BECKHOFF New Automation Technology

Manual | EN

CXxxxx-M930/B930

Profinet Optional Interface for CX9020, CX5xx0 and CX20xx

2/5/2021 | Version: 1.1

Table of contents

1 Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:
EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
with corresponding applications or registrations in various other countries.

EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

1.1 Representation and structure of warnings

The following warnings are used in the documentation. Read and follow the warnings.

Warnings relating to personal injury:

⚠ DANGER
Hazard with high risk of death or serious injury.

⚠ WARNING
Hazard with medium risk of death or serious injury.

⚠ CAUTION
There is a low-risk hazard that can result in minor injury.

Warnings relating to damage to property or the environment:

NOTE
There is a potential hazard to the environment and equipment.

Notes showing further information or tips:

This notice provides important information that will be of assistance in dealing with the product or software. There is no immediate danger to product, people or environment.

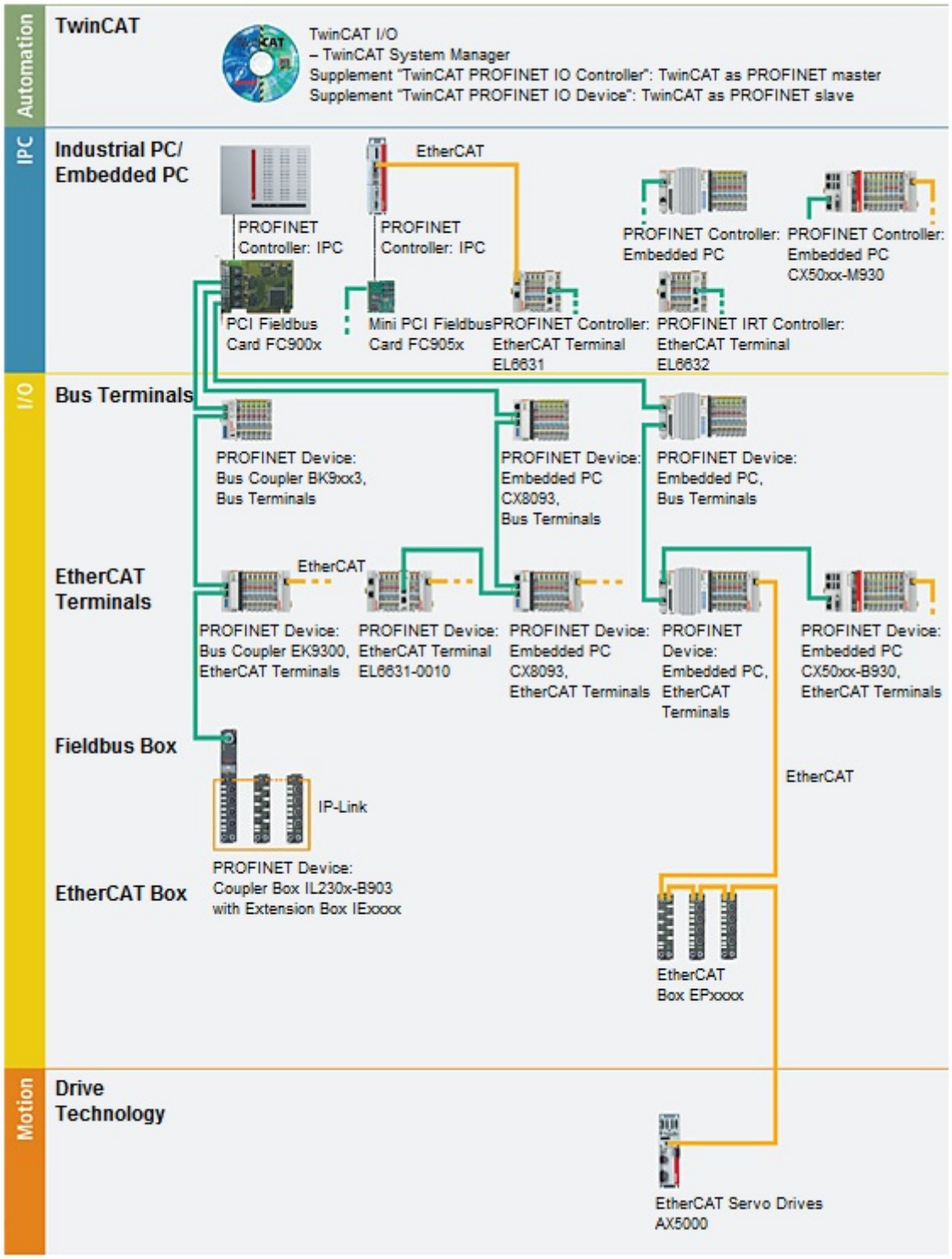
1.2 Documentation issue status

Version	Modifications
1.0	First version
1.1	Chapter "Technical data" and "PROFINET connection" adapted.

2 System overview PROFINET

2.1 Beckhoff components

PROFINET is the open Industrial Ethernet-standard of the PNO (PROFINET user organization). PROFINET IO describes the exchange of data between controllers and field devices in several real-time classes: RT (software-based real-time) and IRT (hardware-supported isochronous real-time). In addition, further Ethernet traffic can be transmitted in the NRT (non-real-time) time slot of the PROFINET cycle. RT can be networked with commercially available switches; switches with corresponding hardware support are required for IRT.



Beckhoff PROFINET components

Components	Comment
Embedded PCs	
CX8093	Embedded PC with PROFINET RT Device fieldbus interface
CX50xx-M930	Embedded PC with optional interface PROFINET RT Controller
CX50xx-B930	Embedded PC with optional PROFINET RT Device interface
EtherCAT terminals	
EL6631	PROFINET IO controller
EL6631-0010	PROFINET IO device
EL6632	PROFINET-IRT controller
Bus Coupler	
BK9053	PROFINET "Compact" Bus Coupler for Bus Terminals
BK9103	PROFINET Bus Coupler for Bus Terminals
EK9300	PROFINET Bus Coupler for EtherCAT Terminals
Fieldbus Box	
IL230x-B903	PROFINET Coupler Box
PC Fieldbus Cards	
FC900x	PCI Ethernet card for all Ethernet (IEEE 802.3)-based protocols
FC9x51	Miniature PCI Ethernet card for all Ethernet (IEEE 802.3)-based protocols
TwinCAT	
TwinCAT PROFINET IO Controller	TwinCAT as PROFINET master
TwinCAT PROFINET IO Device	TwinCAT as PROFINET slave

2.2 Technical data – Profinet

Optional interface M930

Technical data	M930
Fieldbus	Profinet RT Controller
Data transfer rate	100 Mbaud
Bus interface	2 x RJ45 switched
Bus devices	max. 16 for CX9020-M930 max. 32 for CX50x0-M930 max. 64 for CX51x0-M930 max. 64 for CX20xx-M930
Properties	RTClass1

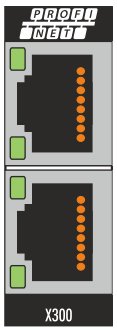
Optional interface B930

Technical data	B930
Fieldbus	Profinet RT device
Data transfer rate	100 Mbaud
Bus interface	2 x RJ45 switched
Extendable process image	1 virtual slave in addition
max. process image	2 slaves x (1440 bytes in / 1440 bytes out)
Properties	RTClass1

3 Connection and cabling

3.1 PROFINET connection

The latest generation of Embedded PCs can be ordered with the optional PROFINET interface (M930/B930). Both Ethernet interfaces are switched and dependent on each other. The optional interface X300 is identified as X300 on the devices and has as black border to identify it.



In a PROFINET slave the incoming PROFINET signal is connected to the upper port of the X300 interface. The lower port relays the signal to further PROFINET slave devices.

Assignment of the LAN ports (X300)

PIN	Signal	Description
1	TD +	Transmit +
2	TD -	Transmit -
3	RD +	Receive +
4	connected	reserved
5		
6	RD -	Receive -
7	connected	reserved
8		

3.2 Cabling

Transmission standards

10Base5

The transmission medium for 10Base5 consists of a thick coaxial cable ("yellow cable") with a max. transmission speed of 10 Mbaud arranged in a line topology with branches (drops) each of which is connected to one network device. Because all the devices are in this case connected to a common transmission medium, it is inevitable that collisions occur often in 10Base5.

10Base2

10Base2 (Cheaper net) is a further development of 10Base5, and has the advantage that the coaxial cable is cheaper and, being more flexible, is easier to lay. It is possible for several devices to be connected to one 10Base2 cable. It is frequent for branches from a 10Base5 backbone to be implemented in 10Base2.

10BaseT

Describes a twisted pair cable for 10 Mbaud. The network here is constructed as a star. It is no longer the case that every device is attached to the same medium. This means that a broken cable no longer results in failure of the entire network. The use of switches as star couplers enables collisions to be reduced. Using full-duplex connections they can even be entirely avoided.

100BaseT

Twisted pair cable for 100 MBaud. It is necessary to use a higher cable quality and to employ appropriate hubs or switches in order to achieve the higher data rate.

10BaseF

The 10BaseF standard describes several optical fiber versions.

Short description of the 10BaseT and 100BaseT cable types

Twisted pair copper cable for star topologies, where the distance between two devices may not exceed 100 meters.

UTP

Unshielded twisted pair
This type of cable belongs to category 3, and is not recommended for use in an industrial environment.

S/UTP

Screened/unshielded twisted pair (screened with copper braid)
Has a general screen of copper braid to reduce influence of external interference. This cable is recommended for use with Bus Couplers.

FTP

Foiled shielded twisted pair (screened with aluminum foil)
This cable has an outer screen of laminated aluminum and plastic foil.

S/FTP

Screened/foiled-shielded twisted pair (screened with copper braid and aluminum foil)
Has a laminated aluminum screen with a copper braid on top. Such cables can provide up to 70 dB reduction in interference power.

STP

Shielded twisted pair
Describes a cable with an outer screen, without defining the nature of the screen any more closely.

S/STP

Screened/shielded twisted pair (wires are individually screened)
This identification refers to a cable with a screen for each of the two wires as well as an outer shield.

ITP

Industrial Twisted-Pair
The structure is similar to that of S/STP, but, in contrast to S/STP, it has only one pair of conductors.

3.3 Topology

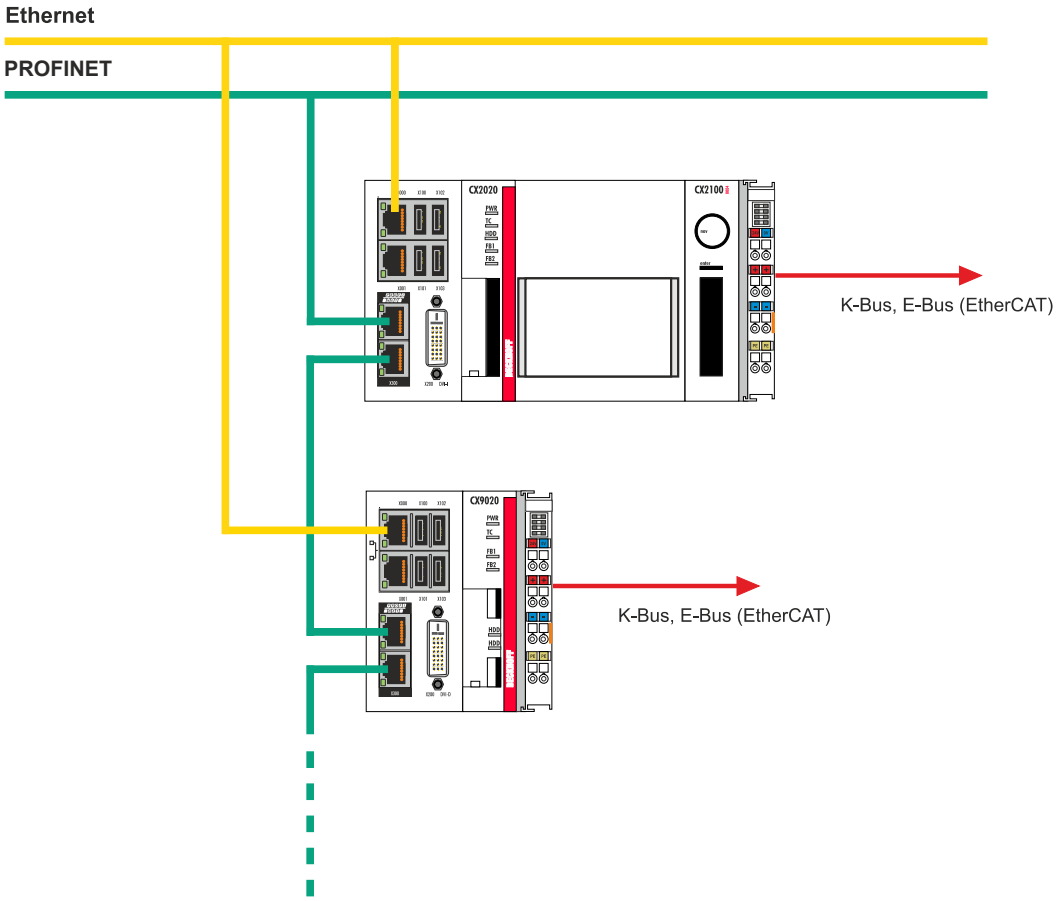


Fig. 1: Example for a PROFINET topology.

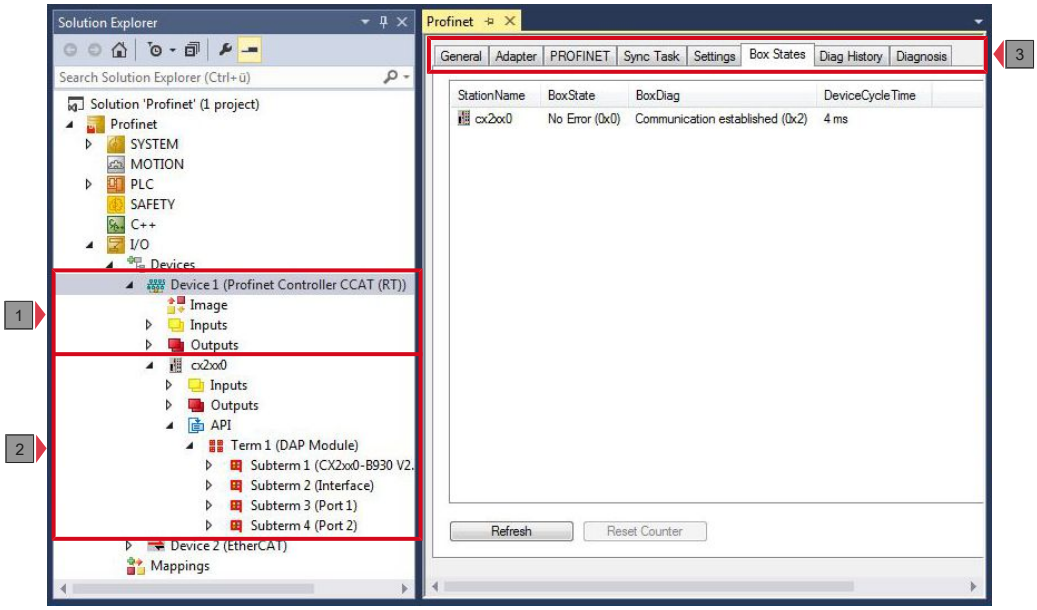
4 TwinCAT tabs

In TwinCAT, information and settings for the PROFINET interface are added under tabs. The main TwinCAT tabs are described in this section. In addition, the section illustrates how the PROFINET interface is displayed in the tree view under TwinCAT.

The tree view and the tabs for a PROFINET interface are identical under TwinCAT2 and TwinCAT3.

4.1 Tree view

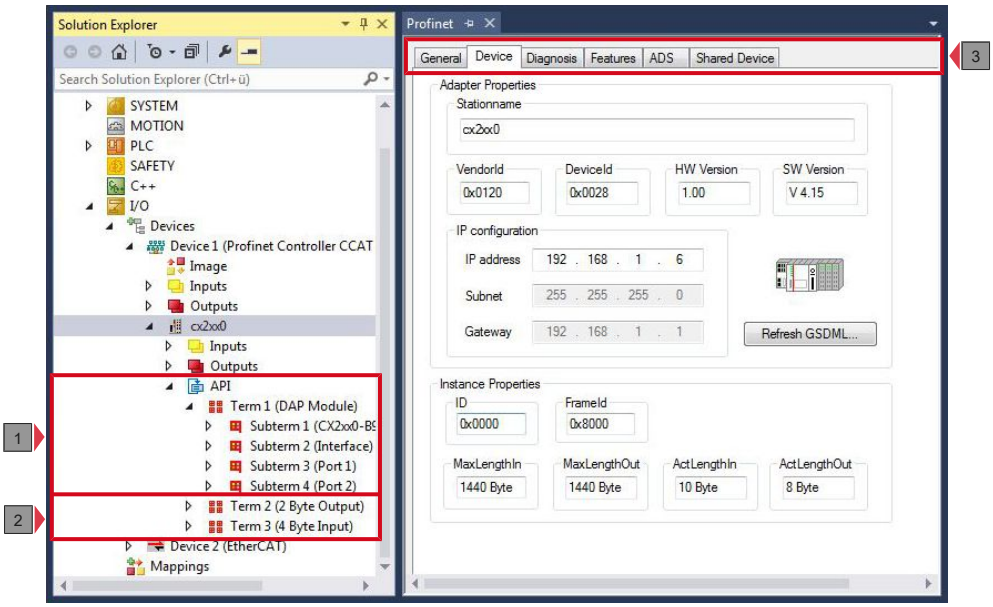
A PROFINET master and a PROFINET slave are displayed as follows in the tree view:



In this sample the slave was linked to the master. TwinCAT was then scanned for the master, and the master was added in TwinCAT together with the slave.

No.	Description
1	Under the PROFINET master, status messages are listed as input variables and output variables. The variables can be linked with the PLC and used for diagnostic purposes (e.g. error codes, counters, etc.).
2	PROFINET slaves are added under the master. Each PROFINET slave has its own input variables for diagnostic purposes, which indicate the state of the communication. The process data is displayed under the API (Application Process Identifier).
3	Further settings for the PROFINET master or slave can be implemented under the tabs. Other tabs are displayed, depending on whether the master or slave is selected in the tree view.

A PROFINET slave and the corresponding tabs are shown as follows in the tree view:



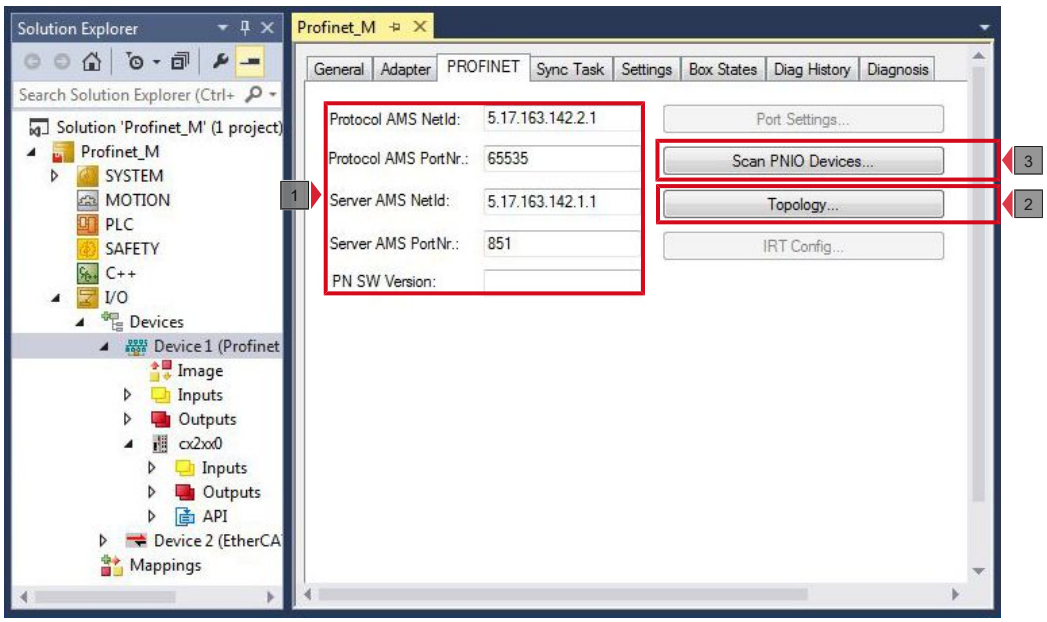
No.	Description
1	The process data is displayed under the API (Application Process Identifier). The DAP (Device Access Point) is always attached by default. It already has fixed properties from the GSDML file, such as process data, interface and PortSub modules.
2	Further user-defined process data are attached as additional modules under the API. Various module types such as Byte, Word, DWord or Real are available.
3	Further settings for the PROFINET slave can be implemented under the tabs. Other tabs are displayed, depending on whether slave or other entries are selected in the tree view.

When the PLC process image is read, the variables for status messages and the variables under the API (Application Process Identifier) can be linked with the variables from the PLC program. Double-click on a variable name in the tree view to open the link dialog. The link variables are identified with a small arrow icon.

Further information about TwinCAT can be found in the TwinCAT documentation on the Beckhoff website: www.beckhoff.de

4.2 Profinet master

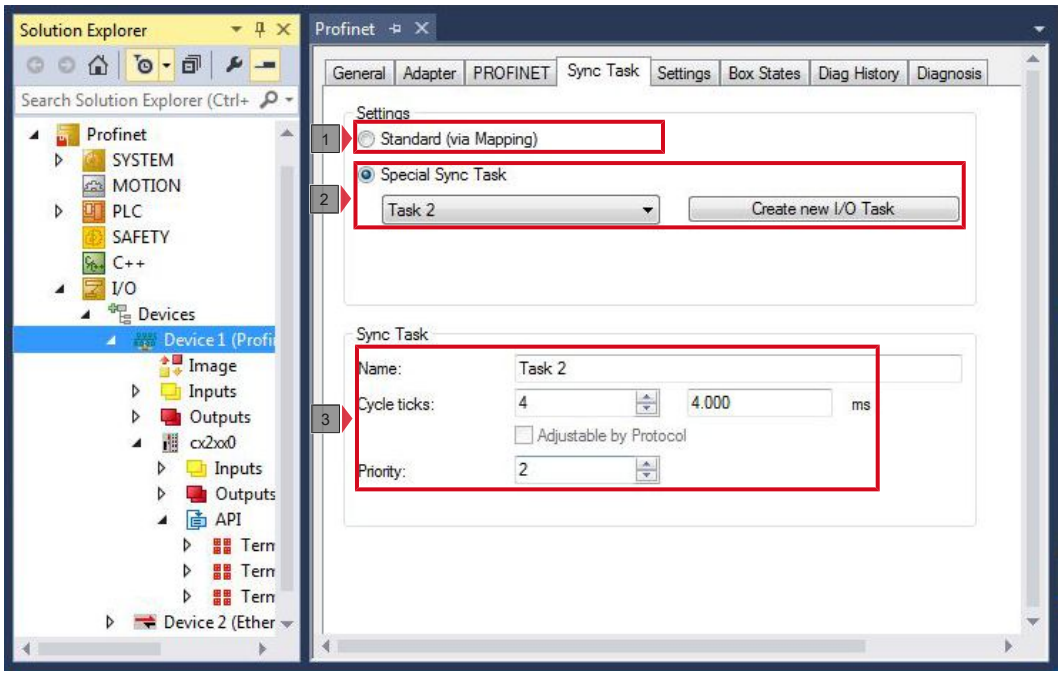
4.2.1 PROFINET



No.	Description
1	All the information required for ADS communication with the PROFINET controller is displayed here. **AMS NetID protocol:** This is the NetID via which the PROFINET controller protocol can be reached via AMS. **AMS PortNo protocol:** This is the PortNo via which the PROFINET controller protocol can be reached via AMS. **AMS NetID server:** This is the NetID to which certain AMS messages are relayed by the PROFINET driver. **AMS PortNo server:** This is the PortNo to which certain AMS messages are relayed by the PROFINET driver
2	The **Topology** button opens a window in which you can compare the online topology with the offline topology.
3	You can use the **Scan PNIO Devices** button to search for further PROFINET slaves. This function is only available in CONFIG mode. After the scan for further PROFINET slaves a further window appears in which all the devices that were found are displayed, along with additional settings.

4.2.2 Sync Task

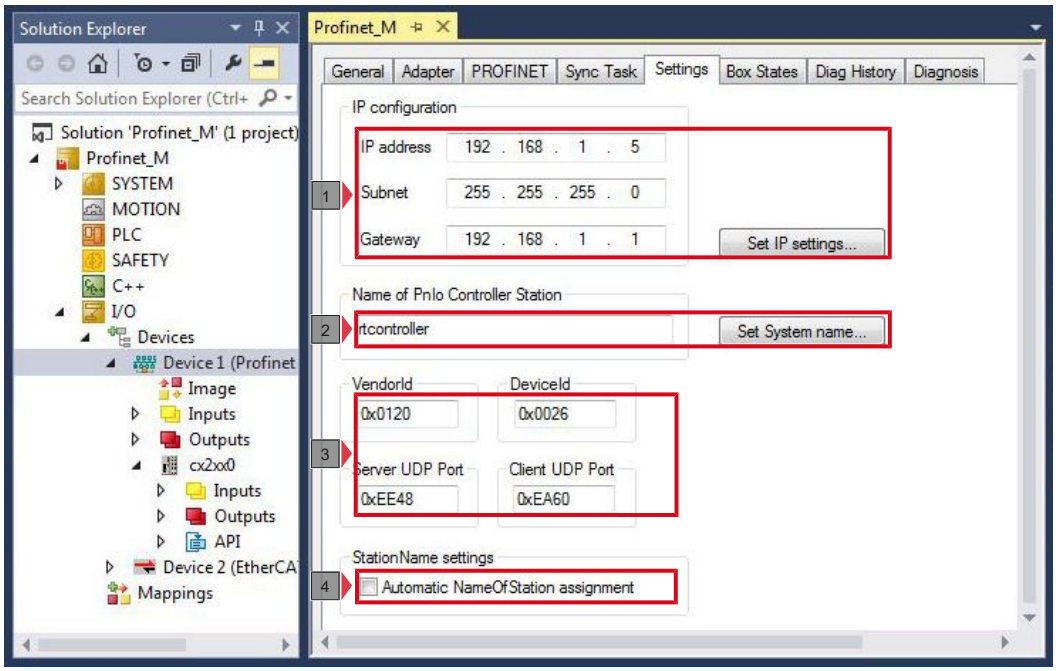
The PROFINET controller protocol must always be linked with a task. The set cycle time is also used for processing the protocol. In theory, the controller can also be processed via a PLC or NC task. However, if a PLC project, for example, is stopped (e.g. by restart or debugging), this results in the PROFINET part also being stopped. In order to avoid such a side effect is it advisable to always create a free-running SyncTask.



No.	Description
1	This option is enabled by default. In this case the Sync task is triggered via the variable mapping. If several tasks are linked to the PROFINET adapter, the task with the higher priority is always used as the Sync task. Pay attention to the Sync task. The Sync task must correspond to the value x^2, i.e. 1 ms, 2 ms, 4 ms, 8 ms, 16 ms etc. The Sync task of the PROFINET controller must not be set under the Sync task of a PROFINET device. Example: If the PROFINET device operates with a 4 ms Sync task, the Sync task of the PROFINET controller may be set to 4 ms, 8 ms, 16 ms, etc. In this case cycle times of 1 ms or 2 ms may not be used.
2	If the Sync task is set via the mapping, note that a breakpoint results in interruption of the task, which also affects the PROFINET communication. You can override this behavior by using the **Special Sync Task** option. The PROFINET device then uses a Sync task that runs independent of the PLC task.
3	At this point the name, cycle time and priority for the cycle time can be set.

4.2.3 Settings

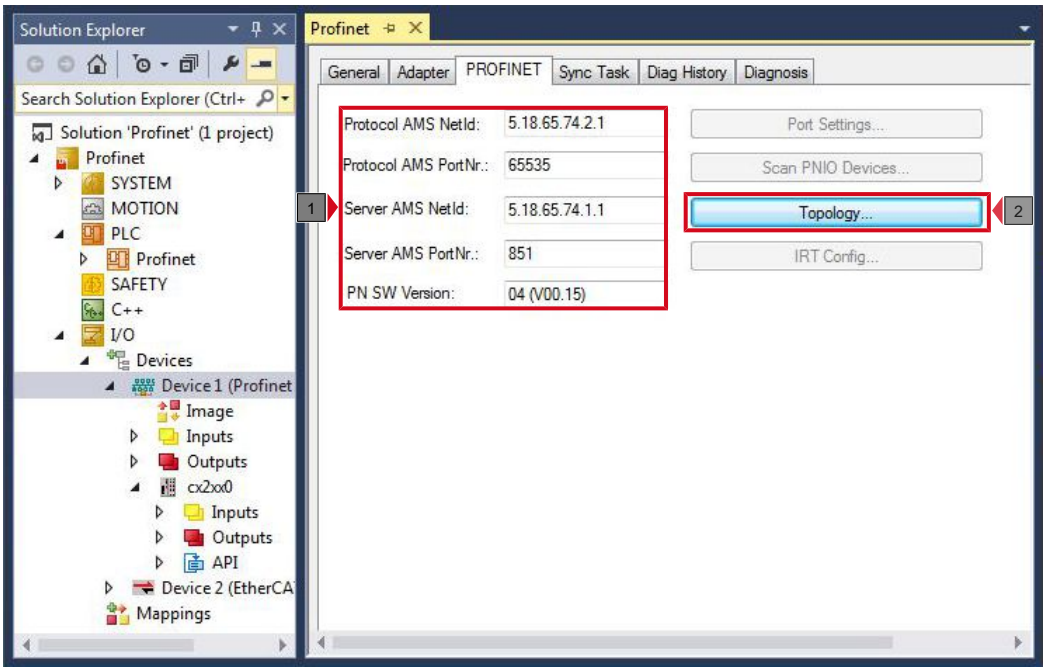
The **Settings** tab contains settings that directly concern the PROFINET controller.



No.	Description
1	At this point you can configure the network settings for PROFINET. By default, values for IP address, subnet and gateway are already entered, but you can change these. Press the **Set IP settings** button to apply your changes. The selection of the address range need not correspond to the network card settings. The PROFINET communication spreads its own net, which can be selected here. If you change the subnet or gateway, the settings are also applied to configured devices.
2	At this point you can set the name for the PROFINET controller. A name is entered by default. Press the **Set System name** button to apply your changes.
3	At this point you can read the VendorID and DeviceID of the controller. You can also set the server and client UDP ports here. However, the default settings are adequate in most cases.
4	If you activate this option, a new PROFINET controller is automatically assigned the name of the previous controller when a device is replaced. This allows you to easily replace a PROFINET controller in the event of a fault.

4.3 Profinet slave

4.3.1 PROFINET

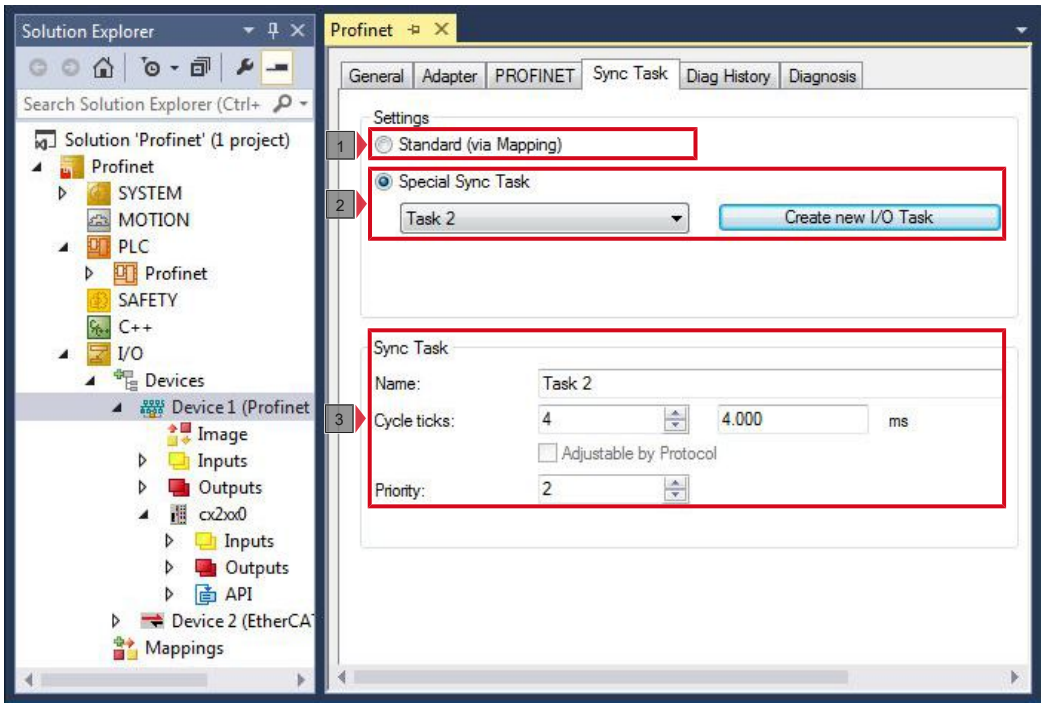


No.	Description
1	All the information required for ADS communication with the PROFINET device is displayed here. **AMS NetID protocol:** This is the NetID through which the PROFINET device protocol can be reached via AMS. **AMS PortNo protocol:** This is the PortNo through which the PROFINET device protocol can be reached via AMS. **AMS NetID server:** This is the NetID to which certain AMS messages are relayed by the PROFINET driver. **AMS PortNo server:** This is the PortNo to which certain AMS messages are relayed by the PROFINET driver
2	This button opens a window in which you can compare the online topology with the offline topology.

4.3.2 Sync Task

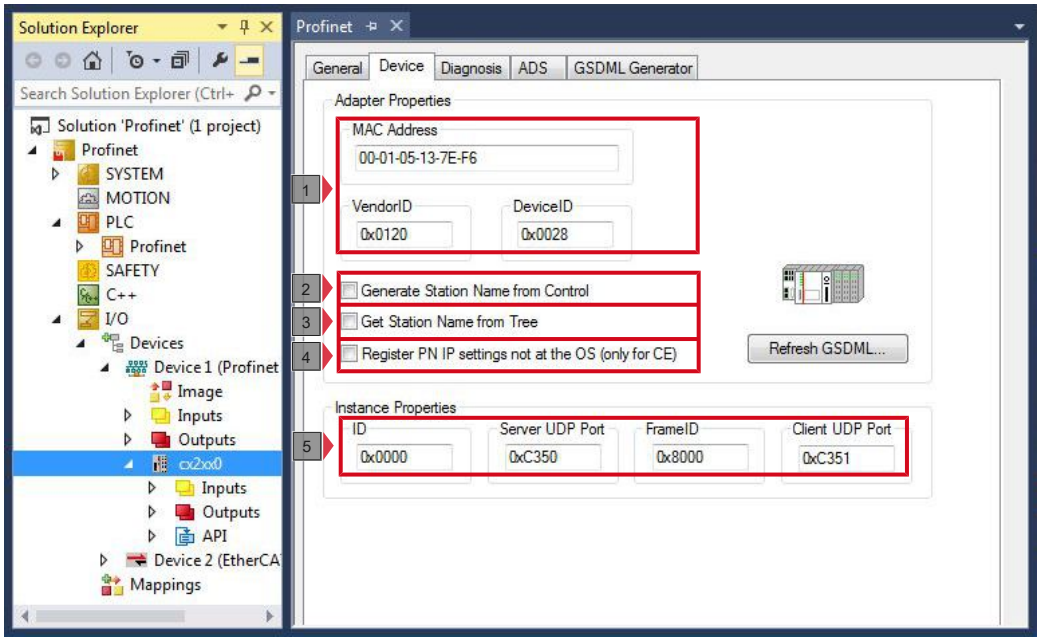
The SyncTask triggers the PROFINET task and thus the speed with which the PROFINET communication operates.

Pay attention to the system utilization rate of your Embedded PCs. The shorter the PROFINET cycle time is, the higher the total system utilization rate will be. A very high system utilization rate can lead to an ADS connection no longer being triggered often enough and this connection may be interrupted.



No.	Description
1	This option is enabled by default. In this case the Sync task is triggered via the variable mapping. If several tasks are linked to the PROFINET adapter, the task with the higher priority is always used as the Sync task. Pay attention to the Sync task. The Sync task must correspond to the value x^2, i.e. 1 ms, 2 ms, 4 ms, 8 ms, 16 ms etc. The Sync task of the PROFINET controller must not be set under the Sync task of a PROFINET device. Example: If the PROFINET device operates with a 4 ms Sync task, the Sync task of the PROFINET controller may be set to 4 ms, 8 ms, 16 ms, etc. In this case cycle times of 1 ms or 2 ms may not be used.
2	If the Sync task is set via the mapping, note that a breakpoint results in interruption of the task, which also affects the PROFINET communication. You can override this behavior by using the **Special Sync Task** option. The PROFINET device then uses a Sync task that runs independent of the PLC task.
3	At this point the name, cycle time and priority for the cycle time can be set.

4.3.3 Device



No.	Description
1	Information on MAC address, VendorID and DeviceID of the device.
2	This option can be used to extend the name via a PLC program. The extension consists of a three-digit numerical value. The value must be entered in the PLC program as a constant value and must be available at the start of the PLC program. Then link the value to the variable PnIoBoxCtrl.
3	With this option, the name is taken from the tree view.
4	This option only applies to Windows CE. In delivery state the PROFINET IP address for the CCAT interfaces is registered with the operating system. This enables you to access the device via TCP/IP mechanisms when TwinCAT is in RUN mode. Enable this option if you do **not** want the PROFINET IP address to be registered with the operating system.
5	These fields can be used to change the InstanceID and FrameID. However, the default settings are adequate for most applications. The Instance ID is incorporated into the formation of the UUID object. The settings should therefore only be changed in exceptional cases.

5 Parameterization and commissioning

5.1 Searching for target systems

Before you can work with the devices, you must connect your local computer to the target device. Then you can search for devices with the help of the IP address or the host name.

The local PC and the target devices must be connected to the same network or directly to each other via an Ethernet cable. In TwinCAT a search can be performed for all devices in this way and project planning subsequently carried out.

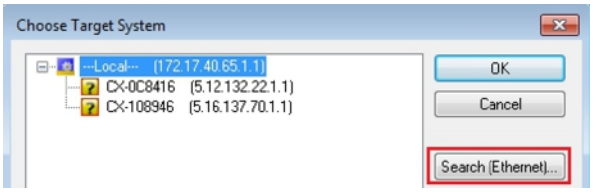
Prerequisites for this step:

- TwinCAT 3 must be in Config mode.
- IP address or host name of the device.

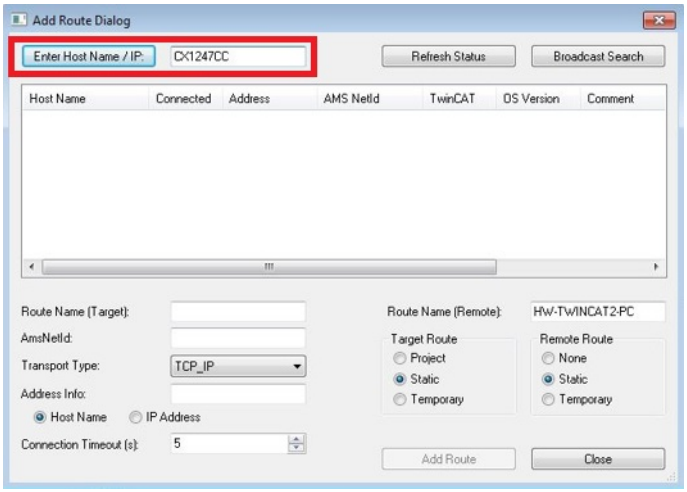
Search for the devices as follows:

1. In the menu at the top click on **File > New > Project** and create a new TwinCAT XAE project.
2. In the tree view on the left click on **SYSTEM**, and then **Choose Target**.

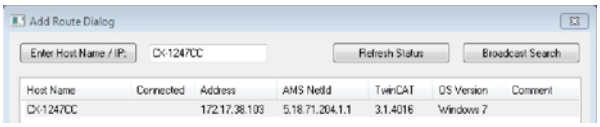
3. Click on **Search (Ethernet)**.



4. Type the host name or the IP address of the device into the **Enter Host Name / IP** box and press **[Enter]**.



5. Mark the device found and click on **Add Route**.

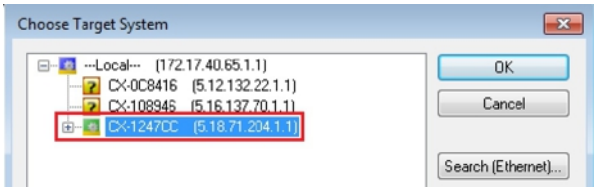


The Logon Information window appears.
Enter the user name and password for the CX in the **User Name** and **Password** fields and click **OK**.

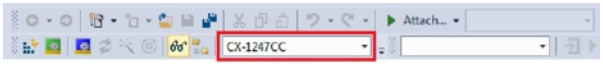
The following information is set as standard in CX devices:
User name: Administrator **Password:** 1

6. If you do not wish to search for any further devices, click on **Close** to close the Add Route Dialog.
The new device is displayed in the Choose Target System window.

7. Select the device you want to specify as target system and click **OK**.



⇨ You have successfully searched for a device in TwinCAT and inserted the device as the target system.
The new target system and the host name are displayed in the menu bar.



Using this procedure you can search for all available devices and also switch between the target systems at any time. Next, you can append the device to the tree view in TwinCAT.

5.2 Adding a PROFINET slave

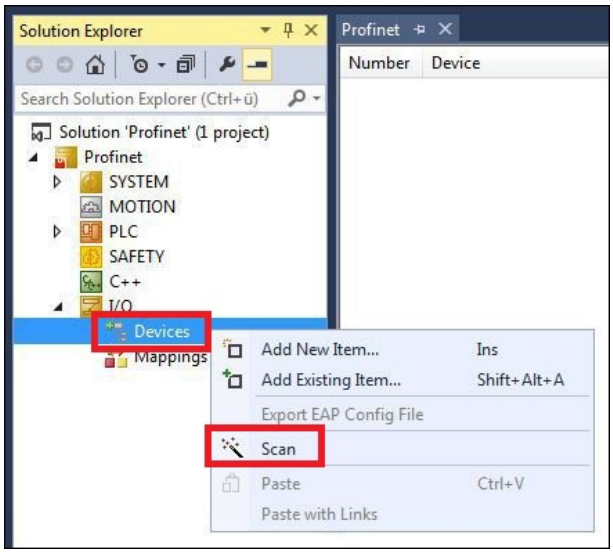
In the example configuration a CX2020 PROFINET slave with B930 option interface is used. In order to ensure that the PROFINET slave is configured and subsequently detected by the PROFINET master with all inputs and outputs, the PROFINET slave must first be added in TwinCAT.

Prerequisites for this step:

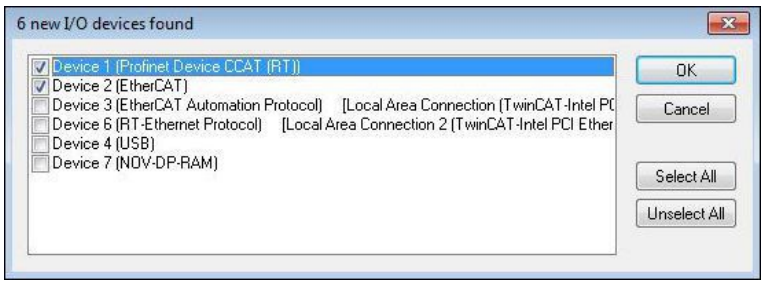
- A scanned and selected target device with PROFINET slave. In this example it is the CX2020 with optional B930 interface.

Add the PROFINET slave as follows:

1. Start TwinCAT and open an empty project.
2. In the tree view on the left, right-click on **Devices**.
3. In the context menu click on **Scan**.

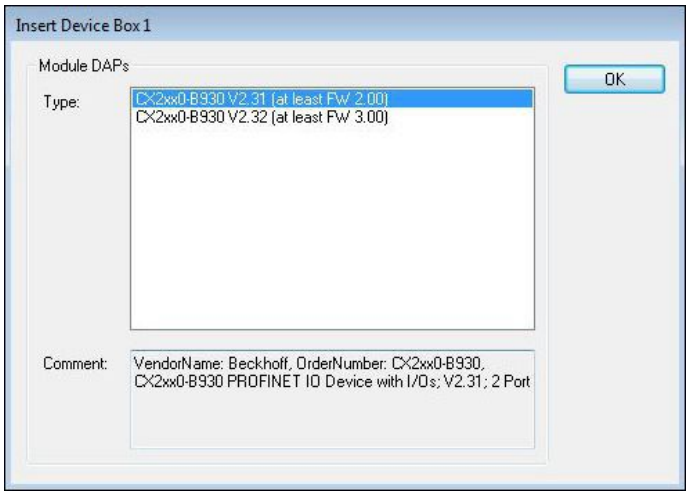


4. Select the devices you want to use and confirm the selection with **OK**.



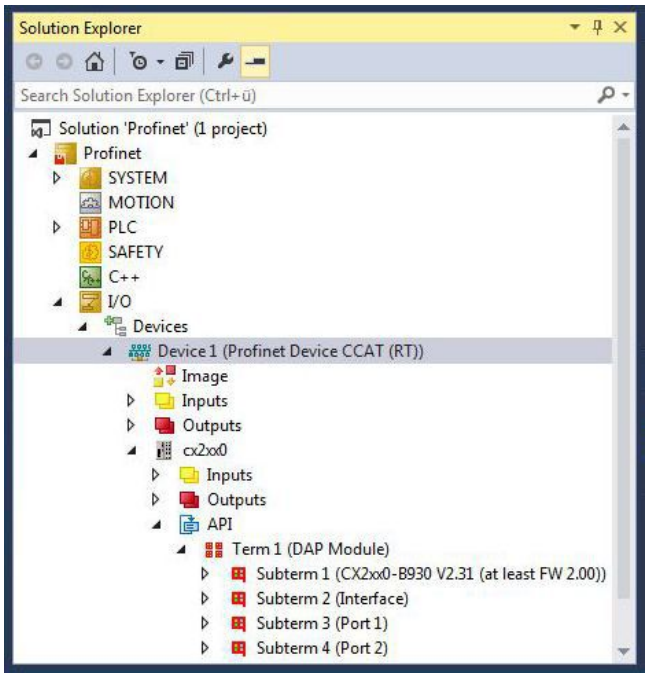
5. Confirm the request with Yes, in order to look for boxes. Device 1 (PROFINET device CCAT (RT)) is integrated. The Insert Device Box 1 window appears.

6. Select the corresponding DAP module, which the PROFINET master supports, and click **OK**.



7. Confirm the request whether to enable FreeRun with Yes.

⇨ The PROFINET slave was successfully added in TwinCAT 3 and is displayed in the tree view with the inputs and outputs.



In the next step you can configure the process data.

5.3 Configuring process data

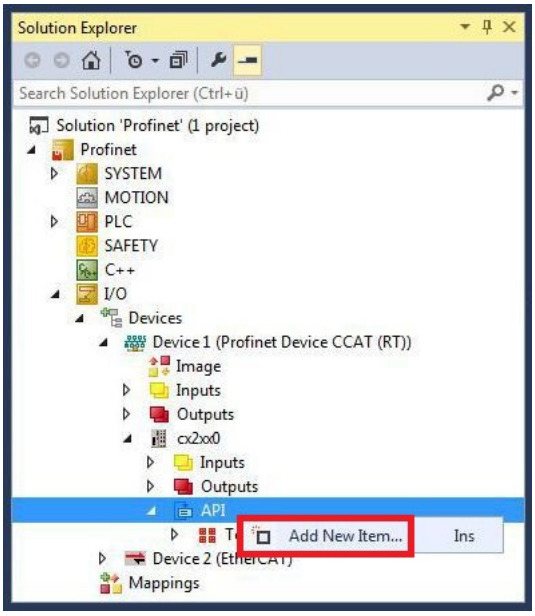
You can configure the process data to be transferred via PROFINET. Various module types, e.g. Byte, Word, DWord or Real are available in various lengths. The module types are created in the tree view of the System Manager under the API (Application Process Identifier).

Requirements:

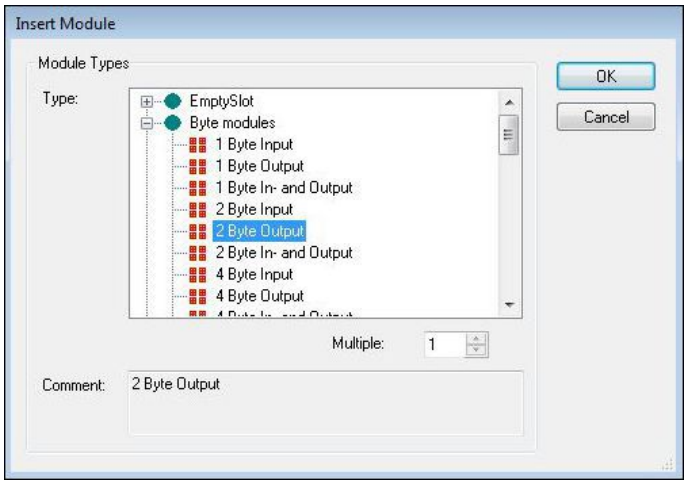
- A PROFINET slave added in TwinCAT

Configure the process data as follows:

1. Right-click on the **API** in the tree view on the left.
2. Click on **Add New Item** in the context menu.



3. Select the required module and enter the number of modules in the field **Multiple**. TwinCAT uses the GSDML file under: *C:\TwinCAT\3.1\Config\Io\Profinet*



⇨ The required number of modules are created under the API (Application Process Identifier). In the next step you can either rotate the process data, create a virtual slave or create a PLC project.

5.4 'Turning' process data

The process data are transferred in Intel format as standard. If the data are required in Motorola format, they have to be 'turned' accordingly. This step illustrates how to 'turn' the data in TwinCAT.

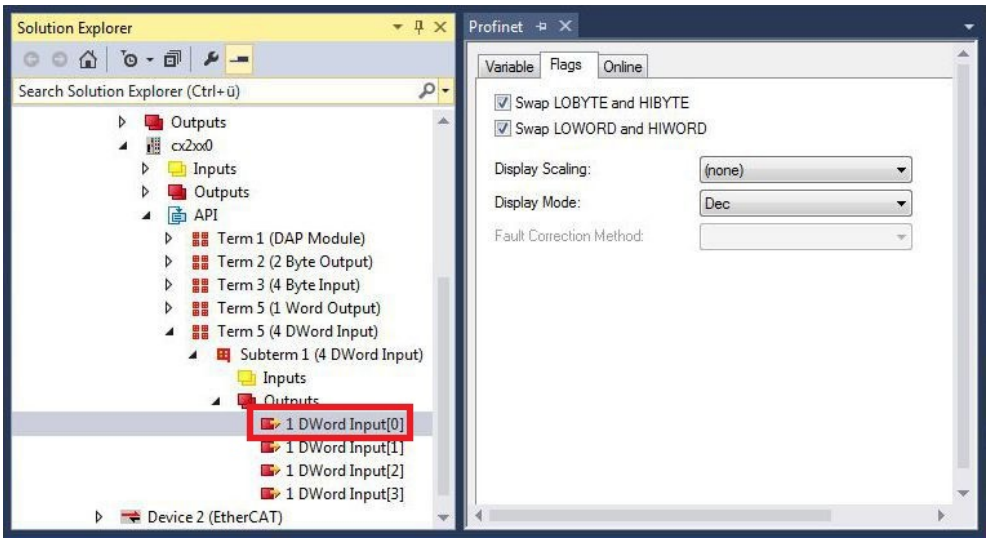
If the standard format is required, you can skip this step.

Prerequisites for this step:

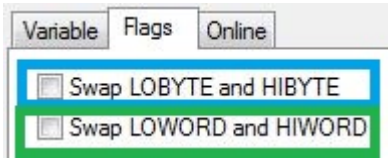
- A parameterized slave.

'Turn' the process data as follows:

1. In the tree view, right-click on a variable containing data to be 'turned'.
2. Click on the **Flags** tab.



3. Click on the required option. For WORD variables, only LOBYTE and HIBYTE can be swapped. With DWORD variables you can additionally swap the WORD.



⇨ In this way you can 'turn' process data. Use the following example to see how the data change for the individual options.
Example for DWORD.

Data of the slave	Data which the master receives			
Original data	No option selected	Swap Byte (blue)	Swap Word (green)	Swap both (blue and green)
0x01020304	0x01020304	0x02010403	0x03040102	0x04030201

The data can also be 'turned' in the PLC project, using the command ROR.
Example for ST: VarProfinet:=ROR(VarAnalog.8); (*both variables of type WORD*)

5.5 Creating a virtual slave

Additional virtual slaves can be created on the same hardware interface. This enables more data to be exchanged with a PROFINET master, or a connection with a second PROFINET master can be established.

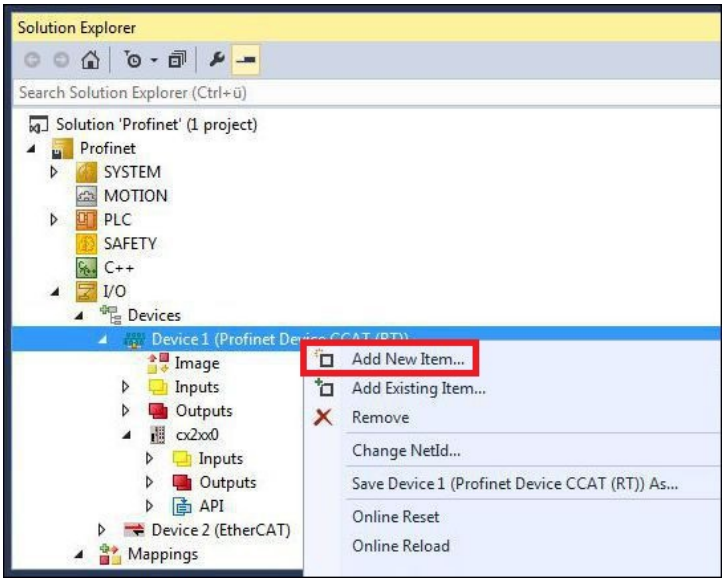
Each virtual slave is assigned a dedicated address via TwinCAT and is configured like an independent device for the PROFINET master.

Prerequisites for this step:

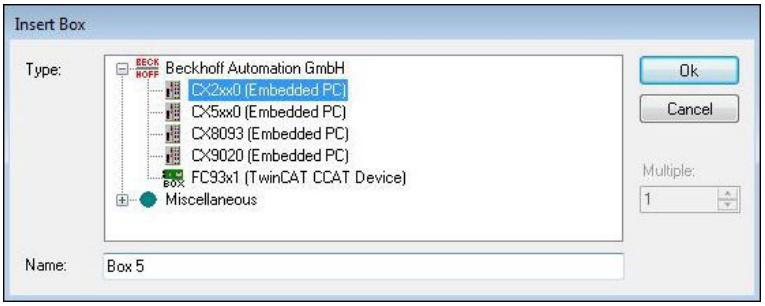
- A PROFINET slave, created in TwinCAT.

Create a virtual slave as follows:

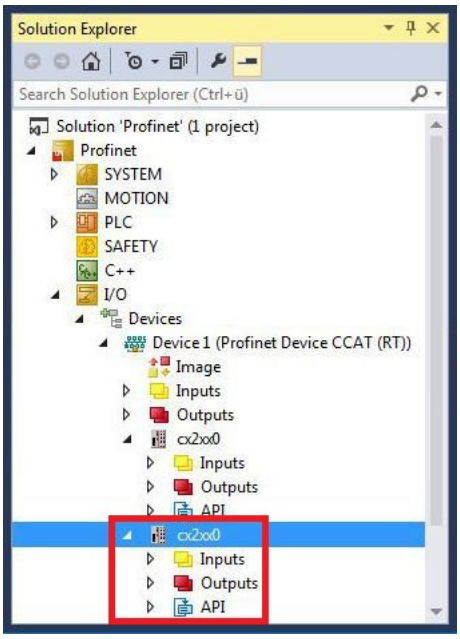
1. Right-click on the PROFINET device in the tree view on the left.
2. Click **Add New Item**.



3. Select the appropriate box, e.g. the CX2xx0 (Embedded PC) if you use a CX20xx Embedded PC with optional PROFINET interface.



⇨ The virtual PROFINET slave is created in the tree view. You can now configure your own process data for the virtual slave.



The MAC address of the virtual slave is editable. Make sure that the MAC address occurs only once in the system. The IP address is assigned by the PROFINET master

5.6 Creating a PLC project

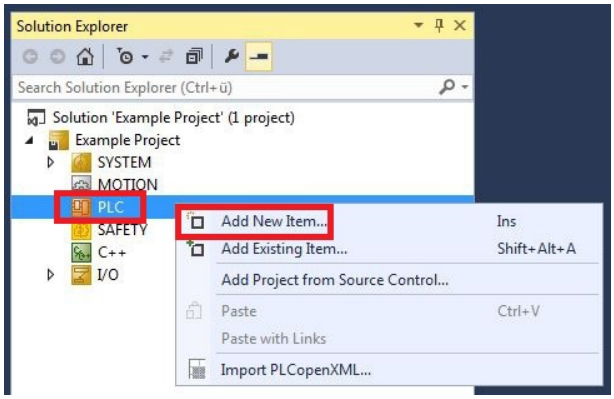
The next steps describe how to create a PLC project in TwinCAT and add it in the tree view.

Prerequisites for this step:

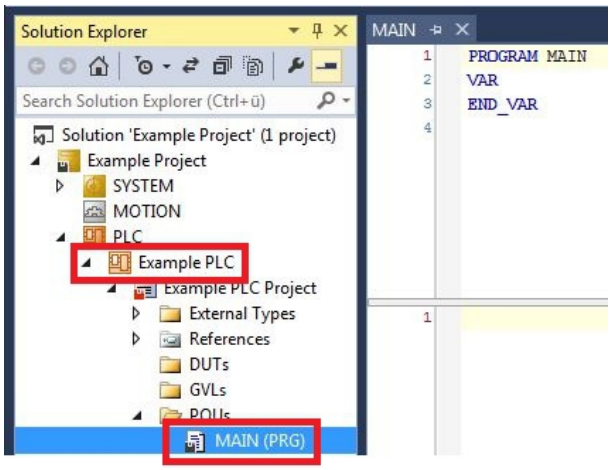
- A newly created TwinCAT XAE project.

Create a PLC project as follows:

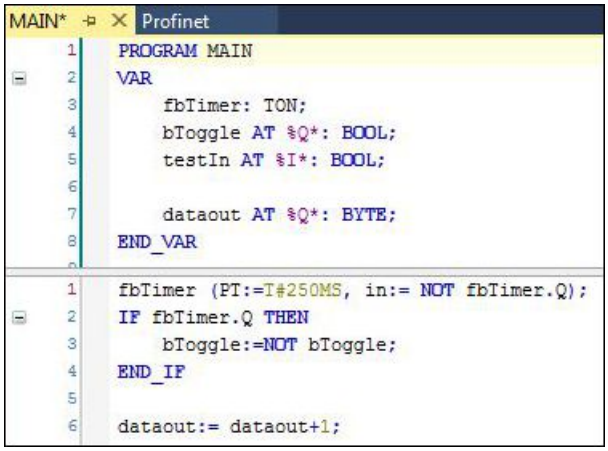
1. Right-click on **PLC** in the tree view.
2. In the context menu click on **Add New Item** and select the **Standard PLC Project**.



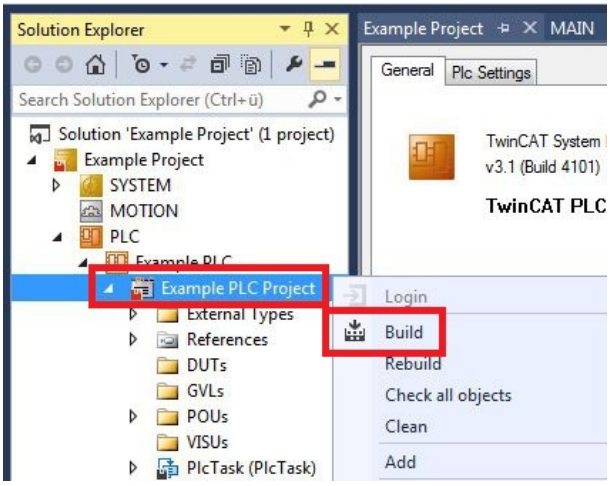
3. In the tree view click on the newly created PLC project, then double-click on **MAIN (PRG)** under **POUs**.



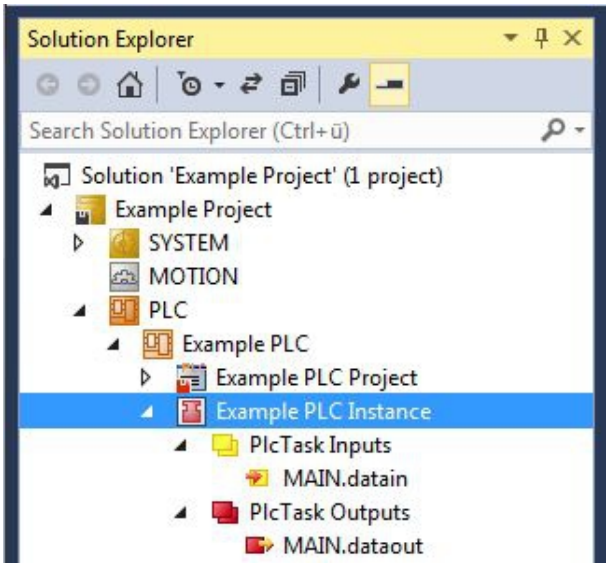
4. Write a small program, as shown in the diagram below.



5. In the tree view right-click on the PLC project, then click on **Build** in the context menu.



⇨ You have successfully created a PLC project and added the project in TwinCAT. A PLC instance with the variables for the inputs and outputs is created from the PLC project.



In the next step you can link the variables with the hardware.

5.7 Linking variables

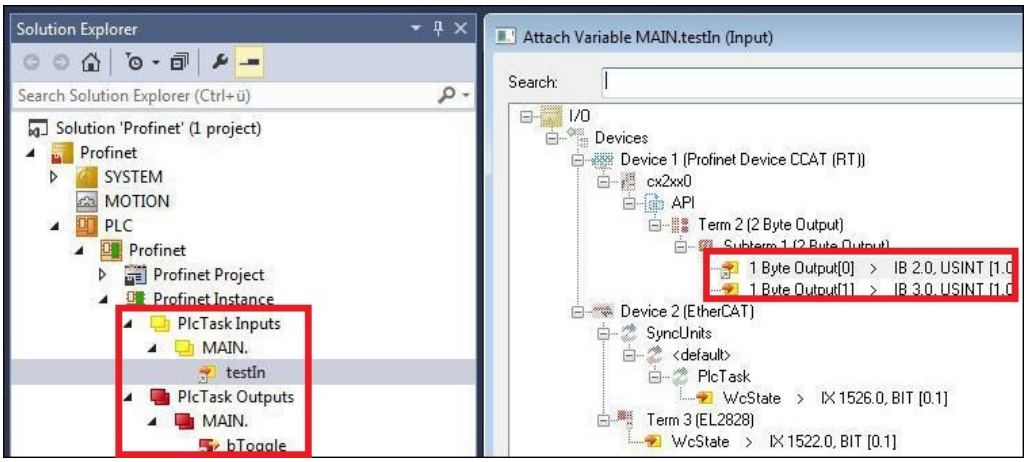
Once the PLC project was successfully added in the System Manager, you can link the newly created input and output variables from the PLC project with the inputs and outputs of your hardware.

Prerequisites for this step:

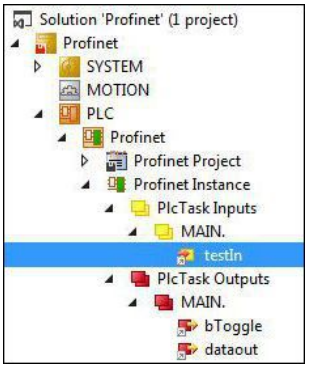
- A PLC program attached in TwinCAT.

Link the variables as follows:

1. Double-click on the input or output variables in the tree view under **PLC**.
   The **Attach Variable** window appears and shows which inputs or outputs can be linked with the variables from the PLC project.

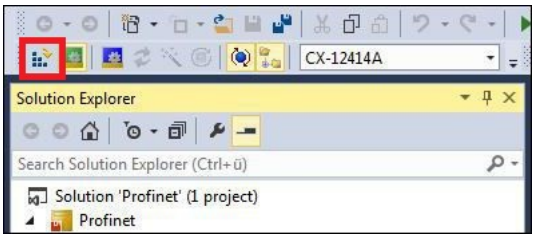


2. Double-click on the inputs or outputs of the hardware in the **Attach Variable** window.
   Link the input variables with the inputs and the output variables with the outputs of the hardware.



   Variables that are already linked are indicated with a small arrow icon in TwinCAT.

3. In the toolbar click on **Activate Configuration**.



   Confirm the request whether TwinCAT is to start in Free Run mode with **Yes**.

⇨ You have successfully linked variables with the hardware. Use Activate Configuration to save and activate the current configuration.

The configuration can now be loaded on the CX, in order to automatically start TwinCAT in Run mode, followed by the PLC project.

5.8 Load configuration to CX

Once variables are linked, the configuration can be saved and loaded on the CX. This has the advantage that the PLC project is loaded and started automatically when the CX is switched on. The start of the previously created PLC project can thus be automated.

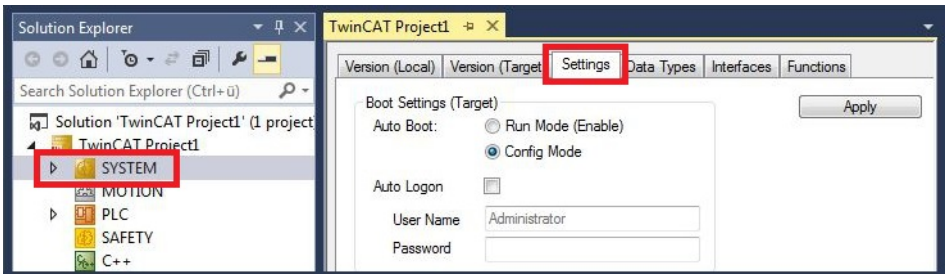
Prerequisites for this step:

- A completed PLC project, added in the System Manager.
- Variables from the PLC project, linked with the hardware in the System Manager.
- A CX selected as target system.

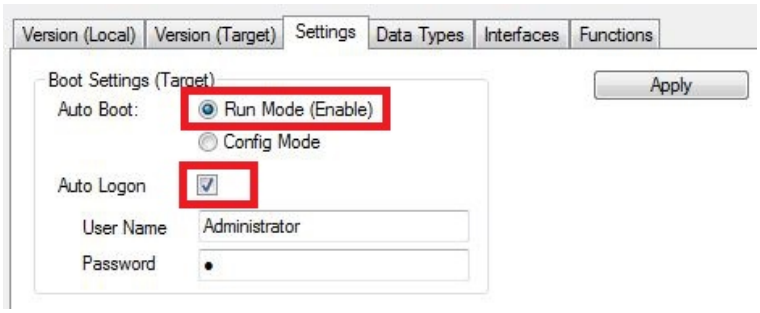
Load the configuration from the System Manager to the CX as follows:

1. In the tree view on the left click on **SYSTEM**.

2. Click on the **Settings** tab.



3. Under Boot Settings select the option **Run Mode (Enable)** and tick the **Auto Logon** checkbox.

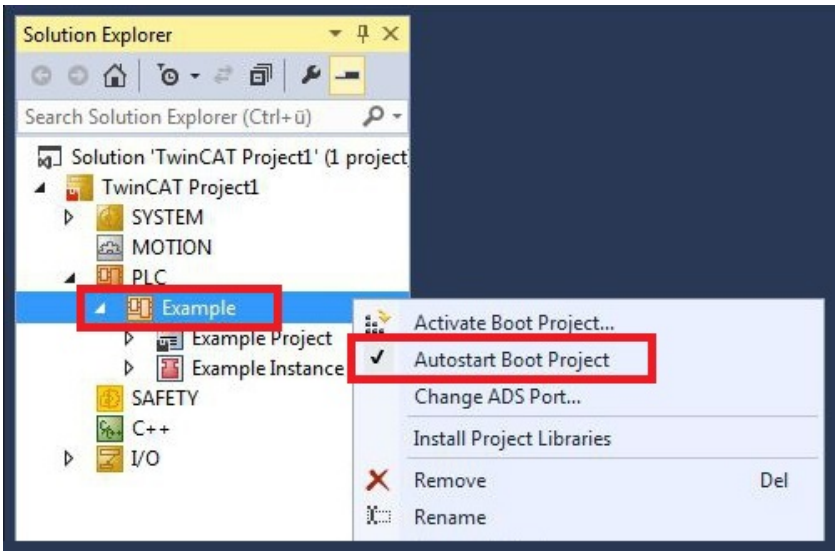


4. Enter the user name and password for the CX in the **User Name** and **Password** fields.

5. Click on **Apply**.

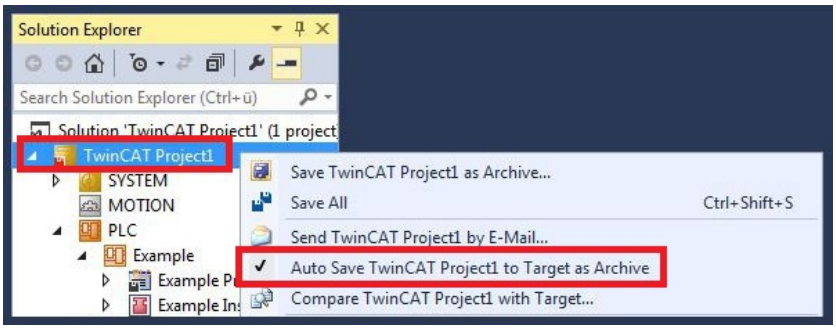
6. In the tree view on the left right-click on the PLC project under **PLC**.

7. In the context menu click on **Autostart Boot Project**.
   The setting is selected



8. Right-click on the project folder in the tree view.

9. In the context menu click on **Auto Save to Target as Archive**.
   The setting is selected.



⇨ You have successfully loaded the CX configuration. From now on, TwinCAT will start in Run mode and the PLC project will start automatically.

Next, the master can be added in a new project in the System Manager and can then be used to find slaves that have already been set up.

5.9 Adding the PROFINET master

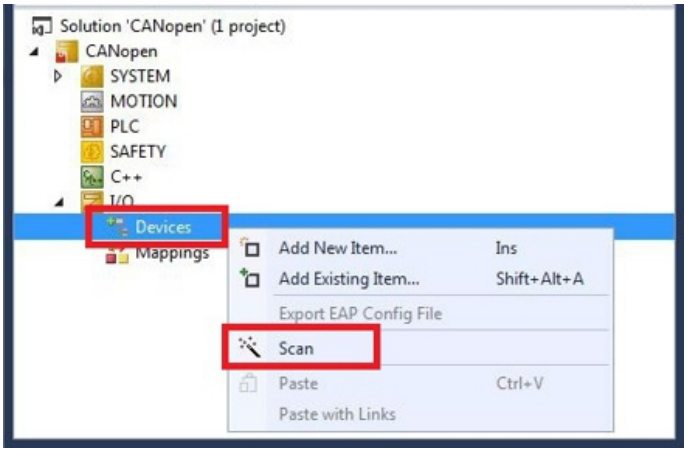
The PROFINET master is added in the TwinCAT System Manager in the same way as the PROFINET slave. You can then find all connected slaves via the master. The following section illustrates how to add a PROFINET master in TwinCAT.

Prerequisites for this step:

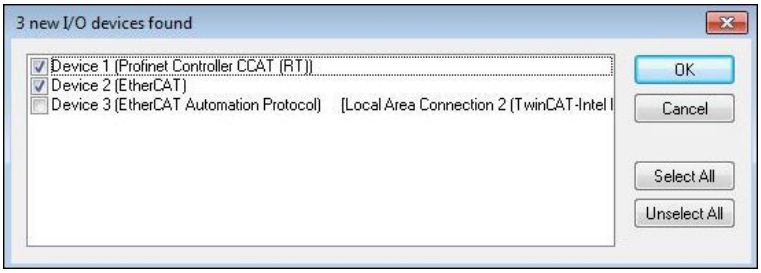
- TwinCAT must be in Config mode.
- A selected target system (in this sample it is the Embedded PC CX5020-M930)

Add a PROFINET master as follows:

1. In the tree view on the left, right-click on **Devices**.
2. In the context menu click on **Scan**.

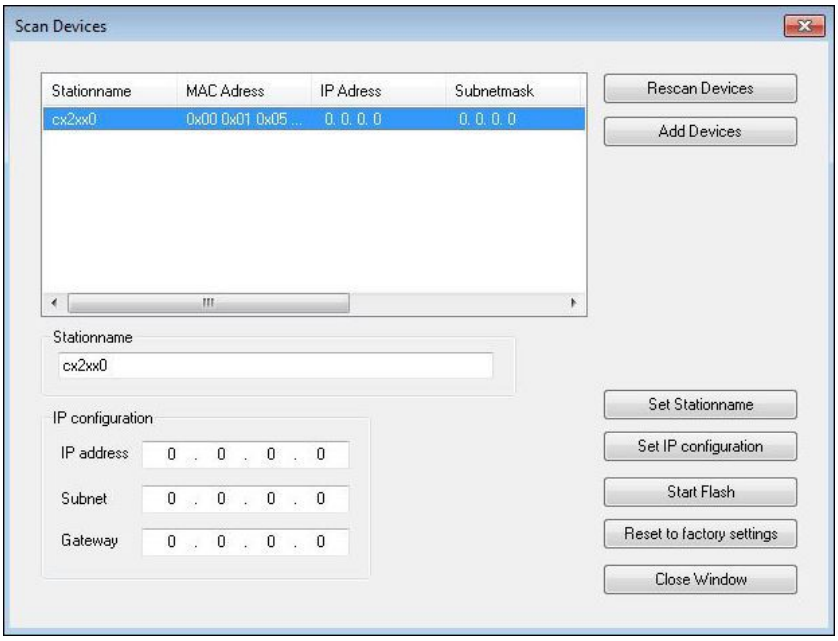


3. Select the devices you want to use and confirm the selection with **OK**.

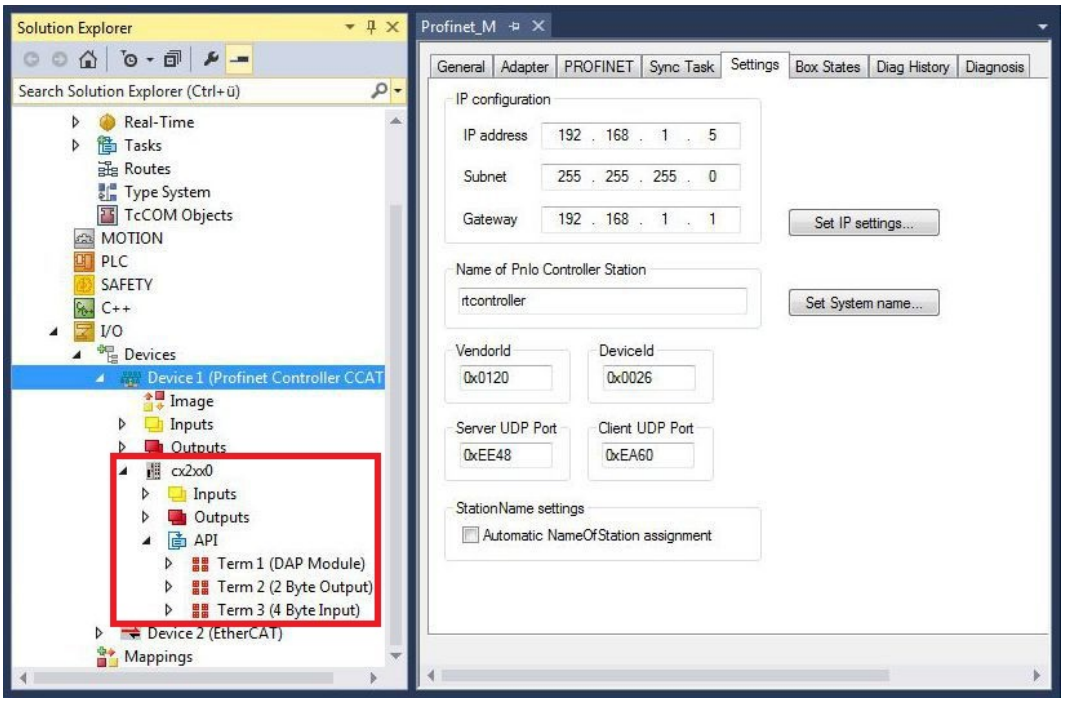


4. Confirm the request with Yes, in order to look for boxes.
   The **Scan Devices** window appears.

5. Select the PROFINET slaves you want to use and click **Add Devices**.



⇨ The selected PROFINET slaves are then displayed on the left in the tree view. The PROFINET slaves are assigned IP addresses based on the PROFINET master.



If you do not find all PROFINET slaves, check the wiring and search for the PROFINET slaves again.

6 Error handling and diagnostics

6.1 Diagnostic LEDs

Table 1: PN diagnostic LED, description of the flashing behavior.

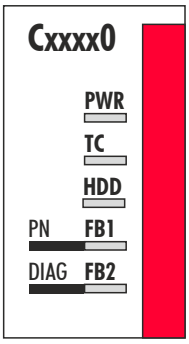
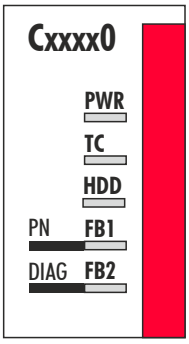
Display	LED	PROFINET status		Meaning
		green	red	
	PN	off	flashing 200ms	Power on, startup phase
		200 ms flashing	off	no PROFINET name
		1 sec off, 200 ms on	off	No IP address
		on	off	Run

Table 2: DIAG diagnostic LED, description of the flashing behavior.

Display	LED	PROFINET diagnosis		Meaning
		green	red	
	DIAG	500 ms flashing	500 ms flashing	PN controller identification. The PN controller is transmitting an identification signal.
		off	200 ms flashing	The establishment of a connection with the controller has not been completed.
		1 s off 200, ms on	off	Problem when establishing a connection, or the actual and target configurations are different.
		200 ms	off	The device is in data exchange but the PLC is in Stop mode.
		on	off	The device is in data exchange.

If a virtual PROFINET slave was configured, this is also covered by the LEDs in the event of an error. The physical device always has higher priority. The status of the virtual slave is only displayed once everything is OK with the physical device.

6.2 Box states

Directly below the PROFINET controller there are variables containing general information about the state of the PROFINET communication.

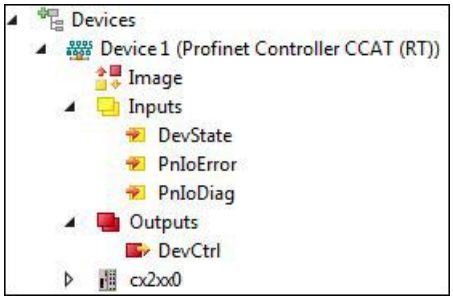


Fig. 2: Diagnostic variables DevState, PnIoError and PnIoDiag in the TwinCAT tree view.

This data is exchanged between the PROFINET driver and the System Manager.

Table 3: DevState, PnIoError and PnIoDiag, description of the variables.

Variable	Description
DevState	The variable DevState contains information about the physical communication status of the PROFINET controller, such as the link status or whether the sender resources are still adequate.
PnIoError	The Error variable shows possible problems when establishing a connection and counts the PROFINET devices affected by an error.
PnIoDiag	The diagnostic variable provides status information about an existing connection. The variable counts the PROFINET device affected by a warning or diagnostics.

The ‘DevCtrl’ output variable currently has no function.

6.3 Cyclic diagnostics

The variables PnIoBoxState and PnIoBoxDiag are available for cyclic diagnostics.

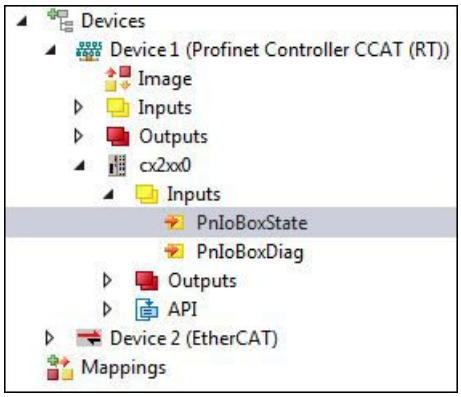


Fig. 3: Diagnostic variables PnIoBoxState and PnIoBoxDiag in the TwinCAT tree view.

These variables are cyclically exchanged with the process image between the PROFINET driver and the System Manager.

PnIoBoxState

Table 4: PnIoBoxState variable, description of the states.

Number	Text	Description	Remedial action / reason
0	No error	No error	No error
1	PROFINET Device state machine is in boot mode	PROFINET Device State Machine is still in the start-up phase	Not an error, wait
2	Device not found	Device does not reply to the Identify Request	Check connection, device connected, was the device called by its correct name?
3	The stationname is not unique	The station name is not unique	There are two or more devices in the network with the same PROFINET name. A correct identification cannot take place.
4	IP could not set	IP address could not be set.	The PROFINET device has rejected the IP settings for some reason. Check whether the IP settings are correct.
5	IP conflict	An IP conflict has occurred in the network.	A possible cause is that several devices have the same IP address.
6	DCP set was not successful	There was no reply or an erroneous reply to a DCP Set.	Check connection, device connected, was the device called by its correct name?
7	Watchdog error	The connection was broken off with a Watchdog error.	Check the cycle time, check the connection, if necessary increase the Watchdog factor.
8	Datahold error	The connection was broken off with a Datahold error.	Frame Data status was invalid for the length of the DataHoldTimer. Restart the device if necessary.
9	RTC3: Sync signal could not be started	For IRT only: the Sync signal could not be started.	Is EtherCAT Sync signal correct or has Sync0 started?
10	PROFINET Controller has a link error	The PROFINET controller has no link.	Check cable and connection.
11	The aliasname is not unique	The alias name is not unique	There are two or more devices in the network with the same alias name. This is made up of the neighborhood information (PortId.ChassisId). A correct identification cannot take place.
12	The automatic name assignement isn't possible - wrong device type	The automatic name assignment is not possible.	The expected PROFINET device is not in the projected position (VendorId or DeviceId does not correspond). Hence, no automatic naming and thus no device start is possible.

Number	Text	Description	Remedial action / reason
31	only for EtherCAT gateways: WC-State of cyclic EtherCAT frame is 1	For EL6631 only: EtherCAT WC State is 1	Check the mode on the EtherCAT master & slave (OP?).

PnIoBoxDiag

In contrast to the State variable, the variable PnIoBoxDiag can be used to display more than one state at the same time, i.e. the information is bit-coded and up to 16 parameters can be displayed. The following statuses are currently displayed.

Table 5: PnIoBoxDiag variable, description of the states.

Number	Description
0x0000	No diagnosis
0xXXX1	IOC-AR is not established
0xXXX2	IOC-AR is established
0xXXX4	IOC-AR is established but no ApplReady
0xXXX8	IOC-AR is established but module difference
0xXX1X	At least one AlarmCR get diagnosis alarm
0xX1XX	At least one InputCR is invalid
0xX2XX	At least one InputCR provider is in stop
0xX4XX	At least one InputCR problem indicator is set
0x1XXX	At least one OutputCR is invalid
0x2XXX	At least one OutputCR provider is in stop
0x4XXX	At least one OutputCR problem indicator is set

On the one hand information about the status of the IO Controller Single AR is displayed here. In addition, collective statuses are formed from the Frame Data statuses of the individual CRs. The whole thing happens for the input and the output CRs (currently only one is possible; in future the controller will support several CRs). In addition a PROFINET alarm is also displayed in the “PnIoBoxDiag”

7 Appendix

7.1 Certifications

All products of the Embedded PC family are CE, UL and EAC certified. Since the product family is continuously developed further, we are unable to provide a full listing here. The current list of certified products can be found at www.beckhoff.com.

FCC Approvals for the United States of America

FCC: Federal Communications Commission Radio Frequency Interference Statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

FCC Approval for Canada

FCC: Canadian Notice

This equipment does not exceed the Class A limits for radiated emissions as described in the Radio Interference Regulations of the Canadian Department of Communications.

7.2 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages:
http://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49(0)5246/963-0
Fax:	+49(0)5246/963-198
e-mail:	info@beckhoff.com

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49(0)5246/963-157
Fax:	+49(0)5246/963-9157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49(0)5246/963-460
Fax:	+49(0)5246/963-479
e-mail:	service@beckhoff.com

8 List of tables

9 List of figures

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com