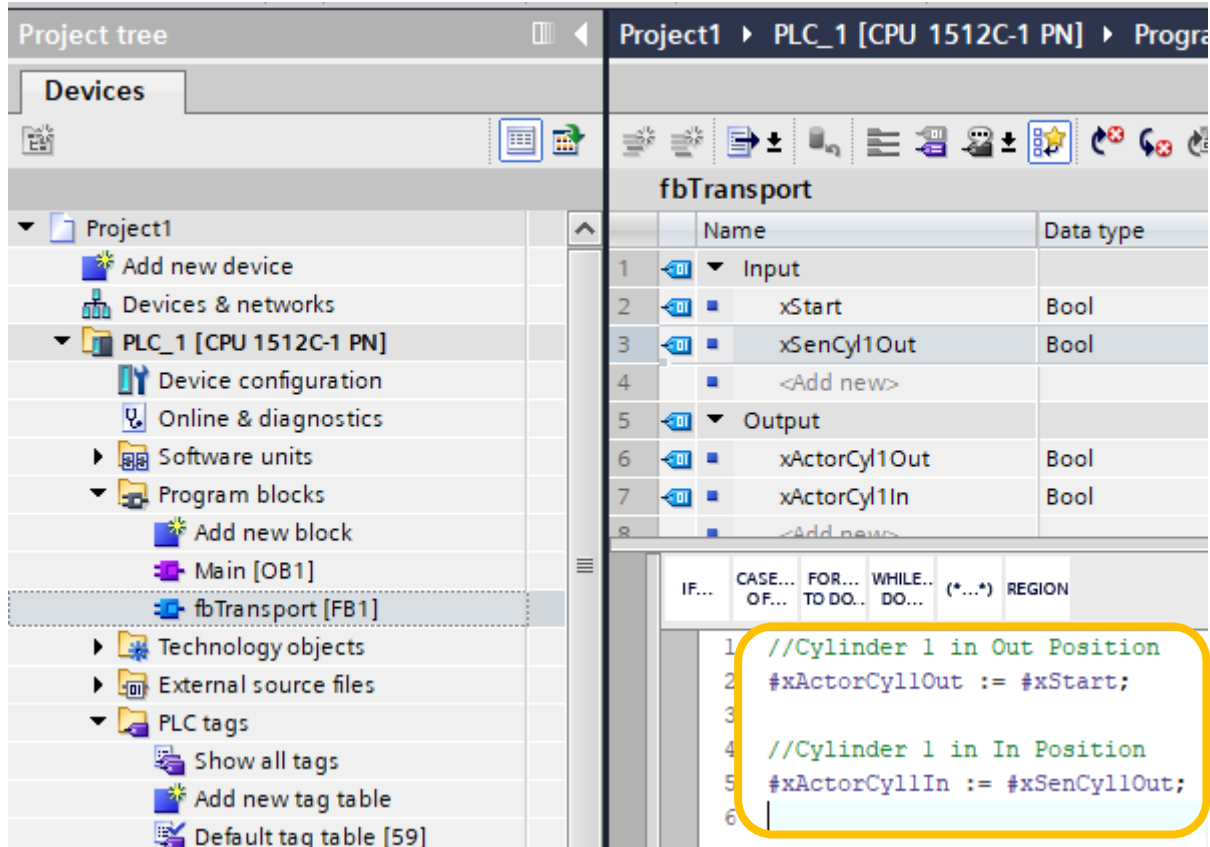
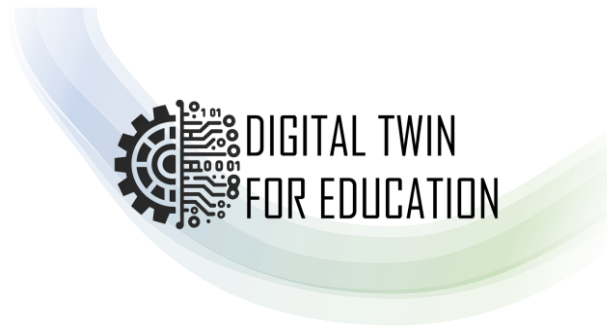
Video: 22TIA_FB_Prog

For the first test, a few input and output variables are created and used to write a small PLC program for cylinder 1.



Module 4

Documentation
Key Concept Product development



The screenshot shows the Siemens SIMATIC Manager interface. On the left is the 'Project tree' with a 'Devices' tab. Under 'Project1', the 'Program blocks' folder is expanded to show 'fbTransport [FB1]'. The main window displays the 'fbTransport' function block definition. A table lists the block's inputs and outputs:

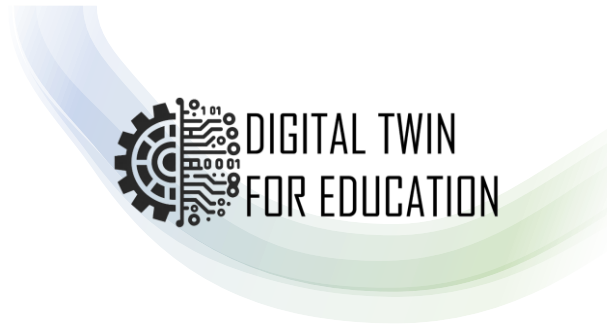
| Name | Data type |
|---------------|-----------|
| Input | |
| xStart | Bool |
| xSenCyl1Out | Bool |
| <Add new> | |
| Output | |
| xActorCyl1Out | Bool |
| xActorCyl1In | Bool |
| <Add new> | |

Below the table, the ladder logic code is shown, with a yellow box highlighting the following lines:

```
1 //Cylinder 1 in Out Position  
2 #xActorCyl1Out := #xStart;  
3  
4 //Cylinder 1 in In Position  
5 #xActorCyl1In := #xSenCyl1Out;  
6
```

Module 4

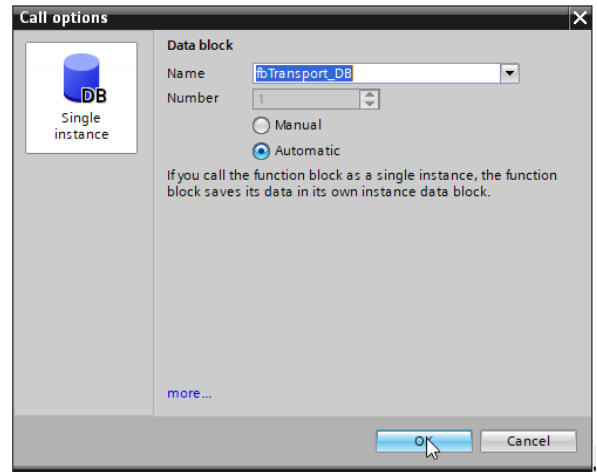
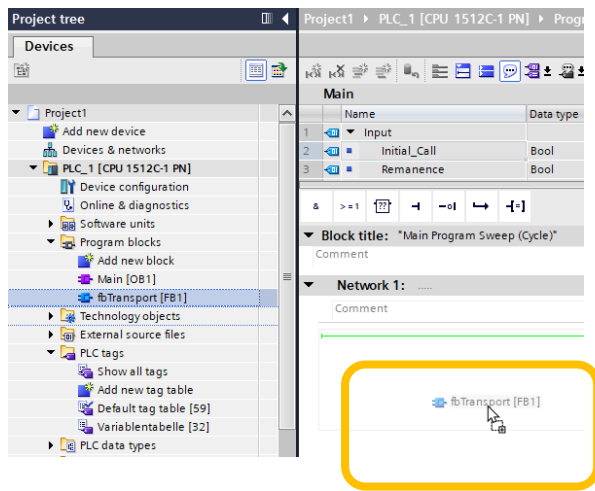
Documentation
Key Concept Product development



23 Call the FB in Main OB1 and save the program

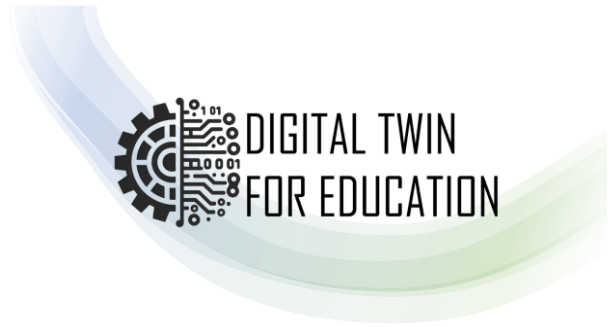
Video: 23TIA_Prog_Save

In MAIN OB1, the variables of the function block are linked to the PLC variables (PLC tags). First, the function block must be dragged and dropped into MAIN OB1.



Module 4

Documentation Key Concept Product development



The screenshot displays the Siemens SIMATIC Manager interface. On the left, the 'Project tree' shows the hierarchy: Project1 > PLC_1 [CPU 1512C-1 PN] > Program blocks > Main [OB1]. Below this, the 'Details view' shows a table of PLC data types:

| Name | Dat... | Details | Comment |
|------|--------|---------|---------|
| xBG1 | Bool | %I10.0 | |
| xBG2 | Bool | %I10.1 | |
| xBG3 | Bool | %I10.2 | |
| xBG4 | Bool | %I10.3 | |
| xBG5 | Bool | %I10.4 | |
| xBG6 | Bool | %I10.5 | |
| xBG7 | Bool | %I10.6 | |
| xBG8 | Bool | %I10.7 | |
| xMB1 | Bool | %Q4.0 | |
| xMB2 | ... | %... | |
| xMB3 | Bool | %Q4.2 | |

The main workspace shows the 'Main' program block configuration. It includes an 'Input' table:

| Name | Data type | Default value | Comment |
|--------------|-----------|---------------|------------|
| Initial_Call | Bool | | Initial ca |
| Remanence | Bool | | =True, if |

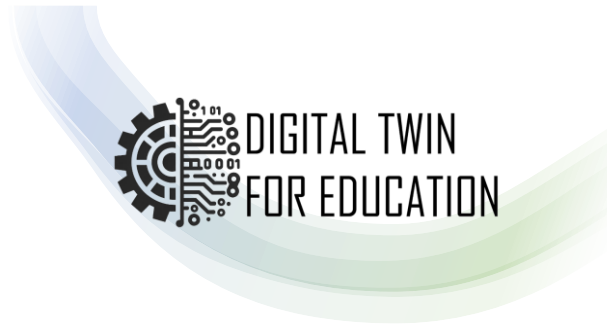
Below the table, the 'Block title' is 'Main Program Sweep (Cycle)'. The 'Network 1' section shows a ladder logic diagram with a function block 'fbTransport' (FB1) and a data block 'fbTransport_DB' (DB1). The diagram includes the following elements:

- Inputs: xSF1 (I11.0), xBG2 (I10.1), xSenCyl1Out.
- Outputs: xActorCyl1Out (Q4.1), xActorCyl1In (Q4.0).
- EN (Enable) and ENO (Enable Out) terminals.

At the bottom, there are tabs for 'General', 'Cross-references', 'Compile', and 'Syntax'.

Module 4

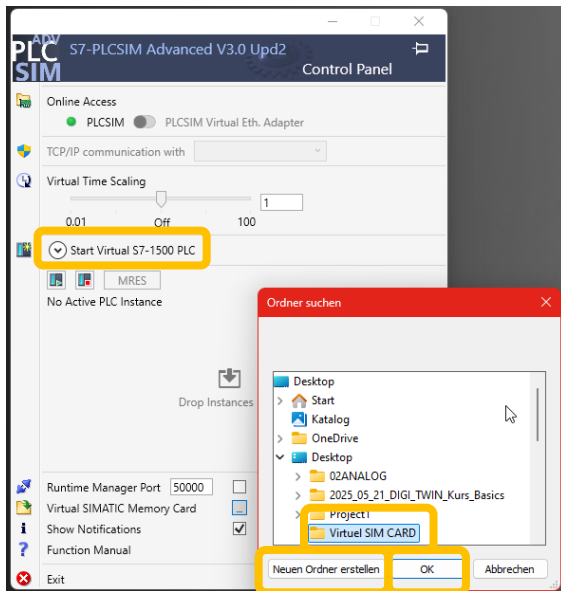
Documentation Key Concept Product development



24 Creating a new simulation instance in PLCSim Advanced

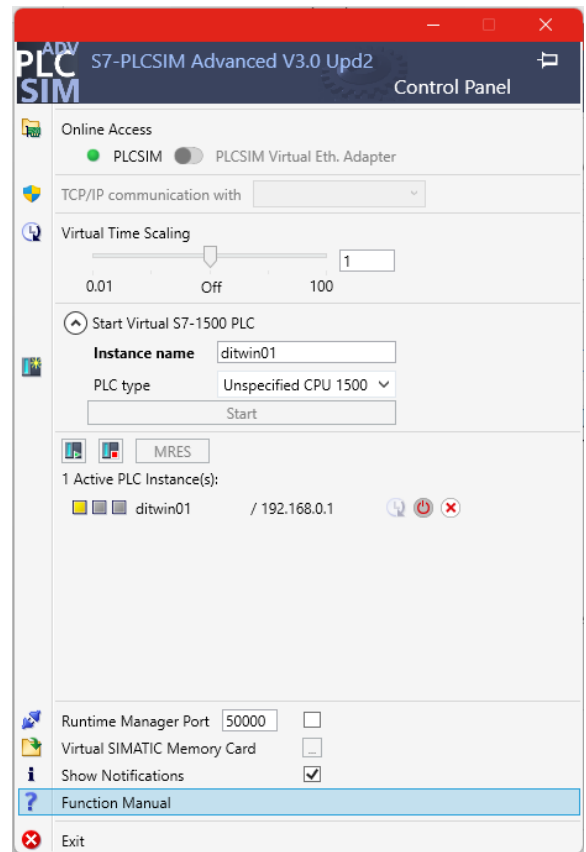
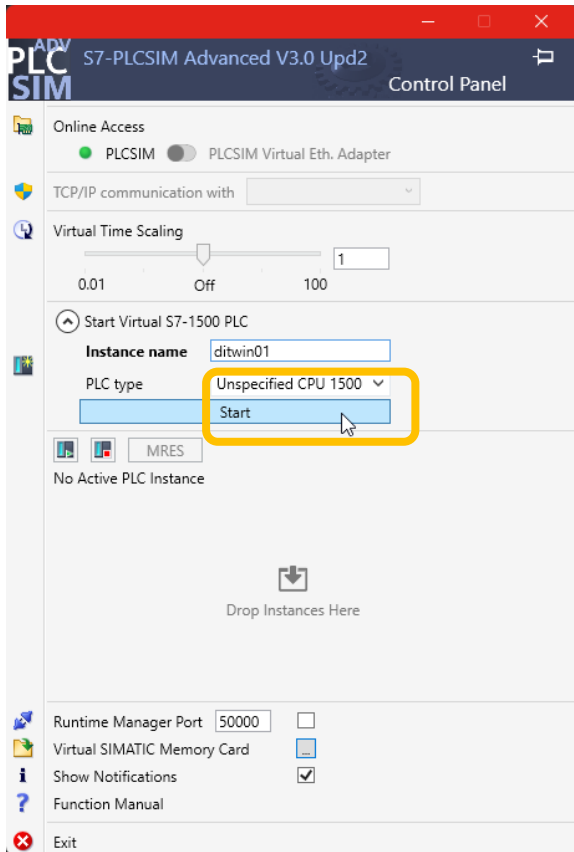
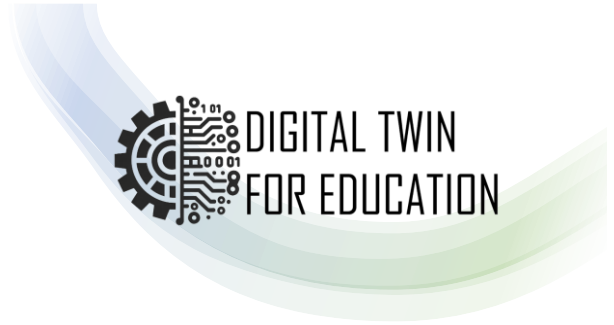
Video: [24_Sim_New_Inst](#)

A virtual soft PLC can be created by starting the PLC Sim Advanced software. This controller, named ditwin01, is used to load the current PLC.



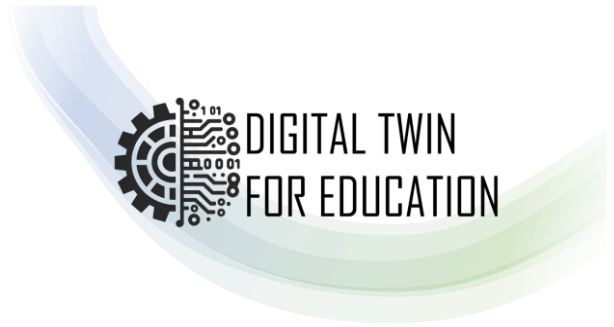
Module 4

Documentation
Key Concept Product development



Module 4

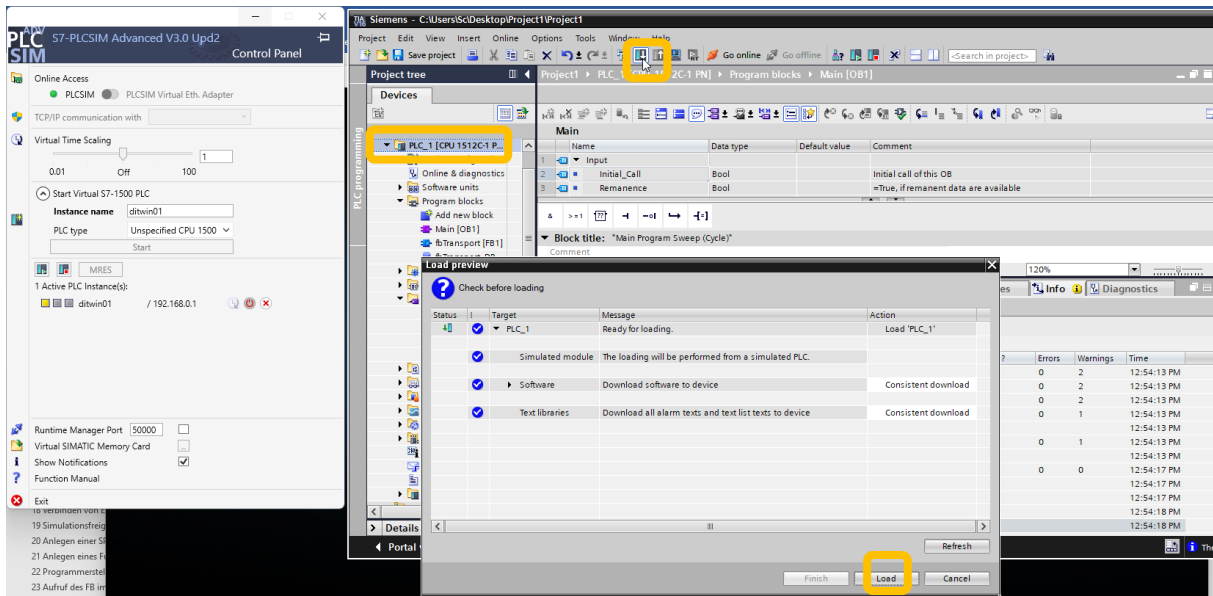
Documentation
Key Concept Product development



25 Load TIA program in PLC Sim

Video: 25_TIA_Sim_Transfer

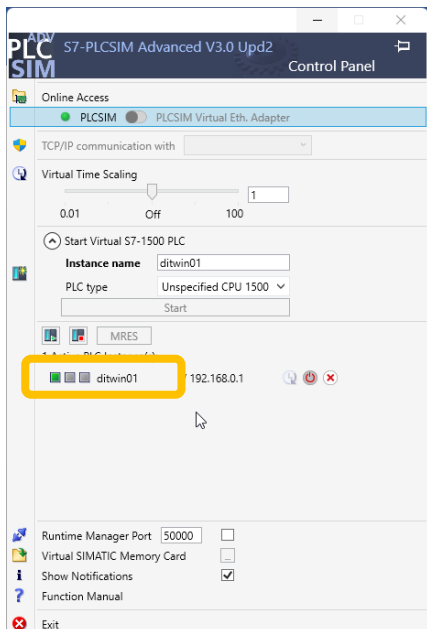
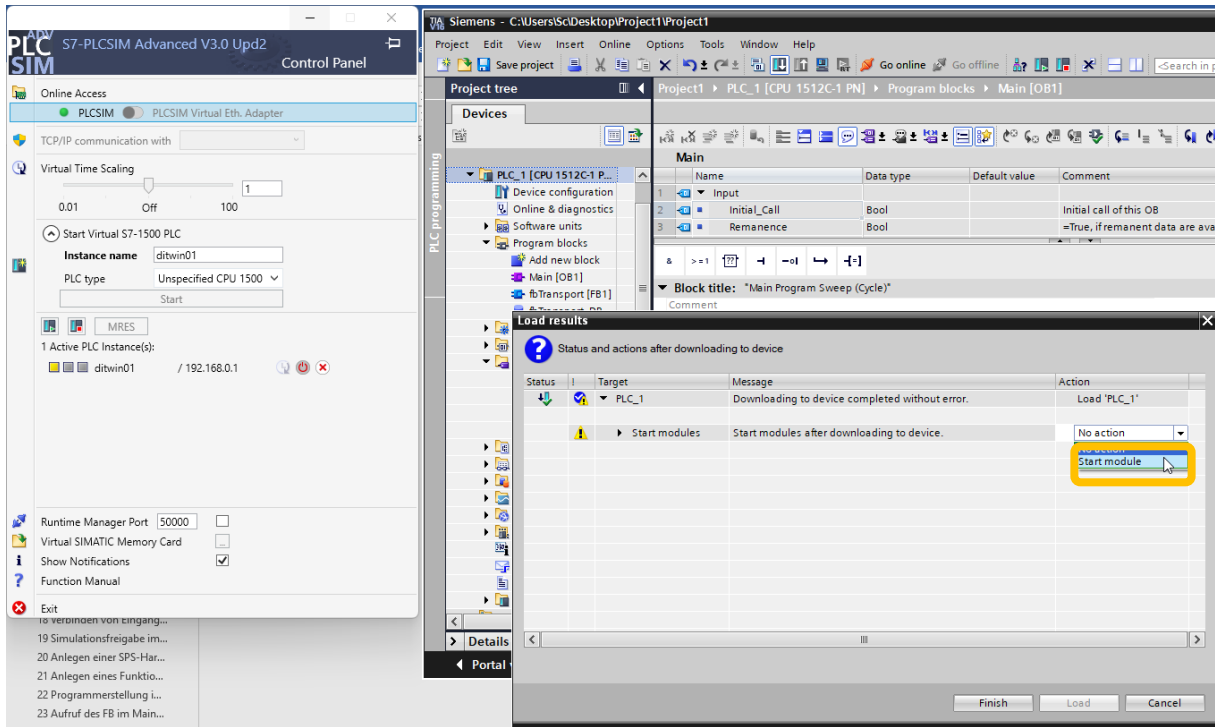
The current PLC program can now be loaded into PLC SIM Advanced.



When the module is started, the PLC changes its status from Stop to Start and the LED turns green.

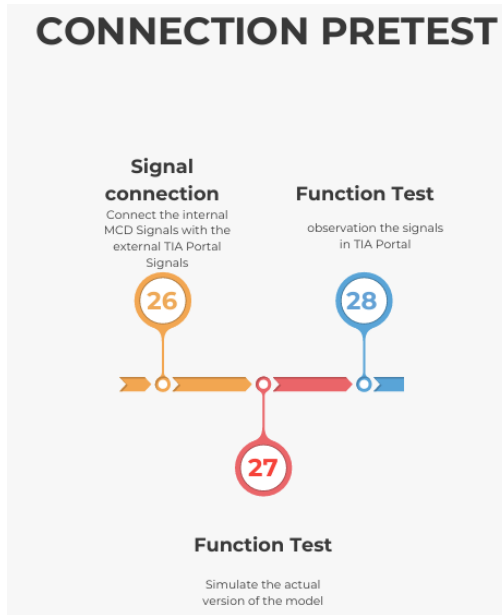
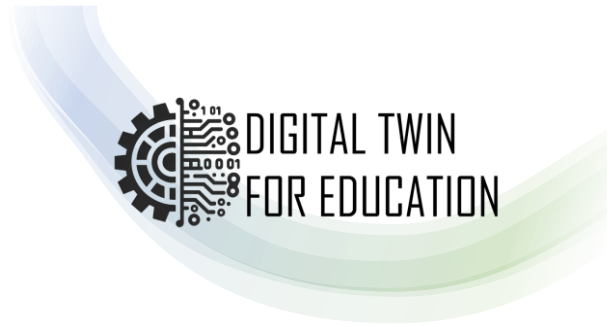
Module 4

Documentation Key Concept Product development



Module 4

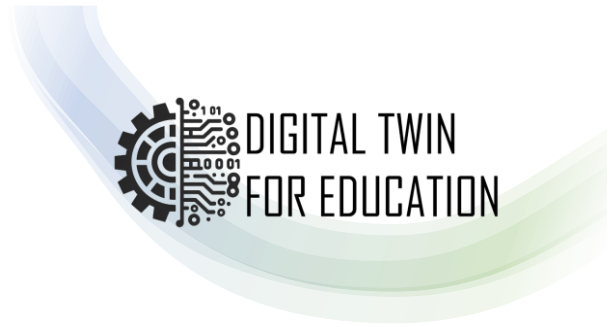
Documentation
Key Concept Product development



In this small section we are making some adjustments in NX and we run the test program.

Module 4

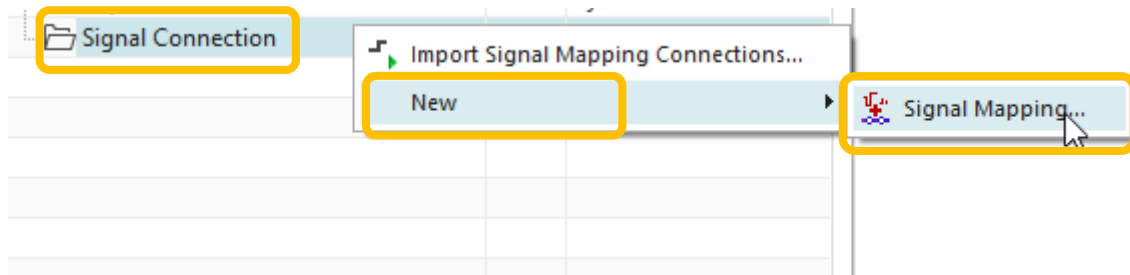
Documentation
Key Concept Product development



26 Creating signal connections in MCD

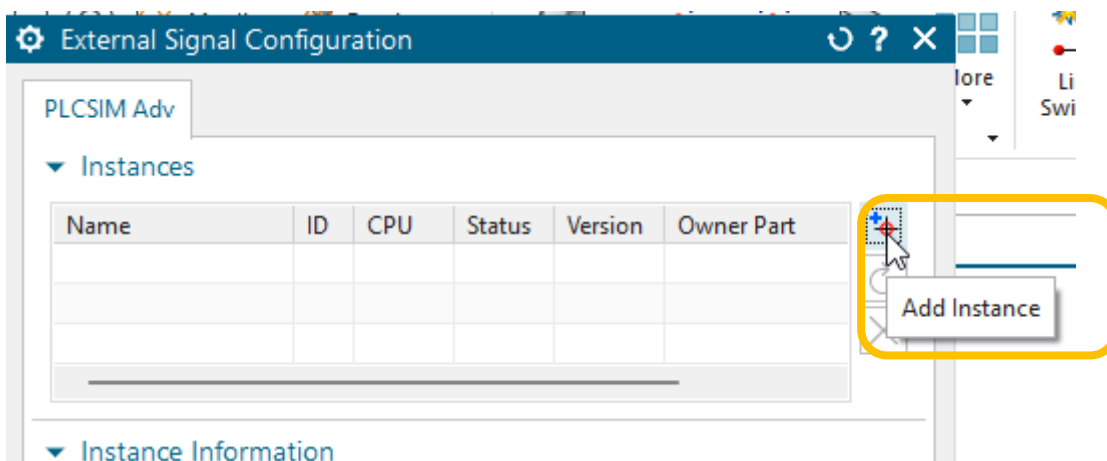
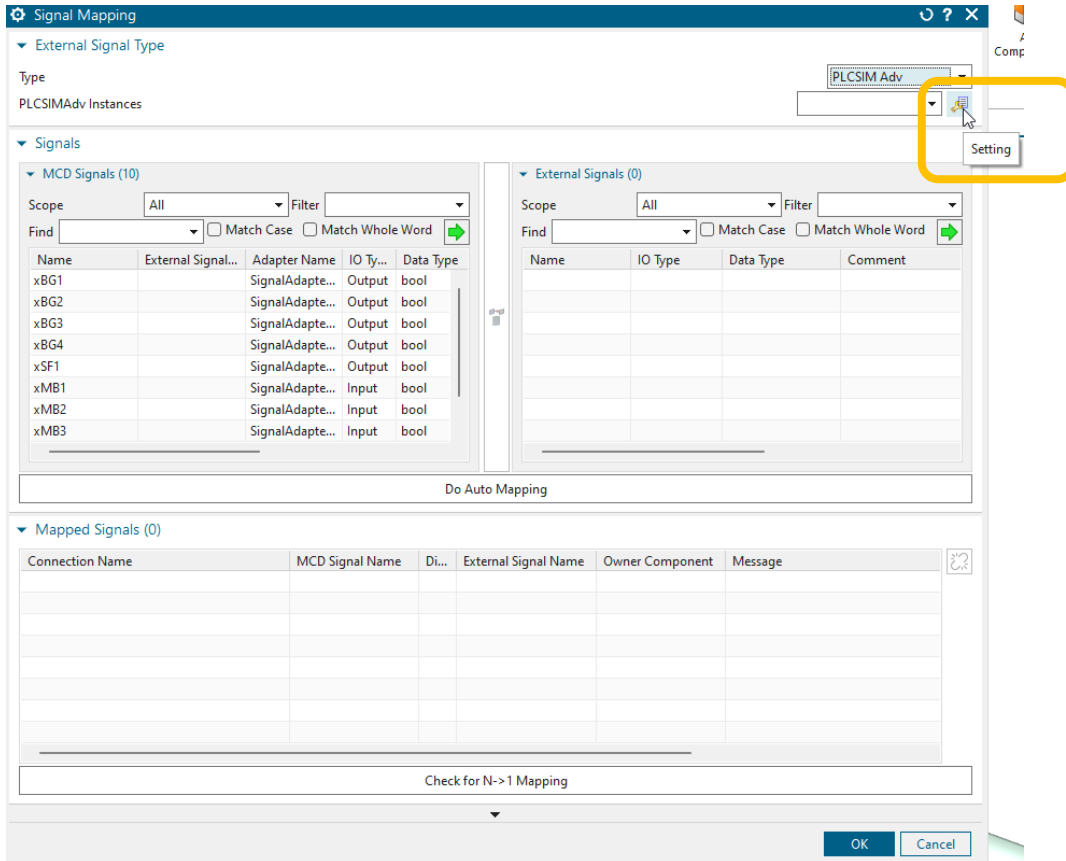
Video: 26_NX_New_Signal_Conn

The simulation level is then ready for operation and the signal connections in Siemens NX MCD must be created.



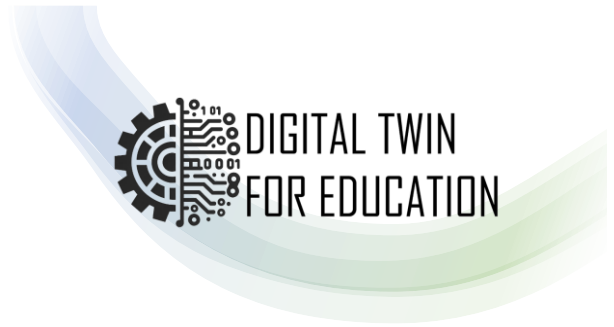
Module 4

Documentation
Key Concept Product development

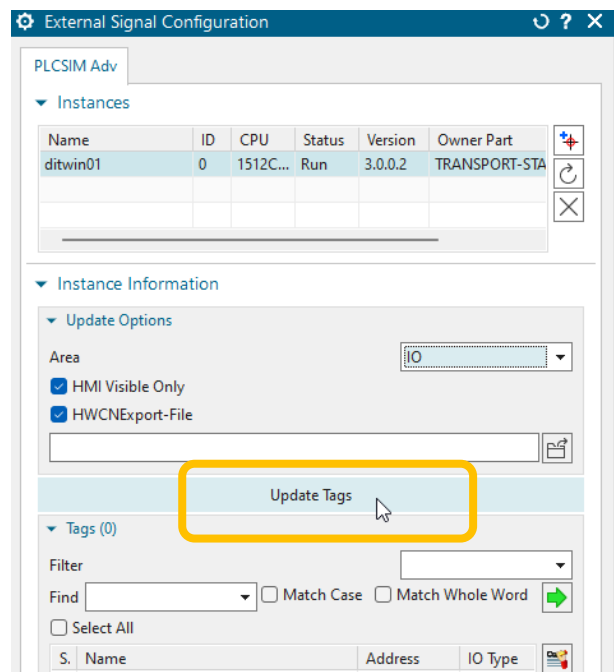
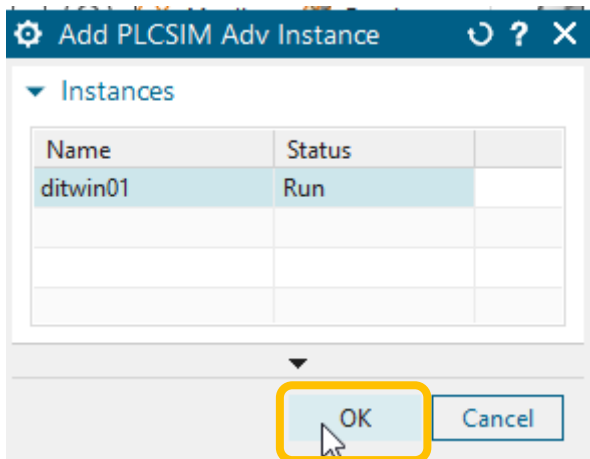


Module 4

Documentation
Key Concept Product development

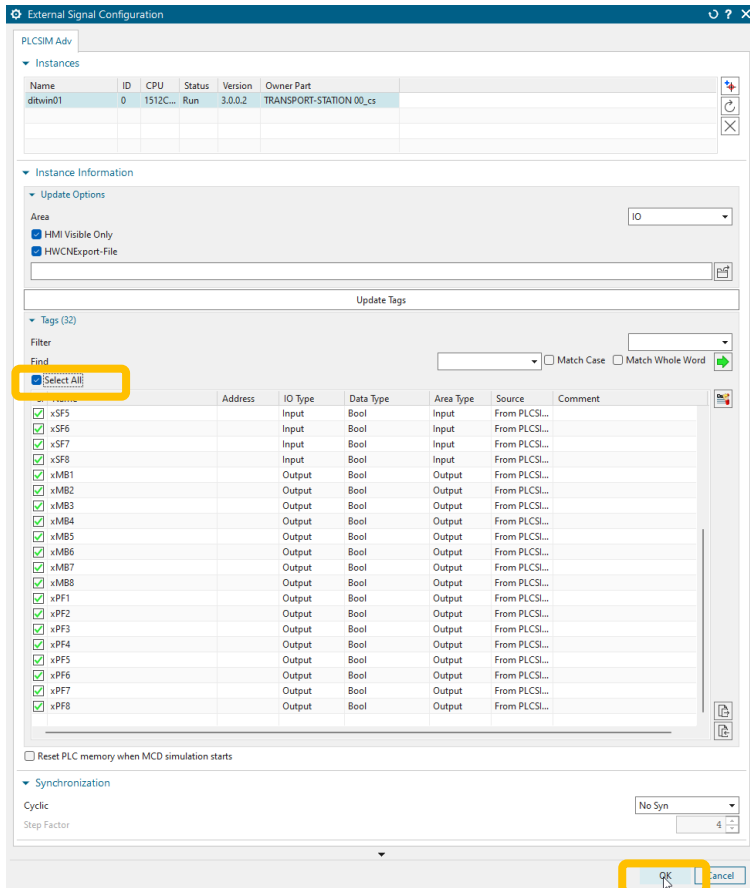
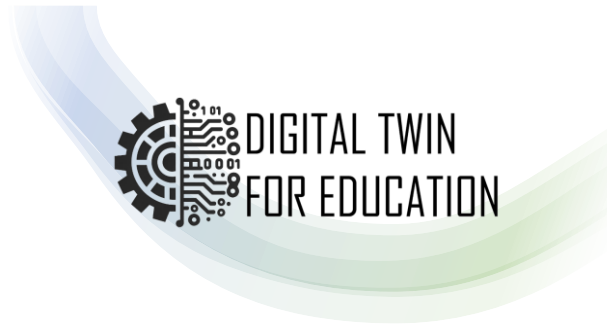


In the signal connections, the soft PLC from PLCSIM Advanced is created as an instance and linked.



Module 4

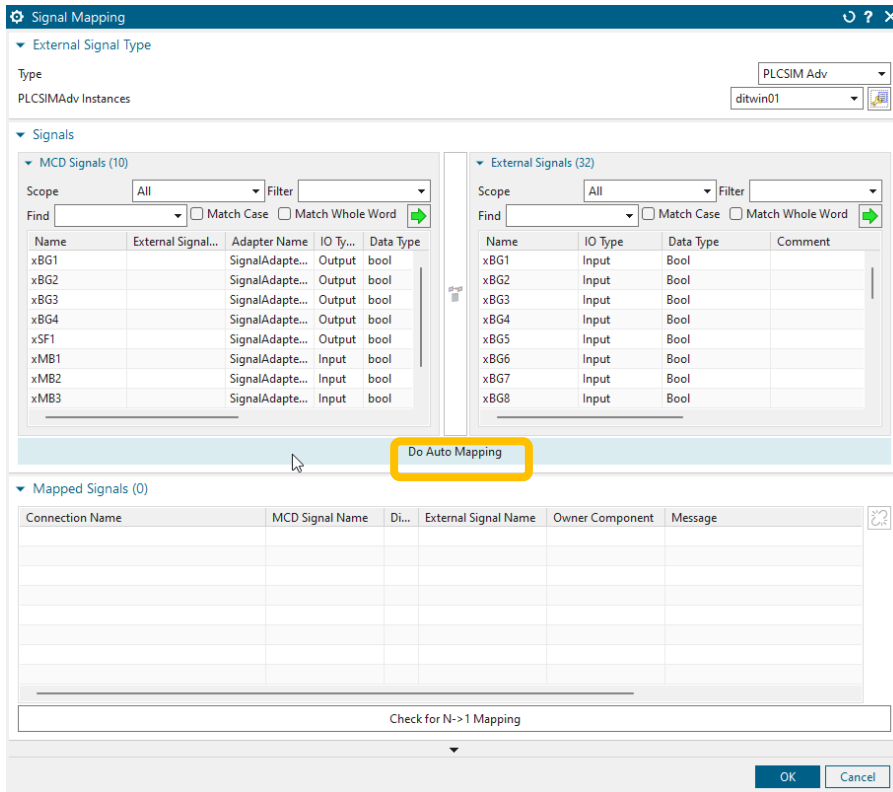
Documentation Key Concept Product development



Select all variables and then select automatic assignment. The automatic assignment can be performed successfully thanks to the identical naming.

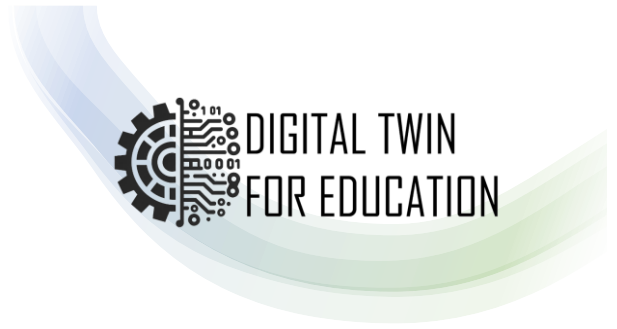
Module 4

Documentation Key Concept Product development



Module 4

Documentation
Key Concept Product development



27 Temporary storage and testing of the software status in NX

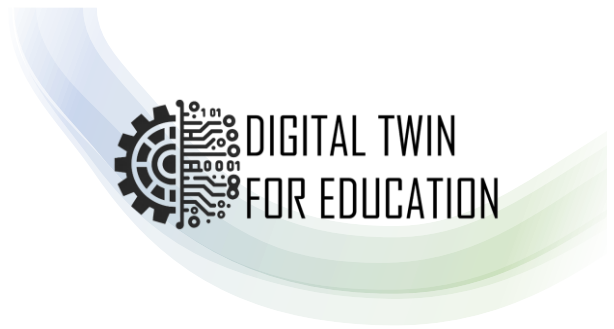
Video: 27_NX_Save_Test

Video: 27_TIA_Observe

Testing and observation allow the current status to be simulated and checked.

Module 4

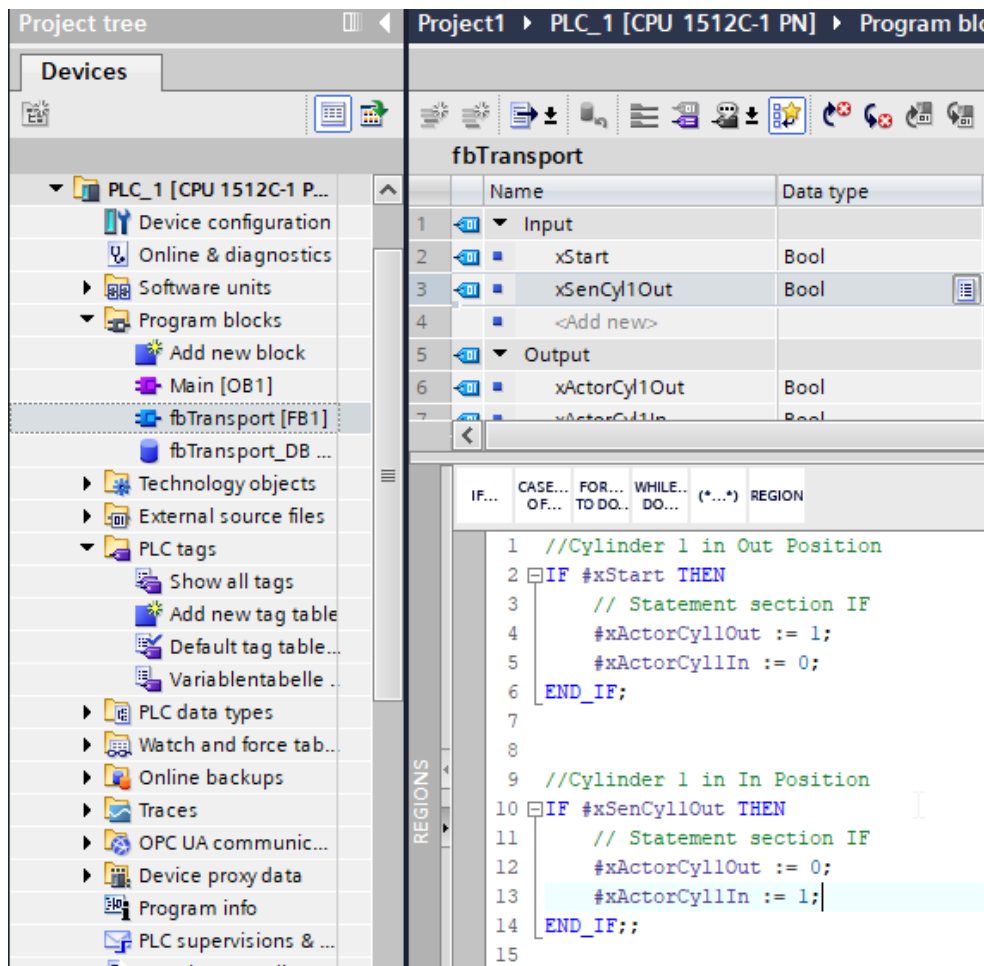
Documentation
Key Concept Product development



28 Testing the software status in the TIA Portal

Video: 27_TIA_Test

The current function block is tested in conjunction with NX MCD and PLCSIM. The variables can also be monitored in observation mode.



The screenshot displays the Siemens TIA Portal interface. On the left, the 'Project tree' shows the 'fbTransport [FB1]' block selected under 'Program blocks'. The main window shows the configuration for 'fbTransport' with a table of inputs and outputs:

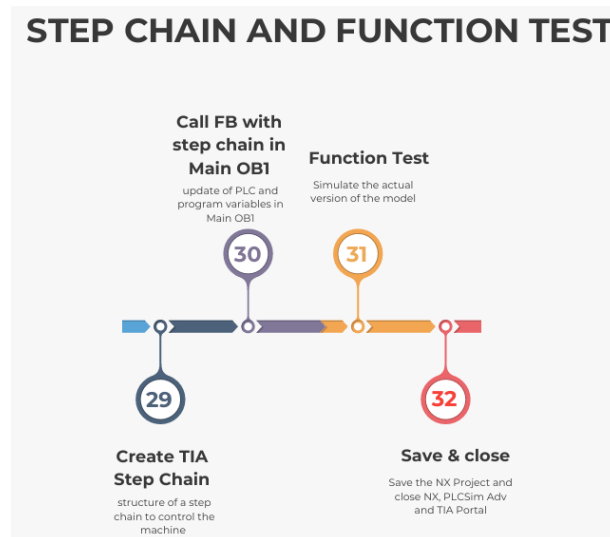
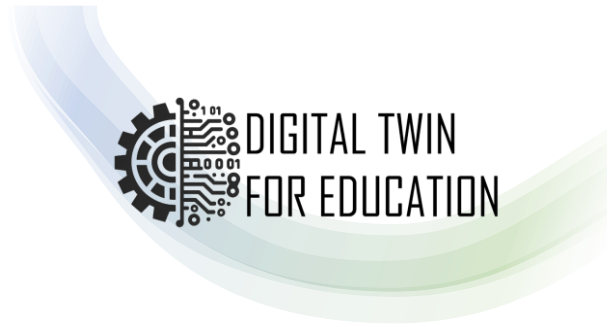
| Name | Data type |
|-----------------|-----------|
| 1 Input | |
| 2 xStart | Bool |
| 3 xSenCyl1Out | Bool |
| 4 <Add new> | |
| 5 Output | |
| 6 xActorCyl1Out | Bool |
| 7 xActorCyl1In | Bool |

Below the table, the ladder logic code is visible:

```
1 //Cylinder 1 in Out Position
2 IF #xStart THEN
3   // Statement section IF
4   #xActorCyl1Out := 1;
5   #xActorCyl1In := 0;
6 END_IF;
7
8
9 //Cylinder 1 in In Position
10 IF #xSenCyl1Out THEN
11   // Statement section IF
12   #xActorCyl1Out := 0;
13   #xActorCyl1In := 1;
14 END_IF;;
15
```

Module 4

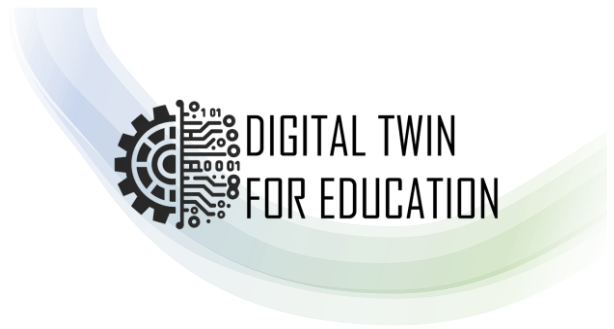
Documentation
Key Concept Product development



Now we are able to write a „step- chain“ program in TIA Portal and test it with our digital twin model in NX MCD.

Module 4

Documentation Key Concept Product development



29 Creation of a step chain program

Video: 29_TIA_Step_Chain

In the next step, the small PLC program for cylinder 1 will be overwritten and replaced by a more comprehensive control program with a step chain.

To do this, new variables must be implemented in the function block in the inputs and outputs area. In addition, the step number with the designation *istep*, which has the default value 0, is added to the static variables area.

| fbTransport | | | | | | | | |
|-------------|---------------|-----------|---------------|------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|
| Area | Name | Data type | Default value | Retain | Accessible f... | Writa... | Visible in ... | Setpoint |
| Input | xStart | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | xSenCyl1Out | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xSenCyl1In | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xSenCyl2Out | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xSenCyl2In | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| Output | xActorCyl1Out | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xActorCyl1In | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xActorCyl2Out | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xActorCyl2In | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | xConveyorOn | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| Static | istep | Int | 0 | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

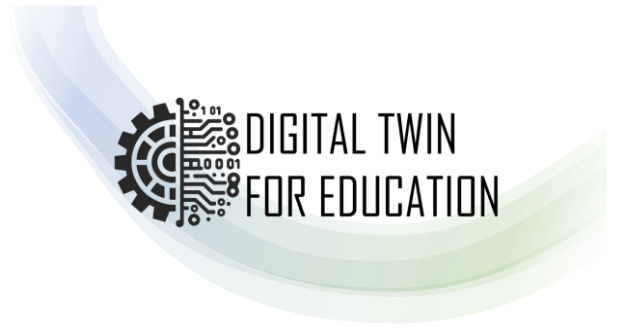
A possible implementation of the step chain in SCL could look like the screenshot below.

```
IF... CASE... FOR... WHILE... (*...*) REGION
OF... TO DO... DO...

1 CASE #istep OF
2 0:
3 IF #xStart THEN // Statement section case 1
4 #istep := #istep + 1;
5 END_IF;
6 1:
7 IF #xSenCyl1Out THEN // Statement section case 1
8 #istep := #istep + 1;
9 END_IF;
10 2:
11 IF #xSenCyl1In THEN // Statement section case 1
12 #istep := #istep + 1;
13 END_IF;
14 3:
15 IF #xSenCyl2Out THEN // Statement section case 1
16 #istep := #istep + 1;
17 END_IF;
18 4:
19 IF #xSenCyl2In THEN // Statement section case 1
20 #istep := 0;
21 END_IF;
22
23 END_CASE;
```

Module 4

Documentation
Key Concept Product development



30 Control of the outputs of the step chain program

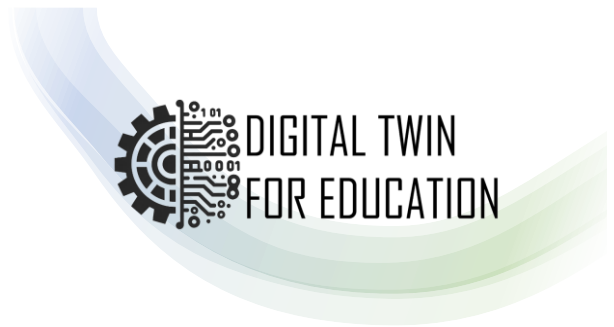
Video: 30_TIA_New_Prog_Chain

After the advance conditions were defined in the previous step, the control of the outputs is now defined in the program.

To do this, only the step numbers are assigned to the individual outputs and connected using a basic logical OR (OR) link.

Module 4

Documentation
Key Concept Product development



fbTransport

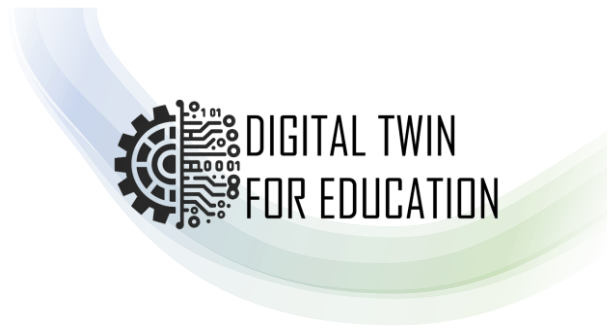
| Name | Data type | Default value | Retain | Accessible f... | Writa... V |
|-------------|-----------|---------------|------------|-------------------------------------|-------------------------------------|
| Input | | | | <input type="checkbox"/> | <input type="checkbox"/> |
| xStart | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| xSenCyl1Out | Bool | false | Non-retain | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

```
IF... CASE... FOR... WHILE... (*...*) REGION
OF... TO DO... DO...

1 CASE #iStep OF
2   0:
3     IF #xStart THEN // Statement section case 1
4       #iStep := #iStep + 1;
5     END_IF ;
6   1:
7     IF #xSenCyl1Out THEN // Statement section case 1
8       #iStep := #iStep + 1;
9     END_IF;
10  2:
11    IF #xSenCyl1In THEN // Statement section case 1
12      #iStep := #iStep + 1;
13    END_IF;
14  3:
15    IF #xSenCyl2Out THEN // Statement section case 1
16      #iStep := #iStep + 1;
17    END_IF;
18  4:
19    IF #xSenCyl2In THEN // Statement section case 1
20      #iStep := 0;
21    END_IF;
22
23 END_CASE;
24
25 //Controlling the outputs
26 #xActorCyl1Out := (#iStep = 1);
27 #xActorCyl1In := (#iStep = 0) OR (#iStep = 2) OR (#iStep = 3) OR (#iStep = 4);
28 #xActorCyl2Out := (#iStep = 3);
29 #xActorCyl2In := (#iStep = 0) OR (#iStep = 1) OR (#iStep = 2) OR (#iStep = 4);
30 #xConveyorOn := (#iStep = 4);
31
```

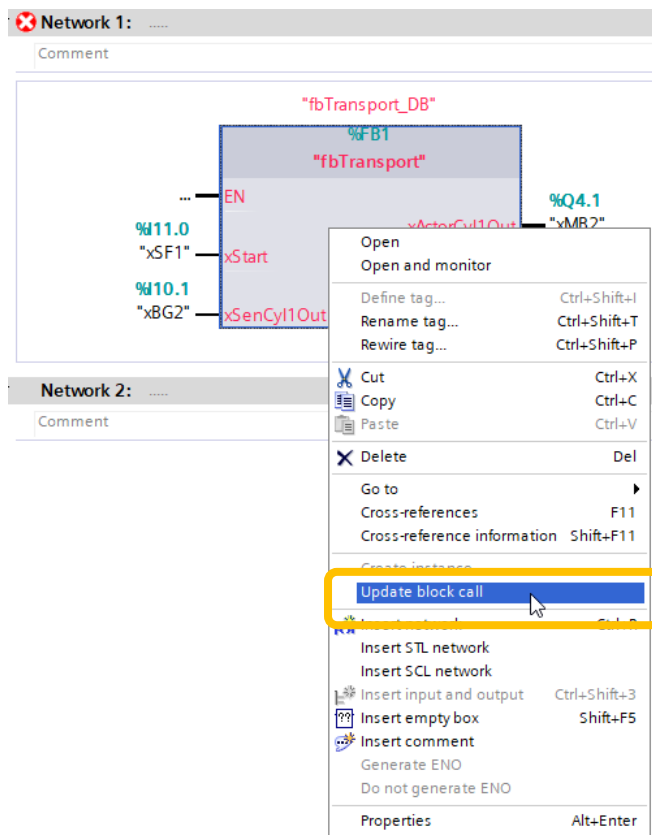
Module 4

Documentation Key Concept Product development



The adjustments in the function block also require adjustments to be made in the MAIN OB1 block with regard to variable assignment.

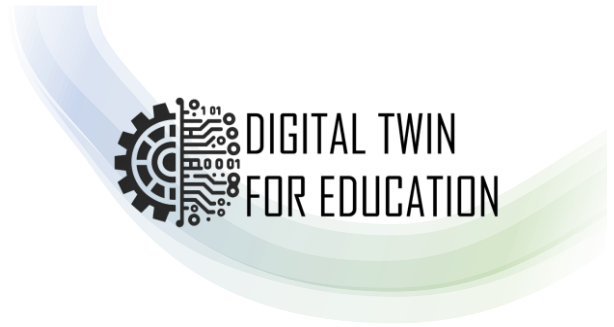
To do this, the content in MAIN OB1 must first be updated so that the PLC variables can then be reassigned.



Simply click on Show all tags once and assign them to MAIN OB1 using drag & drop.

Module 4

Documentation Key Concept Product development



The screenshot displays the SIMATIC Manager interface. On the left, the 'PLC tags' list is visible, with a yellow box highlighting the 'Show all tags' button and a table of tags. The main area shows 'Network 1' of a ladder logic program. A function block 'fbTransport' is connected to various inputs and outputs. A yellow box highlights the input side of the block, and another yellow box highlights the output side.

| Name | DataType | Address |
|------|----------|---------|
| xBG1 | Bool | %I10.0 |
| xBG2 | Bool | %I10.1 |
| xBG3 | Bool | %I10.2 |
| xBG4 | Bool | %I10.3 |
| xBG5 | Bool | %I10.4 |
| xBG6 | Bool | %I10.5 |
| xBG7 | Bool | %I10.6 |
| xBG8 | Bool | %I10.7 |
| xMB1 | Bool | %Q4.0 |
| xMB2 | Bool | %Q4.1 |
| xMB3 | Bool | %Q4.2 |
| xMB4 | Bool | %Q4.3 |
| xMB5 | Bool | %Q4.4 |
| xPF1 | Bool | %Q5.0 |

Network 1:

Block: "fbTransport" (FB1)

Inputs (Left side):

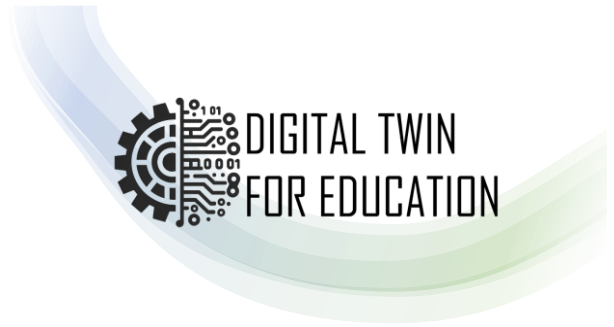
- xStart: %I11.0 ("xSF1")
- xSenCyl1Out: %I10.1 ("xBG2")
- xSenCyl1In: %I10.0 ("xBG1")
- xSenCyl2Out: %I10.3 ("xBG4")
- xSenCyl2In: %I10.2 ("xBG3")

Outputs (Right side):

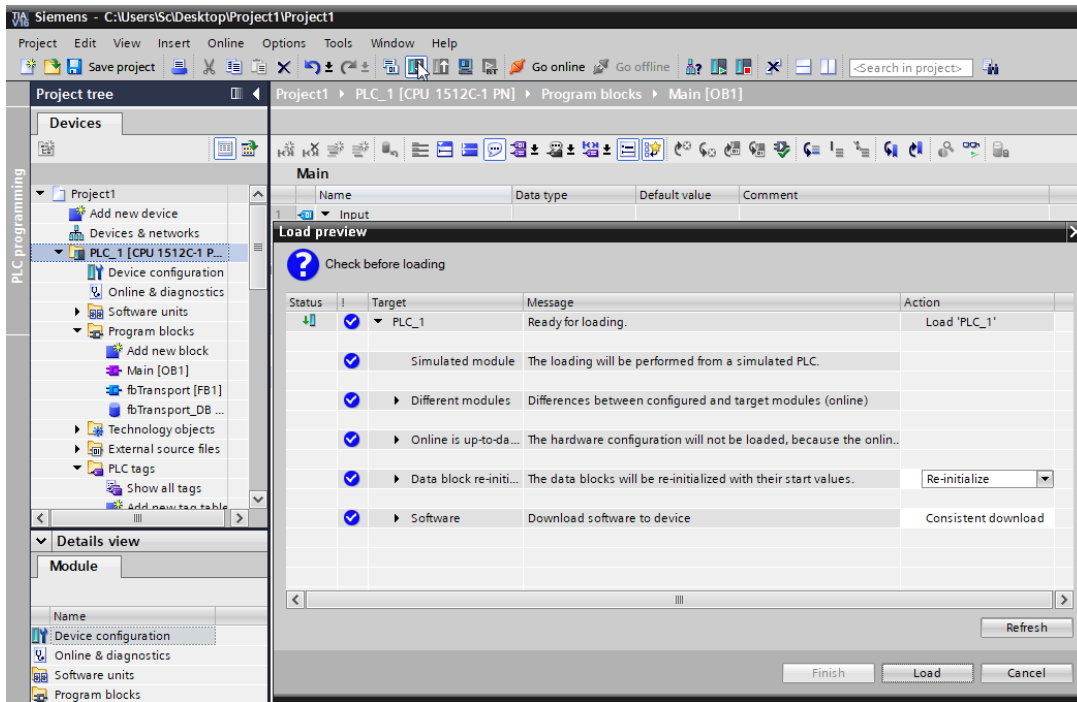
- xActorCyl1Out: %Q4.1 ("xMB2")
- xActorCyl1In: %Q4.0 ("xMB1")
- xActorCyl2Out: %Q4.2 ("xMB3")
- xActorCyl2In: %Q4.3 ("xMB4")
- xConveyorOn: %Q4.4 ("xMB5")

Module 4

Documentation Key Concept Product development

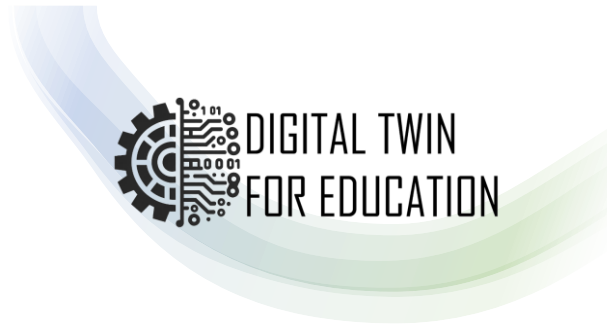


The current software version can then be reloaded from the TIA Portal into the soft PLC in PLCSIM Advanced.



Module 4

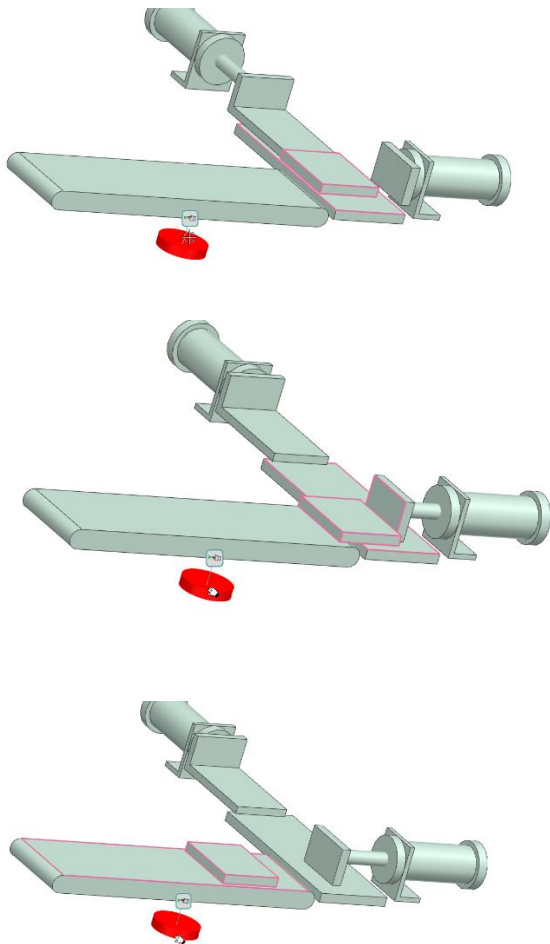
Documentation
Key Concept Product development



31 Testing the software version with NX

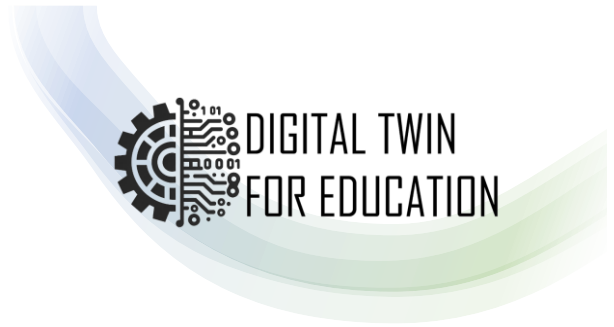
Video: 31_NX_Prog_Chain_Test

Finally, the newly programmed step chain can be tested with the NX model. To do this, first start the simulation (RUN) and press the Start button.



Module 4

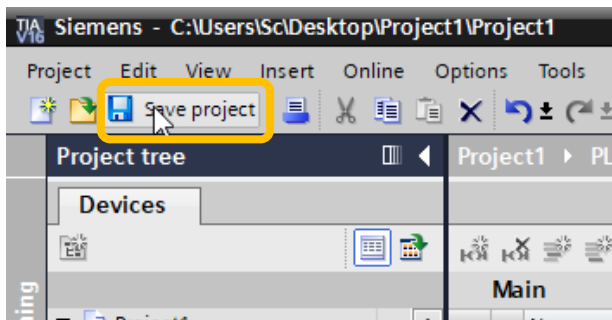
Documentation
Key Concept Product development



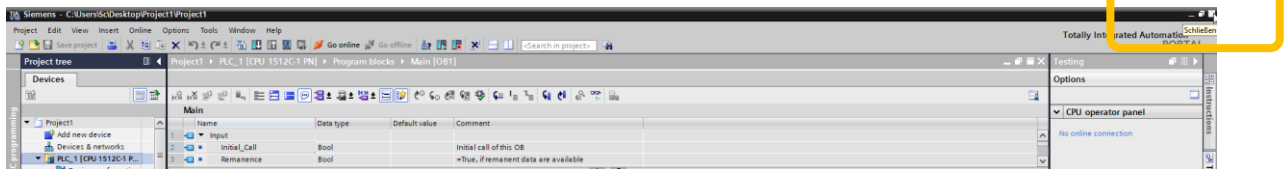
32 Save status and close entire project

Video: 32_Save_Close_All

Finally, all project statuses should be saved so that variations and adjustments can be made afterwards.



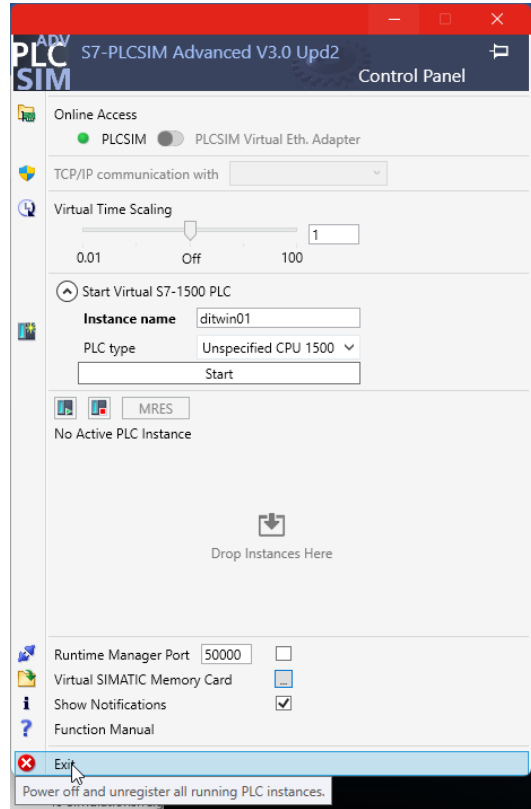
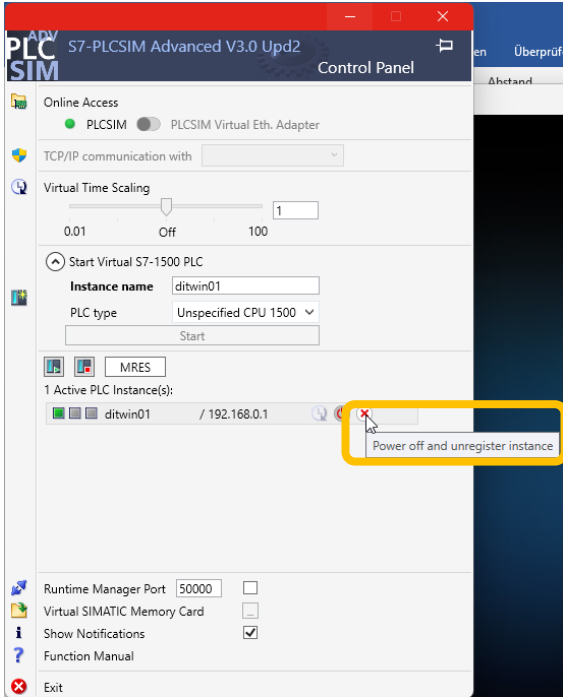
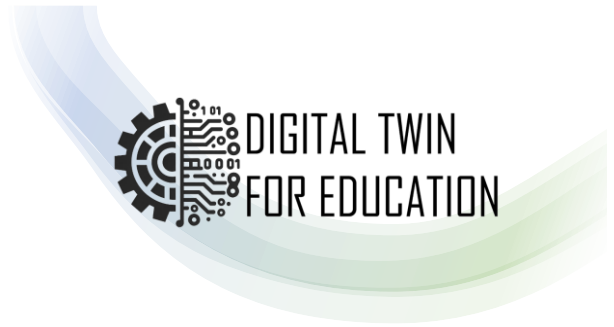
The TIA Portal project can then be closed.



To close PLC SIM Advanced properly, it is best to proceed as shown.

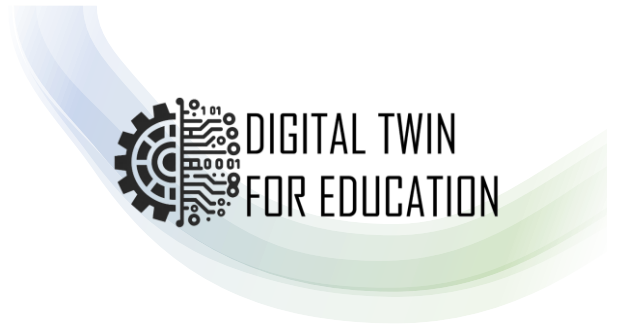
Module 4

Documentation
Key Concept Product development



Module 4

Documentation
Key Concept Product development



Disclaimer

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.