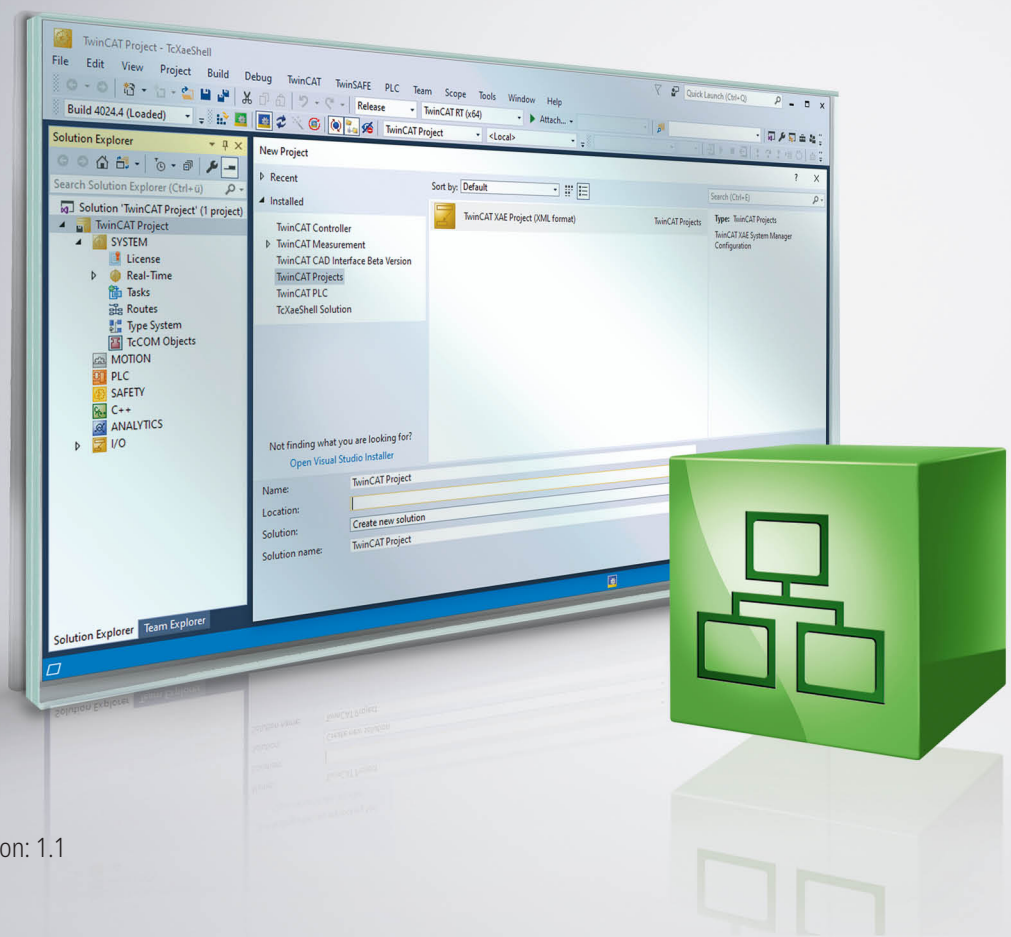
BECKHOFF New Automation Technology

Manual | EN

# TF6271

## TwinCAT 3 | PROFINET RT Controller




2021-06-22 | Version: 1.1

# Table of contents

# 1 Foreword

## 1.1 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®, TwinCAT/BSD®, TC/BSD®, 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

## 1.2 Safety instructions

### Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

### Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

### Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

### Description of symbols

In this documentation the following symbols are used with an accompanying safety instruction or note. The safety instructions must be read carefully and followed without fail!

**⚠ DANGER**

**Serious risk of injury!**

Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.

**⚠ WARNING**

**Risk of injury!**

Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.

**⚠ CAUTION**

**Personal injuries!**

Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.

***NOTE***

**Damage to the environment or devices**

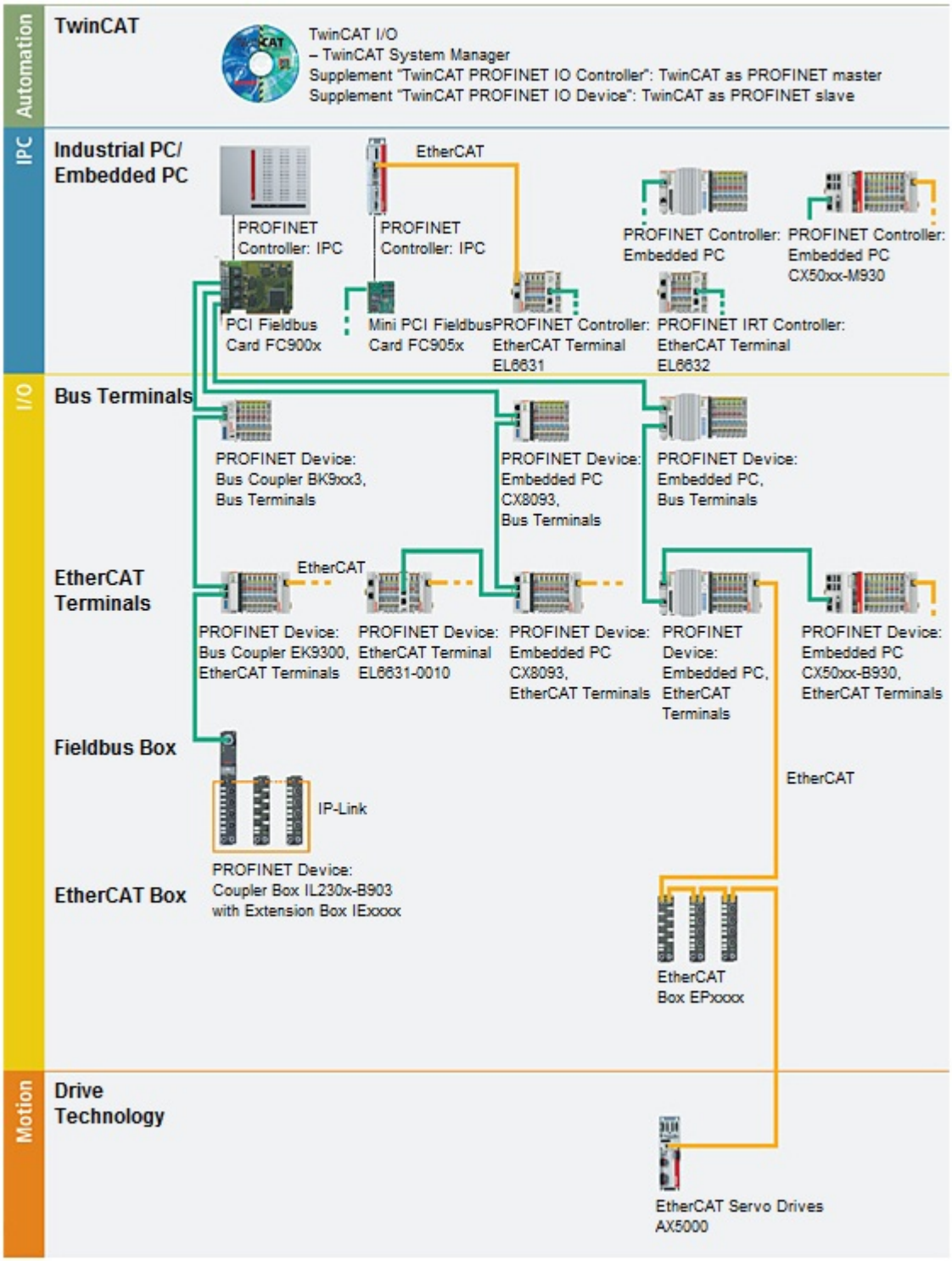
Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.

**Tip or pointer**

This symbol indicates information that contributes to better understanding.

# 2 PROFINET system presentation

PROFINET is the Industrial Ethernet standard of the PNO (PROFIBUS user organization). Internationally established IT standards such as TCP/IP are used for communication.




PROFINET system description

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. PROFINET RT can be operated with commercially available switches, but in this case topology detection is not supported. For better diagnostics and automatic name assignment (alias name), commercially available switches should be used that support at least topology detection; they are referred to as LLDP. All PROFINET devices from Beckhoff (with or without switch) have this feature.

Beckhoff PROFINET components

| Components | Comment |
|---|---|
| **TwinCAT** | |
| TwinCAT PROFINET IO Controller | TwinCAT as PROFINET master |
| **Embedded PCs** | |
| CX50xx-M930 | Embedded PC with optional interface PROFINET RT Controller |
| **EtherCAT Terminals** | |
| EL6631 | PROFINET IO controller |
| EL6632 | PROFINET-IRT controller |
| **PC Fieldbus cards** | |
| FC90xx | PCI Ethernet card for all Ethernet-based protocols (IEEE 802.3) |
| FC9x51 | Mini PCI Ethernet card for all Ethernet-based protocols (IEEE 802.3) |

# 3 Product overview

## 3.1 Function TF6271

TwinCAT PROFINET RT Controller (master) is a function that turns any PC-based controller with Intel® chipset and the real-time Ethernet driver developed by Beckhoff into a PROFINET RT Controller.

### 3.1.1 Technical data

| Technical data | TF6271 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Runtime** | TC1100 | | | | | | | |
| **Target system** | Windows XP, Windows 7/8/10, Windows CE | | | | | | | |
| **PROFINET version** | Conformance class B | | | | | | | |
| **Number of channels** | 1 | | | | | | | |
| **Ethernet-Interfaces** | 100BASE-TX Ethernet with 2 x RJ45 | | | | | | | |
| **Topology** | variable | | | | | | | |
| **Number of possible devices** | Limited by CPU power and memory | | | | | | | |
| **Max. number of process data** | Number and slave type etc., depending on cycle time | | | | | | | |
| **Cycle time (min.)** | RTC1 1 ms | | | | | | | |
| **Performance class (pp)** | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
| | – | – | x | x | x | x | x | x |

**Ordering information**

| | |
|---|---|
| TF6271-00pp | TC3 PROFINET RT Controller |

### 3.1.2 Requirements

#### Software

TF6271 is included in TwinCAT Version 3.1 Build 4020

#### Hardware

For using the TF6271, the target system has to have an Intel® network chipset. (See: Checking the hardware requirements)

**Beckhoff PC**

Beckhoff PC systems are usually preconfigured for the operation of PROFINET devices:

### 3.1.3 Licensing

The TwinCAT 3 function can be activated as a full version or as a 7-day test version. Both license types can be activated via the TwinCAT 3 development environment (XAE).

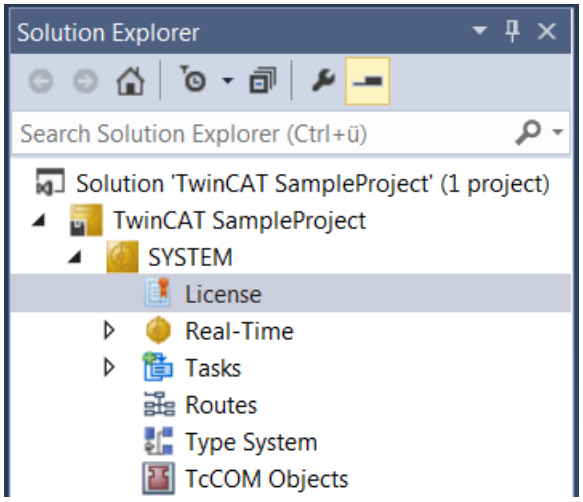
#### Licensing the full version of a TwinCAT 3 Function

A description of the procedure to license a full version can be found in the Beckhoff Information System in the documentation "TwinCAT 3 Licensing".

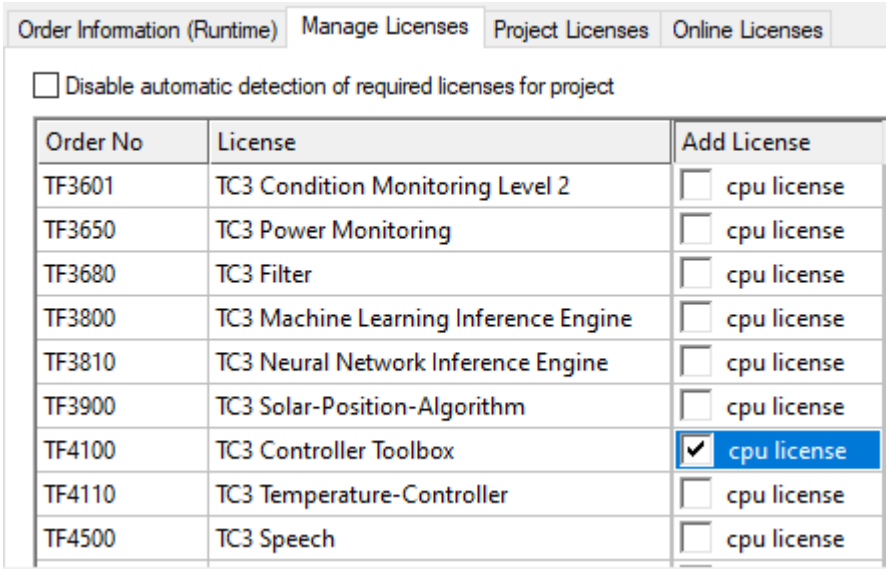
### Licensing the 7-day test version of a TwinCAT 3 Function

A 7-day test version cannot be enabled for a TwinCAT 3 license dongle.

1. Start the TwinCAT 3 development environment (XAE).
2. Open an existing TwinCAT 3 project or create a new project.
3. If you want to activate the license for a remote device, set the desired target system. To do this, select the target system from the **Choose Target System** drop-down list in the toolbar.
   ⇨ The licensing settings always refer to the selected target system. When the project is activated on the target system, the corresponding TwinCAT 3 licenses are automatically copied to this system.
4. In the **Solution Explorer**, double-click **License** in the **SYSTEM** subtree.

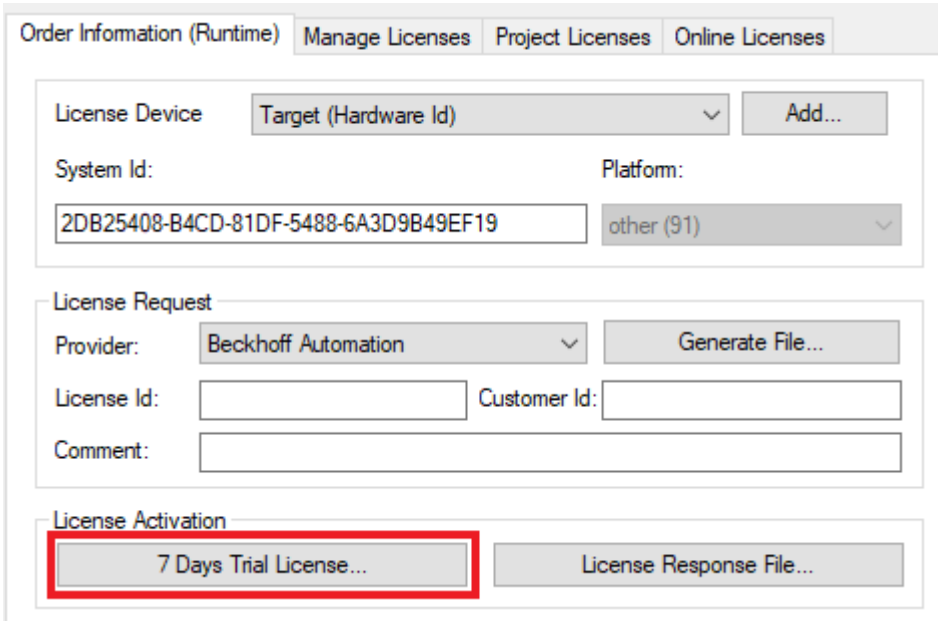



   ⇨ The TwinCAT 3 license manager opens.
5. Open the **Manage Licenses** tab. In the **Add License** column, check the check box for the license you want to add to your project (e.g. "TF4100 TC3 Controller Toolbox").

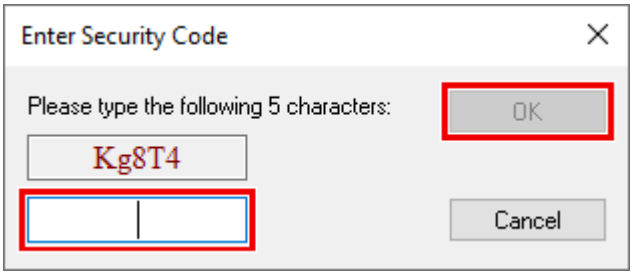



6. Open the **Order Information (Runtime)** tab.
   ⇨ In the tabular overview of licenses, the previously selected license is displayed with the status “missing”.

7. Click **7-Day Trial License...** to activate the 7-day trial license.




⇨ A dialog box opens, prompting you to enter the security code displayed in the dialog.




8. Enter the code exactly as it is displayed and confirm the entry.
9. Confirm the subsequent dialog, which indicates the successful activation.
   ⇨ In the tabular overview of licenses, the license status now indicates the expiry date of the license.
10. Restart the TwinCAT system.

⇨ The 7-day trial version is enabled.

## 3.2 Optional interface -M930

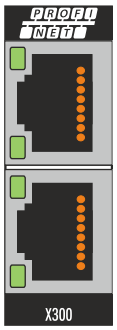
As an option, it is possible to order an Embedded PC with a fieldbus or a serial interface such as "PROFINET RT, Controller, Ethernet (2 x RJ-45 switch)". The optional interface must be ordered ex factory and cannot be retrofitted retrospectively.

### 3.2.1 Technical data

| Technical data | -M930 |
|---|---|
| Fieldbus | PROFINET RT Controller |
| Target system | Windows XP, Windows 7/8/10, Windows CE |
| PROFINET version | Conformance class B, optionally conformance class C |
| Number of channels | 2 (switched) |
| Ethernet-Interface | 100BASE-TX Ethernet with 2 x RJ45 |
| Topology | Variable |
| Number of possible IO devices | limited by CPU power and memory<br>max. 16 for CX9020-M930<br>max. 32 for CX50x0-M930<br>max. 64 for CX51x0-M930<br>max. 64 for CX20xx-M930 |
| Cycle time | RTC1 1 ms |
| Max. number of process data | Depending on cycle time, number and type of slave, etc. |

### 3.2.2 PROFINET connection

The optional interface is identified as "X300" on the devices and has as black border to identify it.

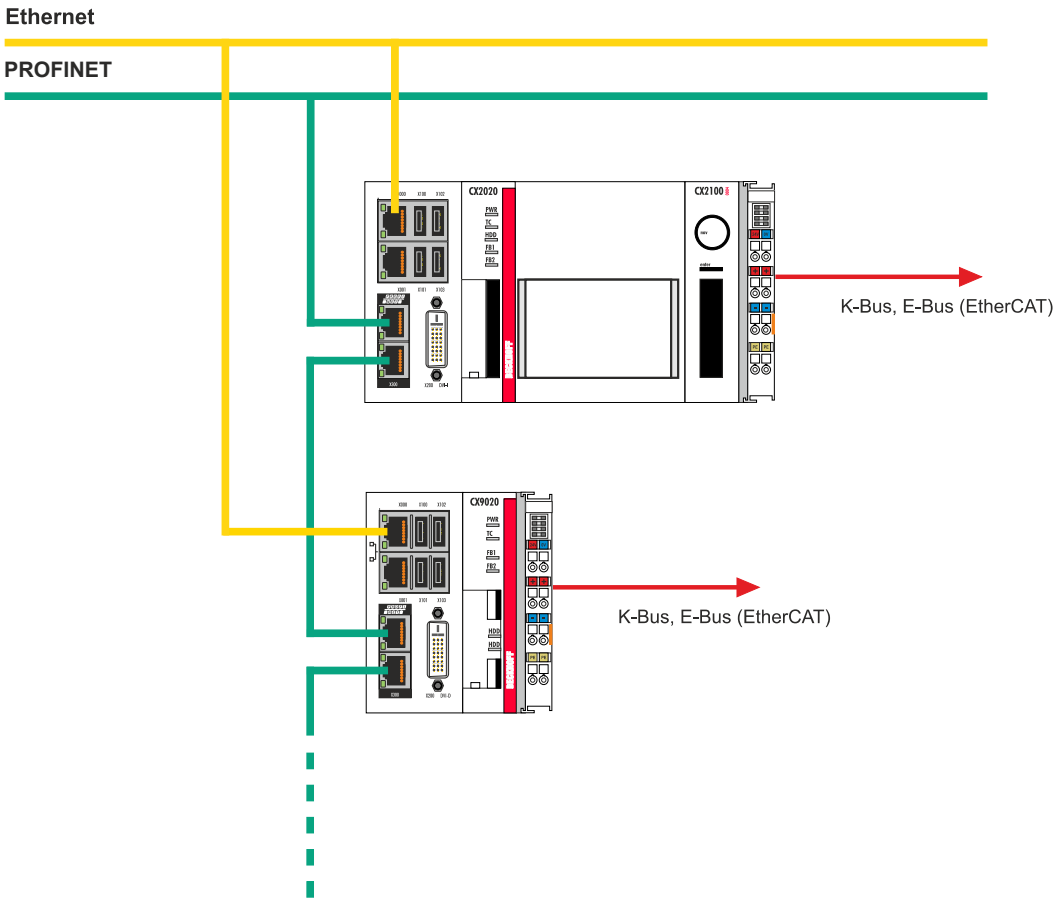



**LAN assignment (x300)**


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

### 3.2.3 Topology

Example for a PROFINET topology:




## 3.3 EL663x EtherCAT Terminal

The EL6631 PROFINET RT Controller (Master) terminal supports the complete real-time functionality (RT) as well as extensive diagnostic possibilities. All services according to conformance class B are supported. Up to 15 PROFINET RT devices can be projected on the EL6631.

The EL6632 PROFINET-IRT controller terminal offers the complete RT (real time) or IRT (isochronous real-time) functionality and a wide range of diagnostic options. All services in accordance with Conformance Class C are supported. Depending on the cycle time, up to five PROFINET-IRT or up to 15 PROFINET RT devices can be operated at the EL6632 in a line topology. The maximum distance between two devices is 100 m.

Protocols such as LLDP or SNMP can be used for network diagnostics.

### 3.3.1 Technical data

| Technical data | EL6631 | EL6632 |
|---|---|---|
| Technology | PROFINET RT | PROFINET IRT |
| Number of ports/channels | 2 (switched) | |
| Ethernet-Interface | 100 BASE Tx Ethernet with 2 x RJ45 | |
| Fieldbus | PROFINET RT controller | PROFINET IRT Controller |
| Cable length | Up to 100 m twisted pair | |
| Hardware diagnostics | Status LEDs | |
| Power supply | Via the E-bus | |
| Electrical isolation | 500 V (E-Bus/Ethernet) | |
| Max. number of IRT-capable devices | - | 5 (depending on the cycle time and number of data) |
| Max. number of RT-capable devices | 15 devices (depending on the cycle time and number of data) | |
| Min. RT cycle | 1 ms | |
| Min. IRT cycle | - | 500 µs |
| Conformance class | B | C |
| Protocol | RT | RT or IRT |
| Drivers | - | TwinCAT 2.11, TwinCAT 3 |
| Configuration | Via EtherCAT master | |
| Current consumption via E-bus | typ. 400 mA | |
| Network diagnostics | LLDP, SNMP | |
| Weight | Approx. 75 g | |
| Operating/storage temperature | 0...+55 °C/-25...+85 °C | |
| Relative humidity | 95% without illumination | |
| Vibration/shock resistance | conforms to EN 60068-2-6/EN 60068-2-27 | |
| EMC immunity/emission | conforms to EN 61000-6-2/EN 61000-6-4 | |
| Protect. class / installation pos. | IP 20/see documentation | |
| Approvals/markings | CE, UL, ATEX | |

## 3.4 Technical data

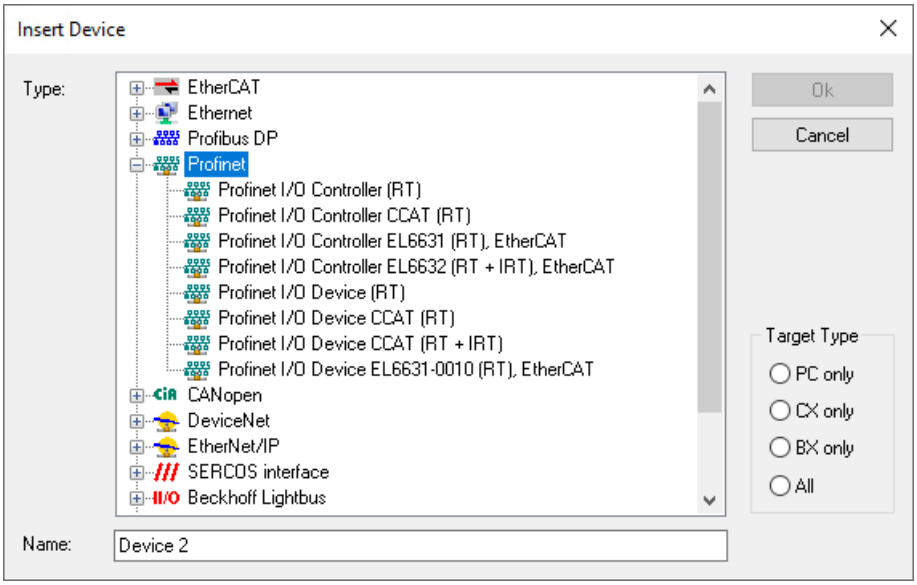
| Technical data | RT Ethernet (TF6271) | Optional interface M930 | EtherCAT (EL6631) | EtherCAT (EL6632) |
|---|---|---|---|---|
| Ethernet-Hardware | RT Ethernet hardware | RT Ethernet hardware | EL6631 PROFINET terminal | EL6632 PROFINET terminal |
| Operating system | Windows XP, Windows 7/8/10, Windows CE | | | |
| Software | TwinCAT I/O, PLC, NC, NC I, CNC | | | |
| Target systems | PC (x86), Windows CE devices | | PC (x86), Windows CE devices with EtherCAT interface | |
| Cycle time | min. 1 ms | | | min. 500 µs |
| Number of possible IO devices | Limited by CPU power and memory | | Up to 15 devices | For IRT up to 5 devices, for RT up to 15 devices |
| Max. number of process data | Number and slave type etc., depending on cycle time | | 1 kbyte input data and 1 kbyte output data | |

# 4 Commissioning

## 4.1 Protocol selection

PROFINET devices must be attached directly to the I/O device, regardless of whether it is a controller (master) or ordinary device (slave). Exceptions are optional interfaces (-B930, -M930), which can be added via the Scan function (CCAT).

Right-click on **I/O - Devices** > **Add New Item...** to select from four different PROFINET I/O Controller protocols.

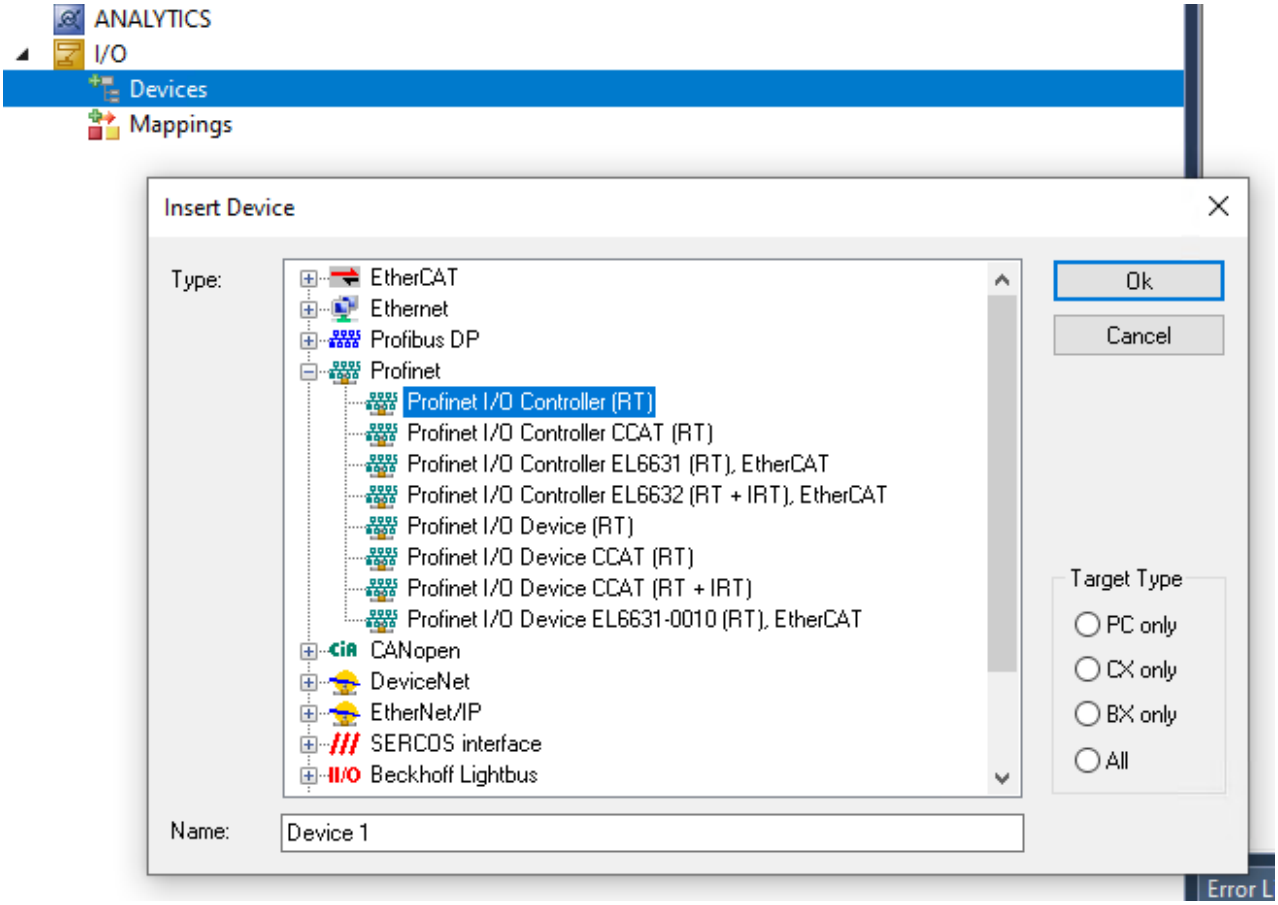



with the purposes described below.

- Profinet I/O Controller (RT): Uses the function TF6271 in conjunction with an Intel chipset
- Profinet I/O Controller CCAT (RT): Uses CCAT-based hardware (e.g. CXxxxx or FC932x) with RT (RTC1) ordering option
- Profinet I/O Controller EL6631 (RT): Uses the EL6631 EtherCAT PROFINET gateway terminal
- Profinet I/O Device EL6632 (RT+IRT): Uses the EL6632 EtherCAT PROFINET gateway terminal

### 4.1.1 Integration via a Real-Time Ethernet interface

Select the PROFINET I/O Controller (RT) according to the configuration. The available adapters are displayed right away when you add them. If these are to be modified or checked afterwards, this can take place on the **Adapter** tab.

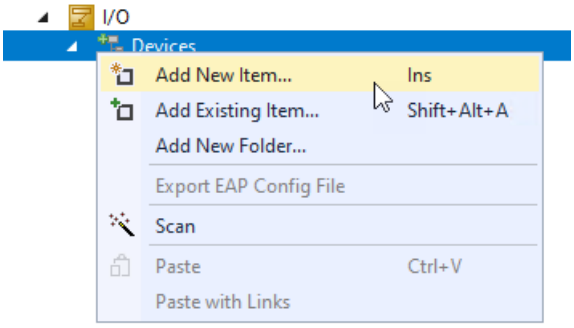


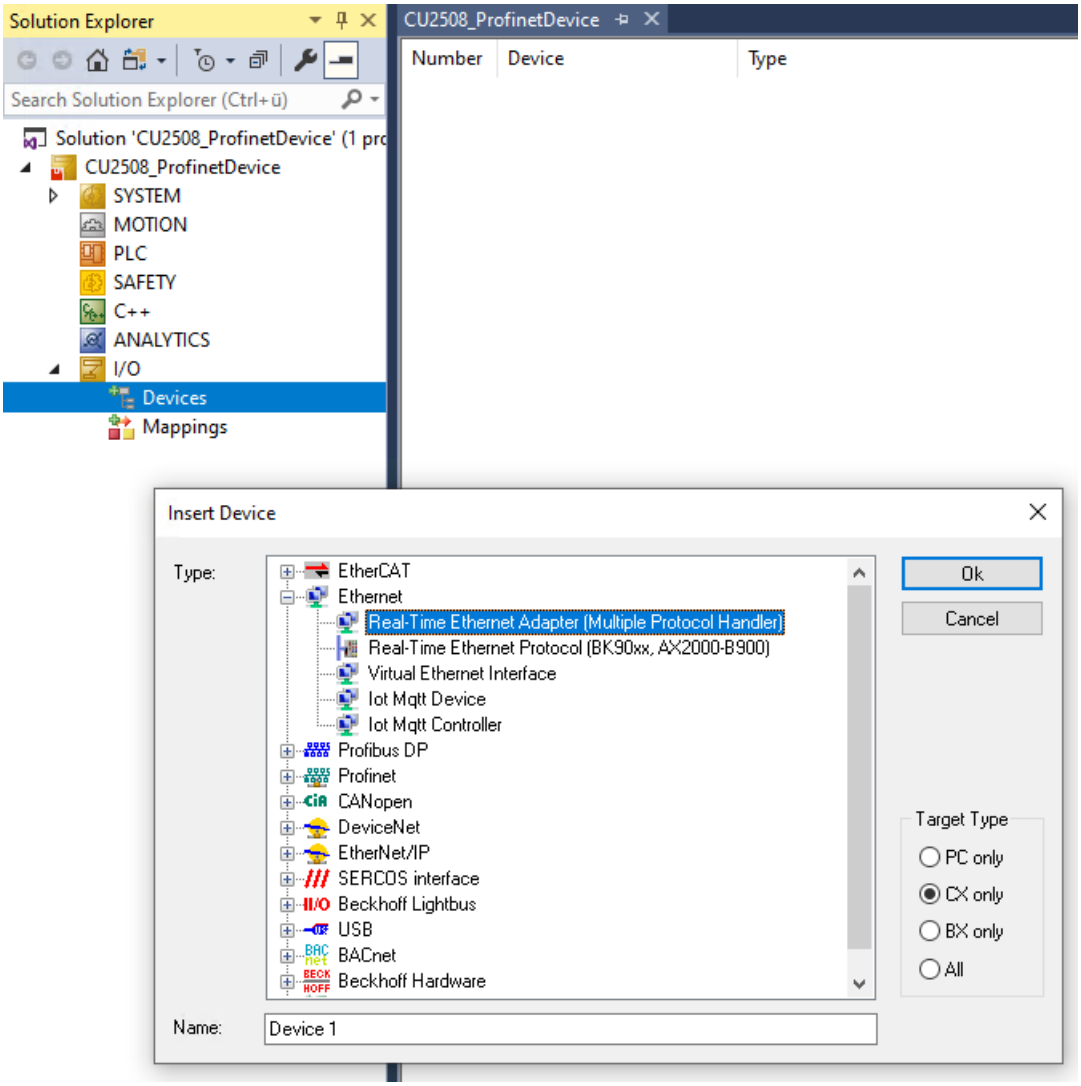

Appending PROFINET devices [▶ 24]

## 4.1.2 Integration via a CU2508 real-time Ethernet port multiplier (Real-Time Ethernet interface)

The following section illustrates how to configure a PROFINET controller using the CU2508 real-time Ethernet port multiplier. However, you can use only one controller or device, if they are in a unique network. If the networks a physically separated, it is possible to use more than one controller or device supplement with one CU2508. This is due to the fact, that the CU2508 uses only one MAC address. The function of several PROFINET segments can only be secured, if the MAC address in a network segment is unique and there is no connection to other networks.

Add the CU2508 real-time Ethernet port multiplier to your TwinCAT project by right-clicking on I/O - **Devices** > **Add New Item** and adding the Real-Time Ethernet Adapter (Multiple Protocol Handler) in the window that opens.

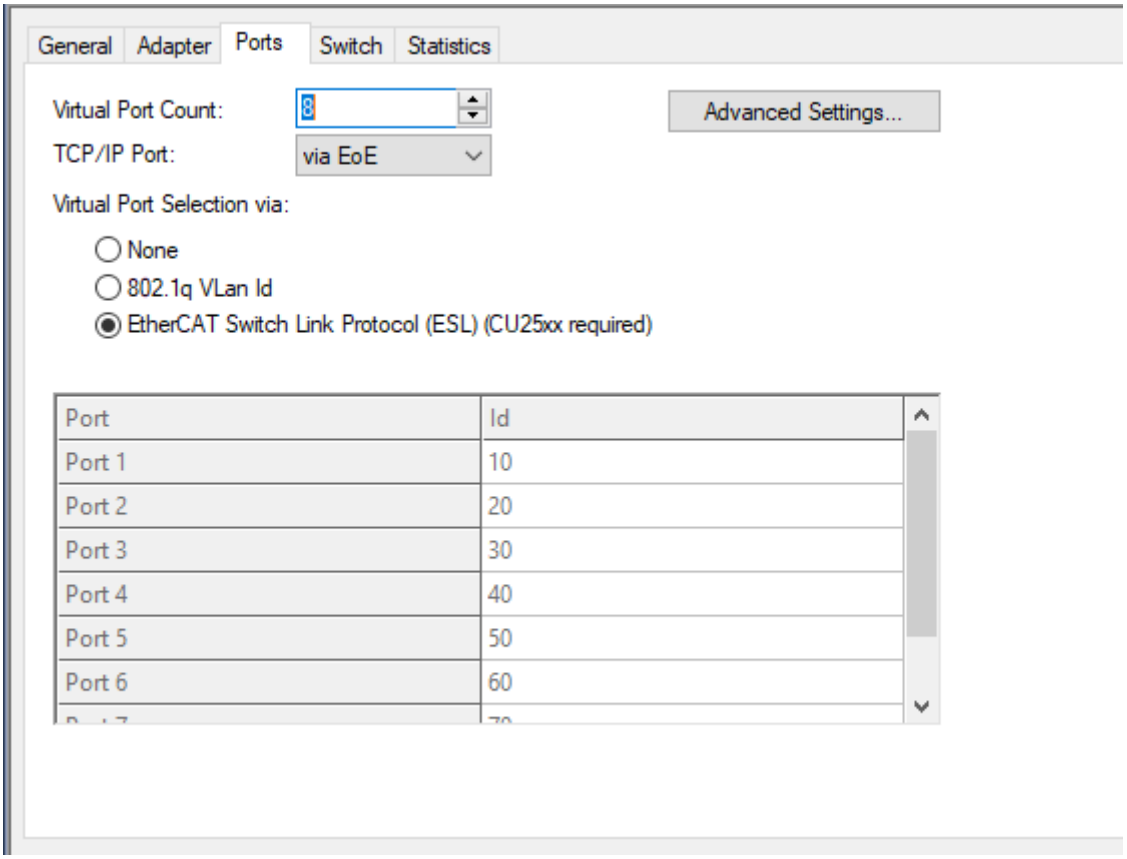




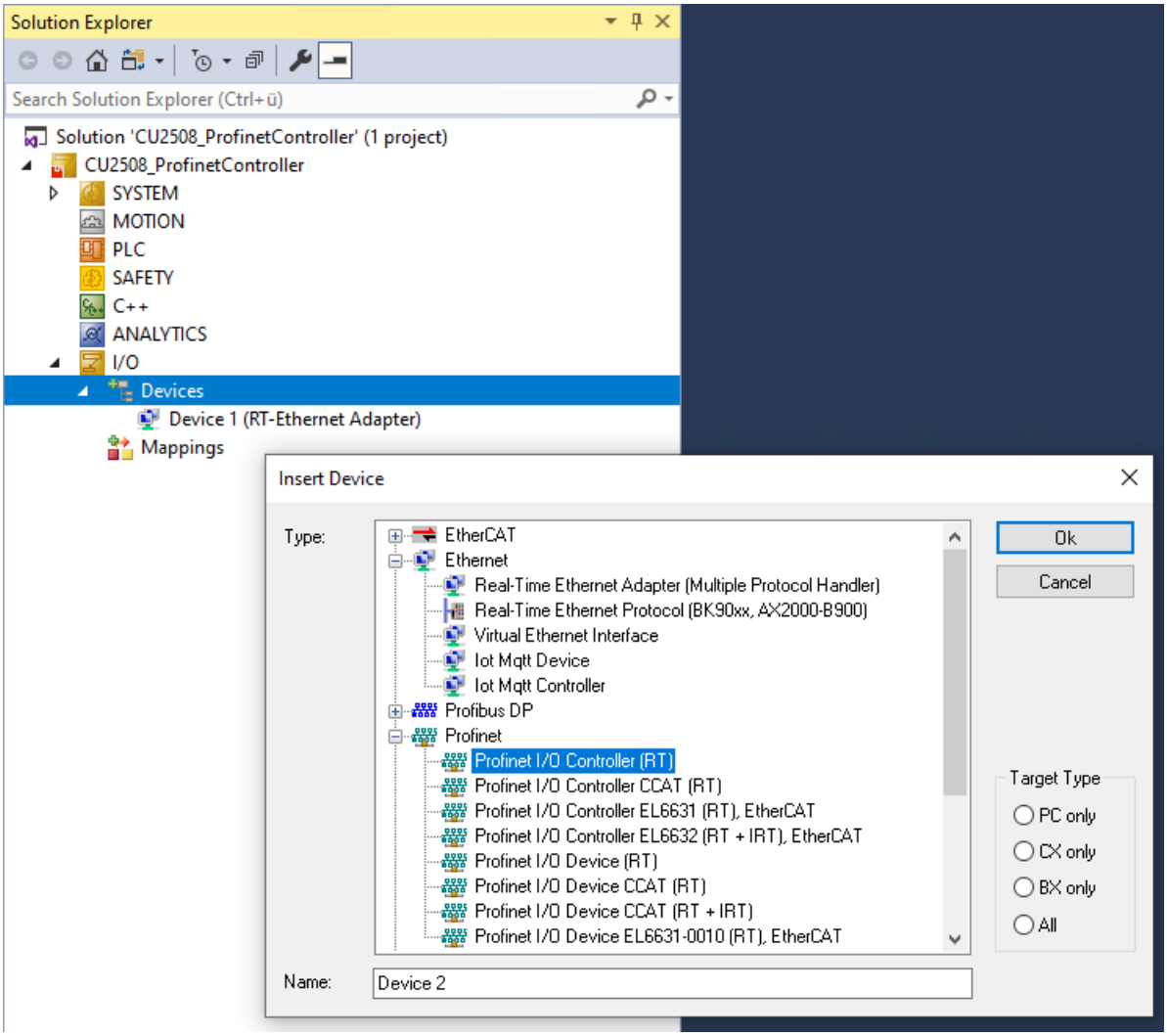

Furthermore, the following settings on the **Ports** tab are necessary for the operation of the CU2508

Virtual Port Selection via:

EtherCAT Switch Link Protocol (ESL) (CU25xx required)



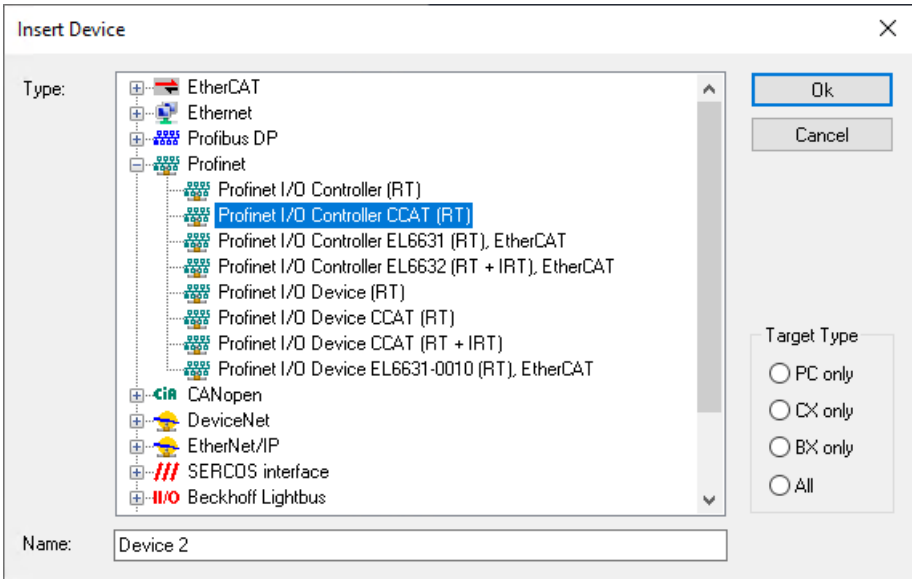

The PROFINET protocol is directly added to the I/O device. Four different PROFINET I/O devices are available for selection. Select the PROFINET I/O Controller (RT) according to the configuration with the CU2508.




Appending PROFINET devices [▶ 24]

### 4.1.3 Integration via an optional interface, -M930

Select the PROFINET I/O Controller (CCAT) according to the configuration. The available adapters are displayed right away when you add them. If these are to be modified or checked afterwards, this can take place on the **Adapter** tab. The optional interfaces (-M930) can be added via the "Scan function" (CCAT).

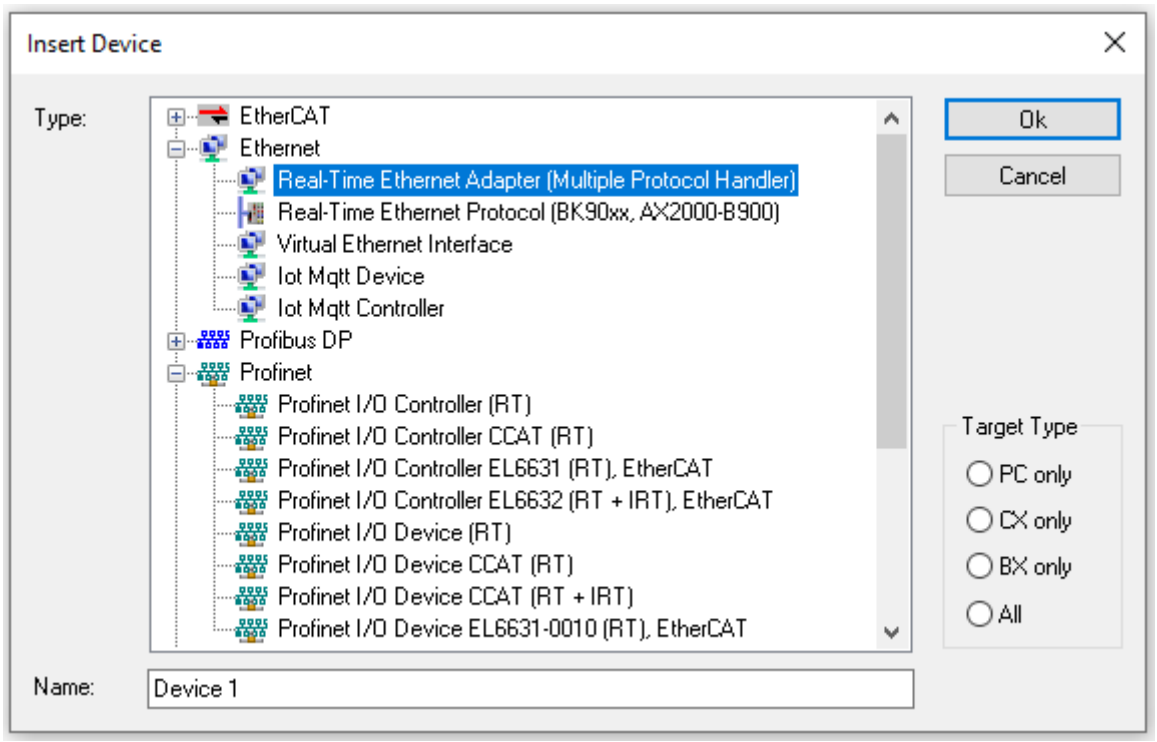



Appending PROFINET devices [▶ 24]

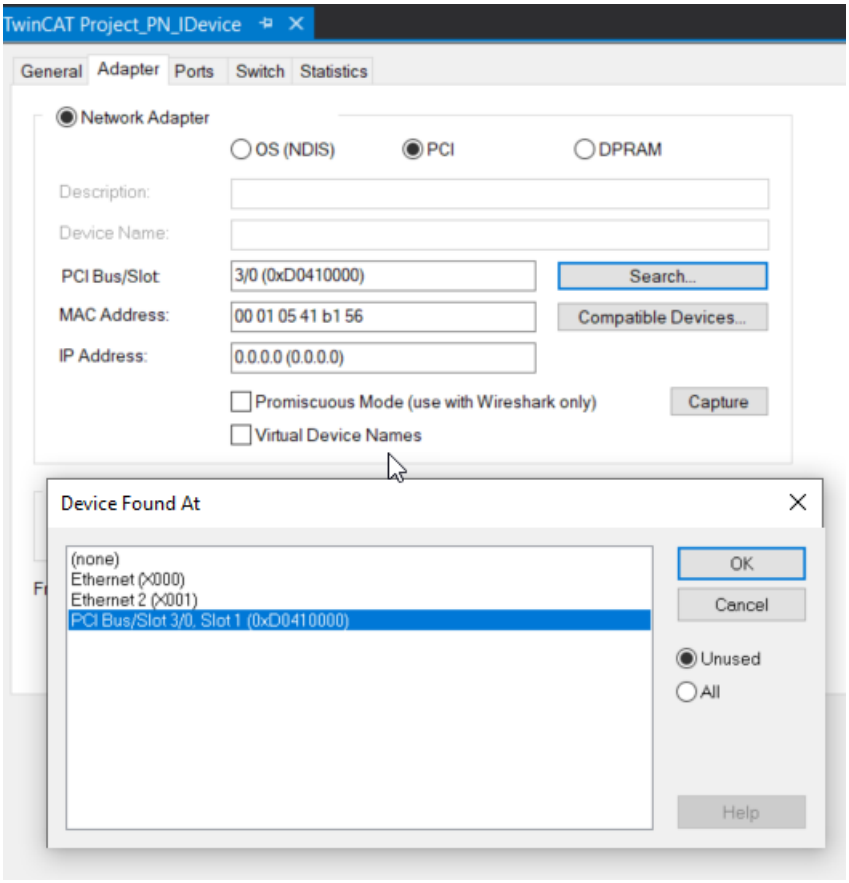
### 4.1.3.1 Creating an I-Device

Using the I-Device functionality, a controller and a device interface can be operated simultaneously via a physical interface. The CCAT -M930 interface and TwinCAT version 4024.13 or higher are required

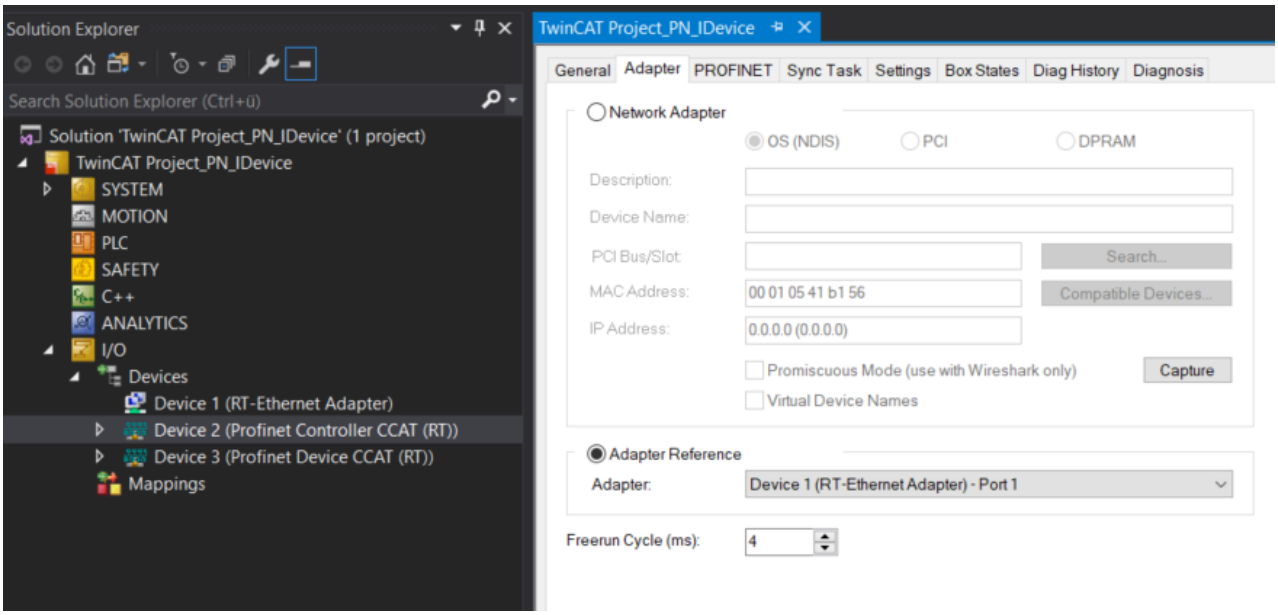
A **Multiple Protocol Handler** must be used for project planning.



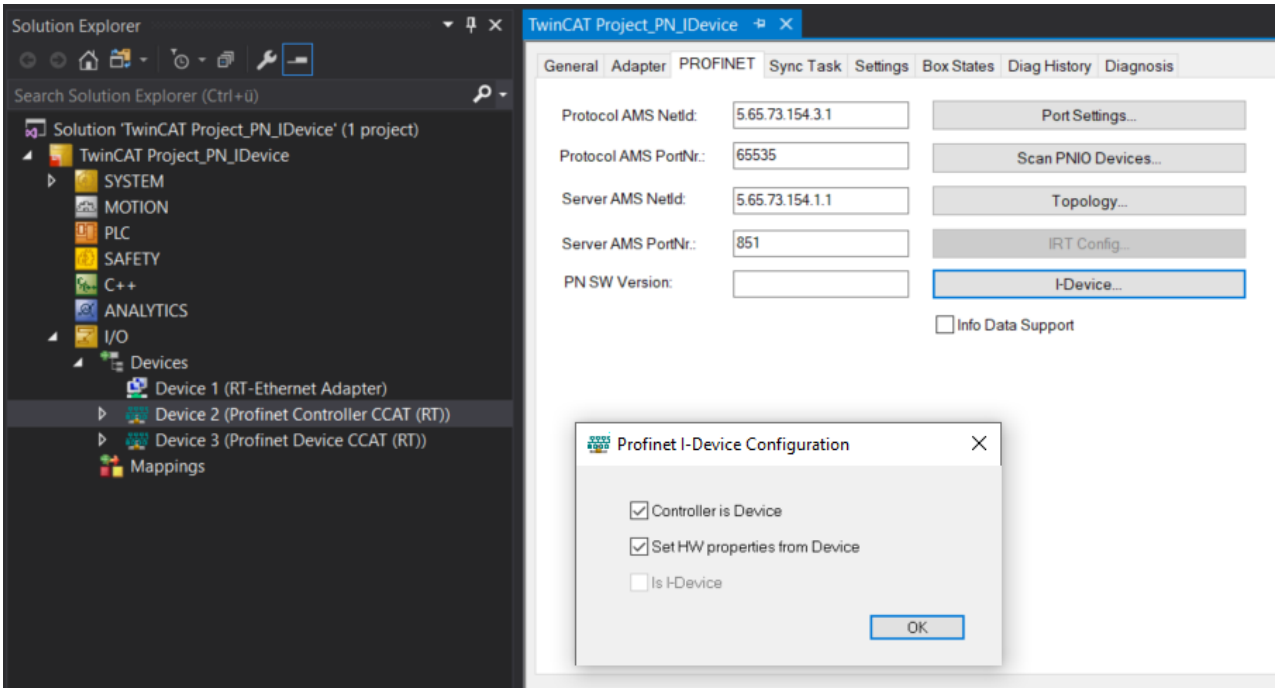

The CCAT interface must be assigned as the adapter.




The PROFINET CCAT controller and device are then projected. For both, the assignment must be made using the same adapter reference to the Multiple Protocol Handler.



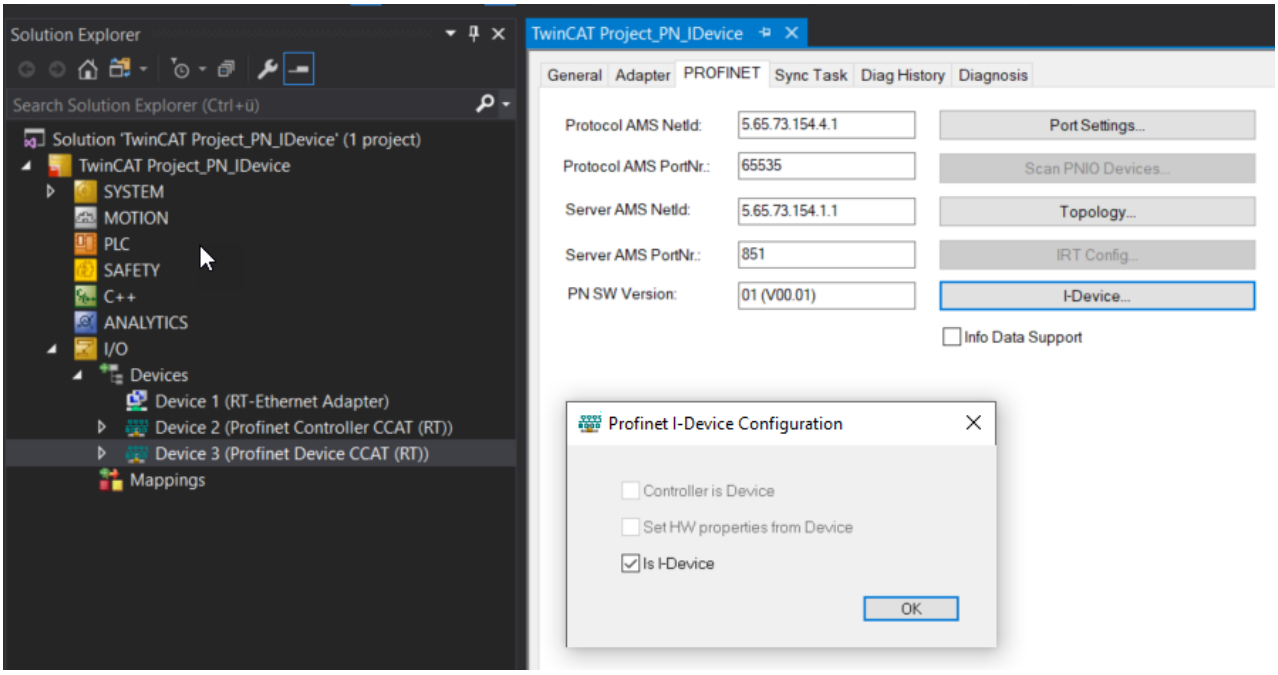

For both protocols the I-Device functionality has to be enabled. The following selection is available on the controller.




This enables the controller as a device at the same time.

Here you can optionally define the hardware properties, e.g. what is signaled via the fieldbus LED. By default this is the controller diagnostics, although it is possible to change to device. This option also affects the port properties (e.g. LLDP). When enabled, the higher-level controller can parameterize the properties of the physical interface.

The I-Device must still be enabled on the Device protocol.



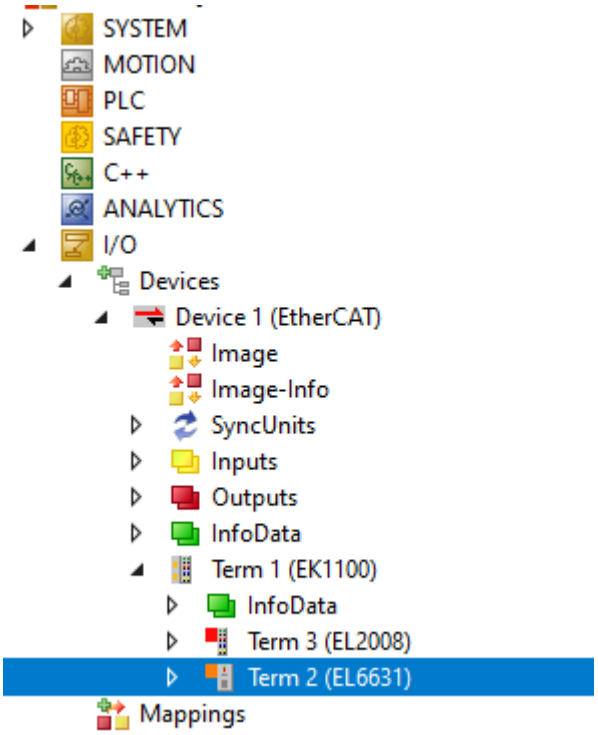

If several PROFINET Device protocols exist in the project, coupling to the corresponding I-Device Controller takes place.

Virtual MAC addresses are assigned on the device side (including the first box). When using multiple I-Device configurations in the same network, the uniqueness of the virtual MAC addresses must be ensured. The last digit of the MAC can be edited for this purpose.

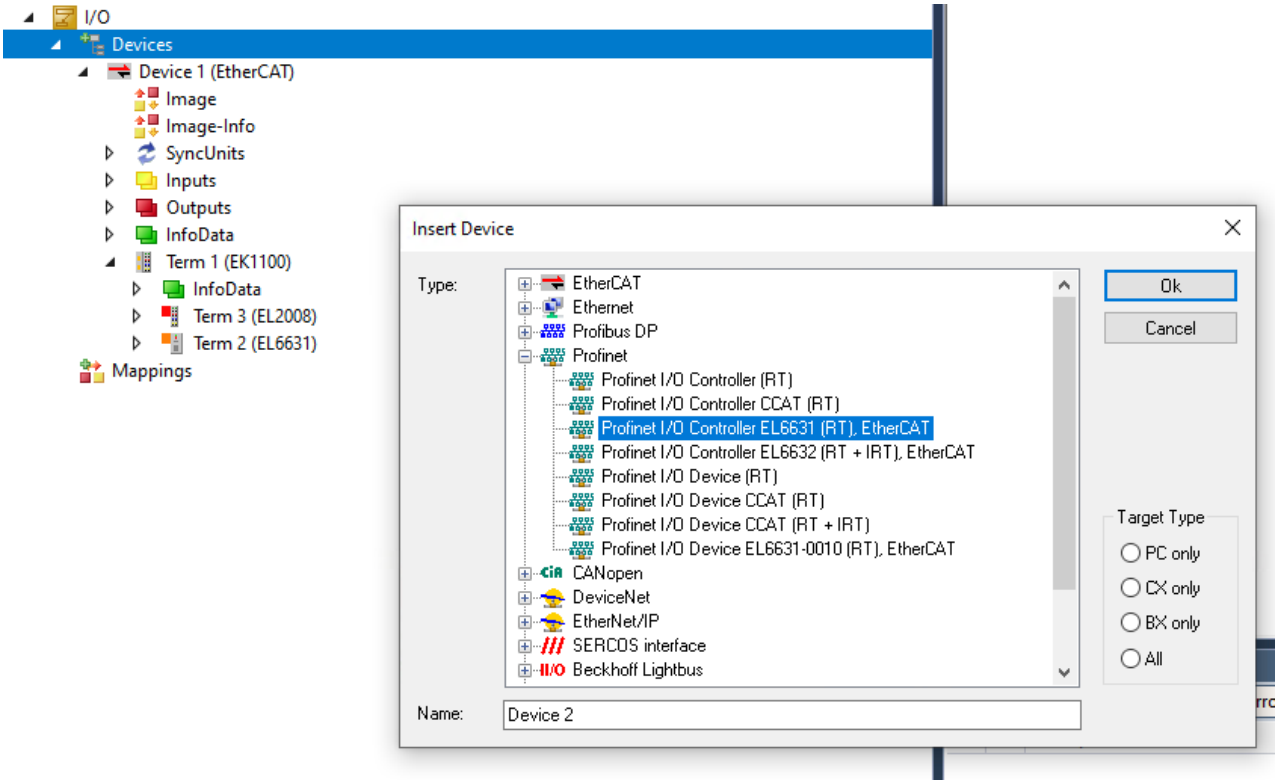
### 4.1.4 Integration via an EL663x

The following section shows how a PROFINET Controller protocol is integrated using the EL6631 PROFINET gateway terminal.

The EL663x is primarily an EtherCAT slave, which means that it must be included in the TwinCAT configuration, since it serves as an adapter for the PROFINET Controller.




The controller protocol is appended directly to the I/O device. Select the appropriate protocol according to the configuration with the EL663x. If such a terminal is located on the projected EtherCAT segment, the associated adapter is entered directly when the protocol is added. For the operation of several EL663x terminals the corresponding PROFINET protocol must be appended several times. If the terminal assignment is to be modified or checked afterwards, this can take place on the **Adapter** tab.




Appending PROFINET devices [▶ 24]

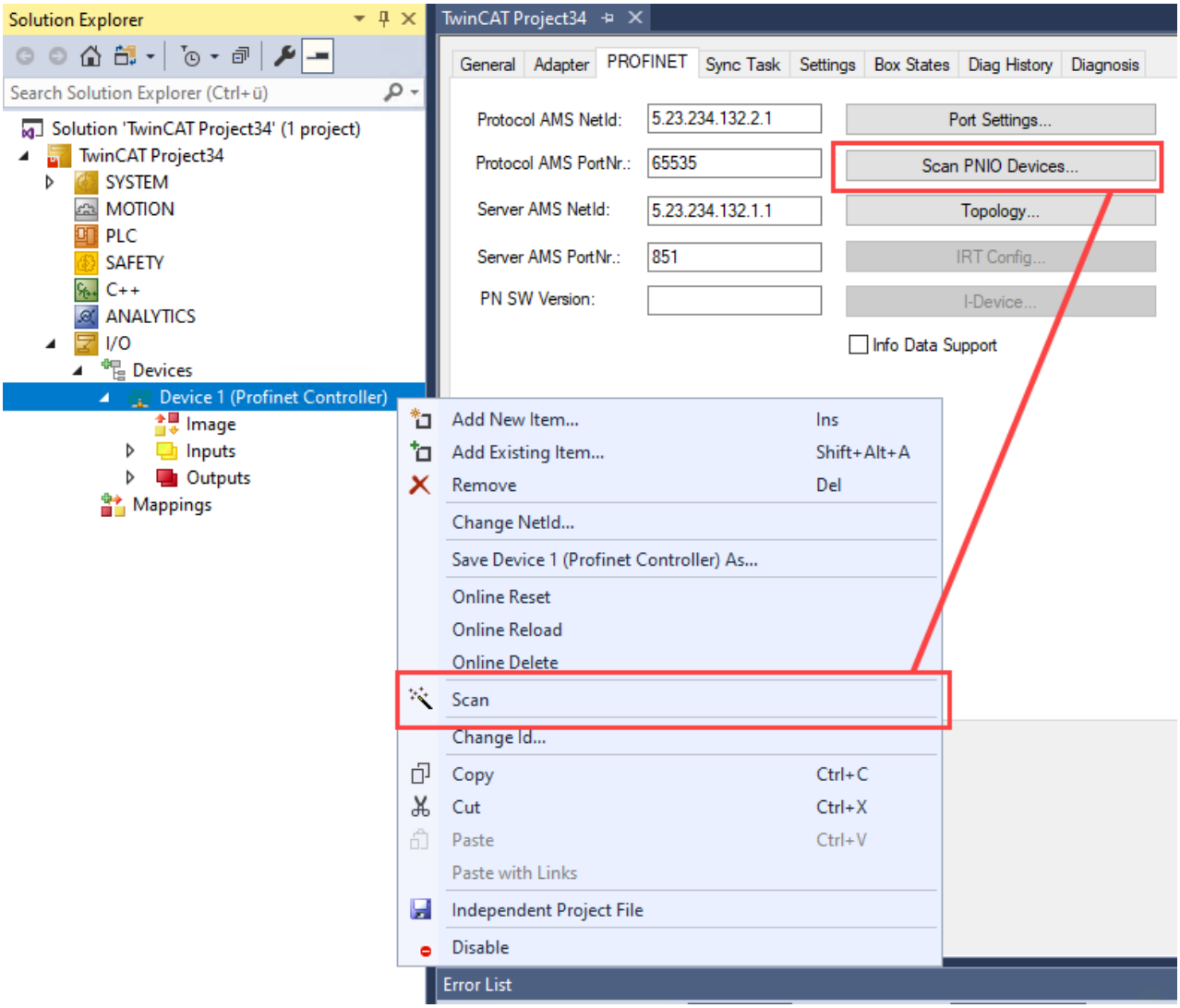
# 4.2 Configuration

## 4.2.1 Appending PROFINET devices

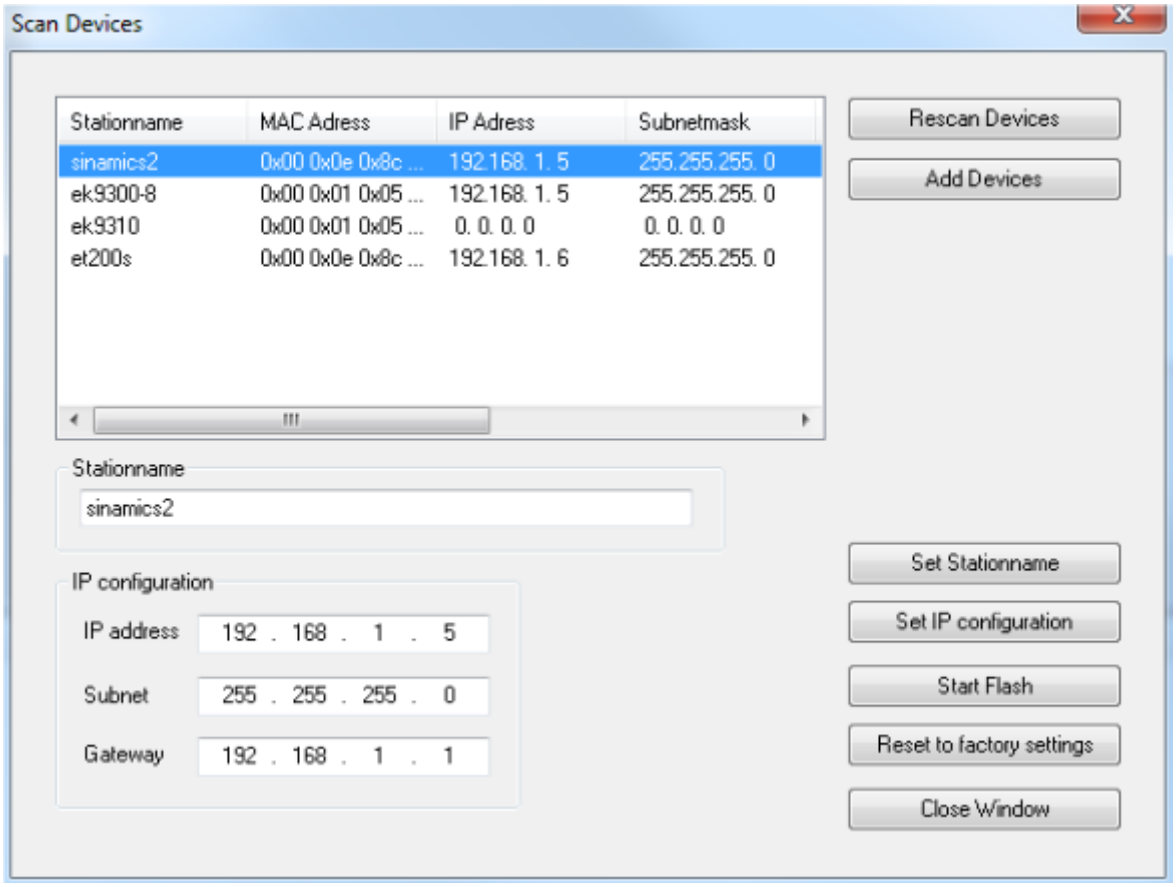
Once the PROFINET Controller has been created, there are two options for adding new PROFINET devices.

### 4.2.1.1 Appending via the Scan function

The first and recommended option is to use **Scan PNIO Devices**. This feature is comparable with the "ScanBoxes" feature which, however, is available only in CONFIG mode. After successful scanning, if PROFINET devices were found a dialog opens in which settings and configuration can be made on the devices and devices with module configuration can be added.

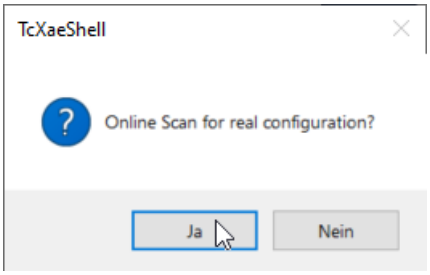



The prerequisite for finding PROFITNET devices on the bus is that they are present and switched on and that the GSDML file of the respective devices is located in the path *C:\TwinCAT\3.1\Config\Io\Profinet*.




Various settings or project planning can be carried out for the devices here. These are adopted only when the corresponding button is explicitly pressed. When setting the name, care must be taken that only PROFINET-compliant characters are used. This also applies to the IP address; only valid combinations of IP and subnet are to be used. Name and IP are checked for correctness when setting PROFINET devices. If this is not the case, DCP_SET is acknowledged with an error. By pressing the **Rescan** button, any changes that were made can be read back. In addition the selected device can be signaled. This functionality is PROFINET-specific. The signaling is vendor-specific. As standard, however, the signal must arrive with a frequency of 2 Hz. For example, the Beckhoff BK9103 Bus Coupler signals itself by the alternate flashing of two LEDs at a rate of 2 Hz. This function is very helpful for identifying the devices in this list. The flashing is stopped again by pressing the button once more. The flashing is stopped by closing the **ScanDevices** window. Subsequently, one or more devices can be marked with the Ctrl button. The selected devices are adopted into the project by pressing **Add Devices**

Upon pressing **Add Devices**, the following question box opens:




**Yes** button:

An attempt is initially made to determine the ModuleIdentNumber of the DAP (Device Access Point) by an implicit read access. If this fails a corresponding dialog opens containing the possible DAPs, which must then be selected manually. If all boxes have been appended, a ‘Reload Devices’ automatically takes place, i.e. the devices (adaptors) created are transmitted to the PROFINET driver. Subsequently, a distinction is made as to whether the box is a normal device or a drive with Profidrive support. For a normal device the actual module population (RealIdentificationData) is read out again via an implicit read access. If it is a

Profidrive device, conversely, the required information is read out by a Profidrive access. A Supervisor AR is established for this purpose. The required write accesses can take place within this. The Submodule interface on the DAP is taken here as the Parameter Access Point. The parameter access takes place via data record 47, much like the case of the Profibus beforehand. When using Sinamics, however, it must be noted that such an access is only permitted from version 4.3 SP2. If an older version is used, a corresponding error message appears and the parameterization must take place manually.

Once the automatic module parameterization has been completed, a question box appears asking whether the port data should be read in automatically. Here again, the port connection for the individual devices is read out via an implicit read access. The real port connection must be known for the various services. This can be limited to diagnostic services, although automatic device startup also requires this option (via alias), or specification of IRT planning. If this dialog is acknowledged with "no" or if the read access has failed, such a connection can also be made manually at the individual ports in the TwinCAT project.

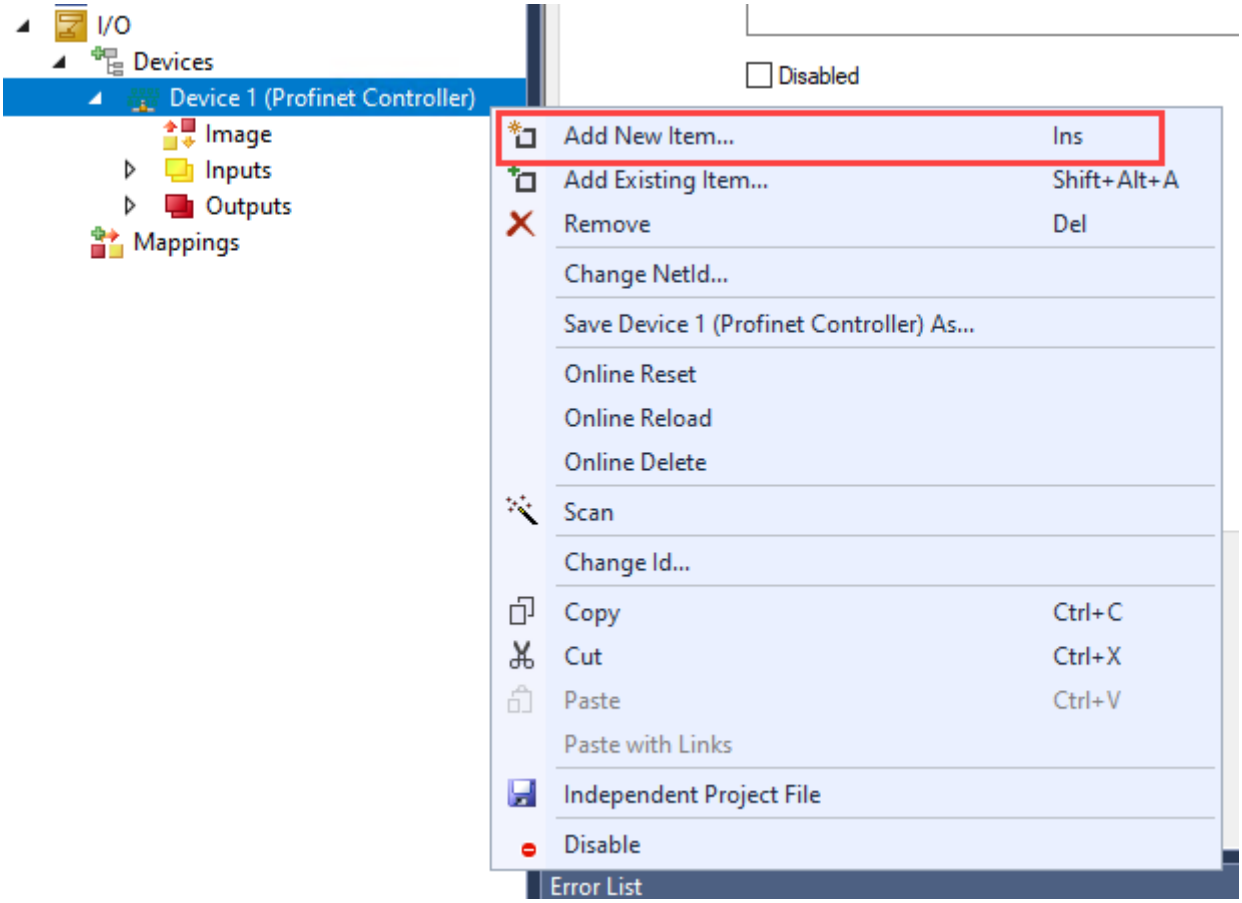
If the port connection has been successfully generated, a query also appears in the case of an IRT Controller (e.g. project planning on an EL6632) asking whether all devices are to be automatically connected in IRT Mode (provided they support it). If this is affirmed the cable length is additionally set to 10 m copper cable on all projected ports. The IRT algorithm requires this information for the calculation of the signal propagation delays. The precise cable length is not so important here (approx. +/-10 m), because the runtime delays tend to be small at 100 Mbit/s (5 ns/m). If the automatic switchover is not to take place immediately, these points can also be modified later either on the protocol or on the individual devices (on the interface or port submodule).

**No** button:

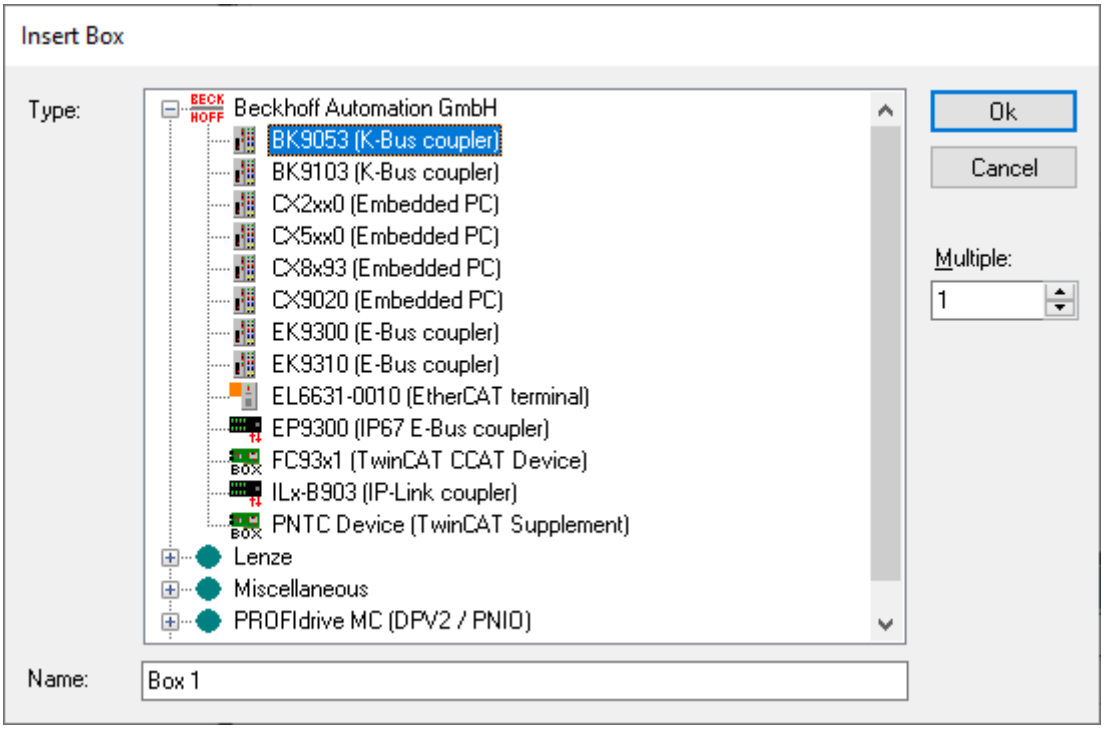
For each device a check is performed to see if the GSDML exists in the appropriate folder (..\ *TwinCAT\3.1\Config\Io\Profinet*). If this is the case, the list of possible DAPs is read in. Subsequently a selection dialog is opened so that the corresponding DAP can be selected. Once the devices have been appended in the project, the API below the Box can be accessed and the modules and submodules can be manually appended via it.

#### 4.2.1.2 Appending via the dialog selection

The second configuration option involves adding the individual devices via the project tree. Right-click on the created PROFINET Controller and select **Add New Item**.




The following dialog opens.




There is a possibility here to select various PROFINET devices. The Beckhoff devices search for the GSDML under a defined path (..*TwinCAT\3.1\Config\Io\Profinet*). These should be already present with the TwinCAT installation. If there are several GSDMLs for the same device here, the one with the latest date is taken. If no device description is found, a corresponding error message appears. Either the GSDML is copied into the

folder and the menu is opened again, or the same procedure is selected as for the third-party devices. Click on **PROFINET IO Device** to navigate to the corresponding GSDML in Windows Explorer. This is then integrated into the project.

The DNS name from the GSDML is taken as the default name. When adding several devices at the same time, the default name is always supplemented by "-No." (where No. = 1..n). The assigned name (under which the device appears in the tree) is also the PROFINET Station Name, i.e. the name that must match the name in the device. The device name can be checked by scanning.

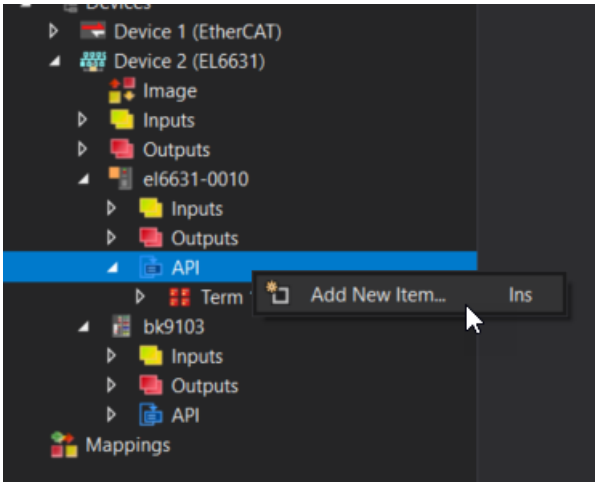
The modules can be attached to the API (Application Profile Interface). The DAP (Device Access Point), which already brings along fixed properties from the GSDML (e.g. process data, interface and port submodules, etc.), is always on Slot 0. This module is always present and cannot be deleted or moved. Each further module is assigned to a certain API. The information regarding its identity comes from the GSDML. By default this is always API 0. Alternatively, an API e.g. for the PROFIDRIVE profile or a fieldbus API is also conceivable. In the API click on **Append PROFINET Module...** to open a device catalog from which you can select and add the required modules. If the modules support it (described in GSDML), the submodules can in turn be appended to them in the same way.

## 4.2.2 Creating modules/process data

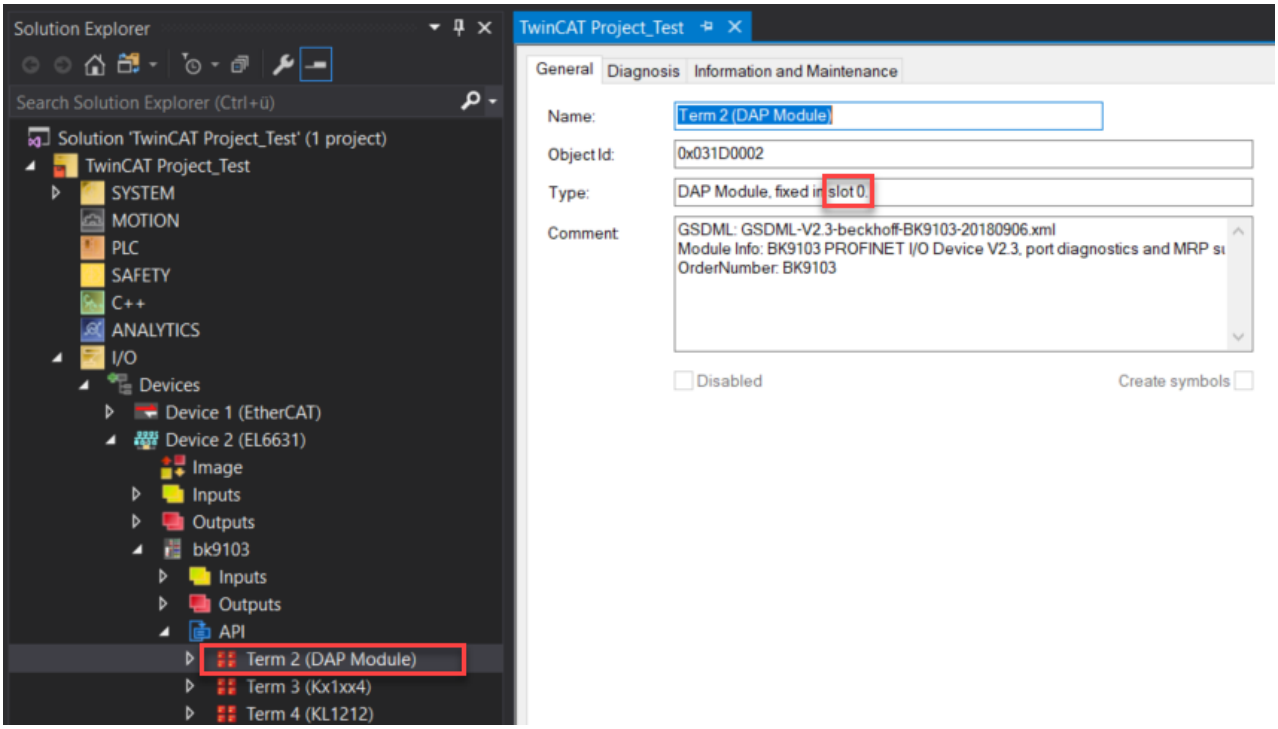
The procedure for creating modules/process data on a PROFINET Controller is the same for all controller types. In the following sections, the BK9053 Bus Coupler and the EL6631-0010 EtherCAT Terminal are used as PROFINET slaves to illustrate the addition of process data in various forms.

### 4.2.2.1 Up to TwinCAT Version 3.1 Build 4024

The modules can be attached to the API (Application Profile Interface). Open by right-clicking on **Add New Item**




The order of the modules in the tree always corresponds to the populated slot, starting at 0.

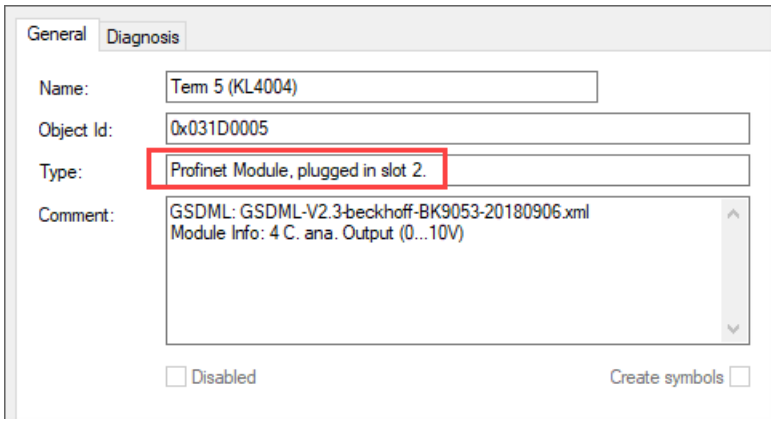



The DAP (DeviceAccessPoint) is added directly with the device. If, according to GSMDL, this is not plugged into slot 0, empty placeholder modules are inserted before it. The DAP is a special module that comes with device properties from the GSDML. The PDEV properties are attached to it in the form of submodules (interface and port). In addition, the DAP can also contain normal submodules with process and record data. It is always fixed and cannot be deleted.

Each additional module is assigned to a specific API. The information regarding its identity comes from the GSDML. By default this is always the API 0. Alternatively, an API e.g. for the PROFIDRIVE profile or a fieldbus API is also conceivable.

If the modules (described in GSDML) support it, the submodules can be projected below them. The subslots are also simply numbered continuously, starting at 1 (modules at 0). The PDEV submodules (interface and port) are exceptions; these are plugged into a fixed subslot specified via the GSDML.

The current slot or subslot can be checked via the associated object.




### 4.2.2.2 From TwinCAT Version 3.1 Build 4024

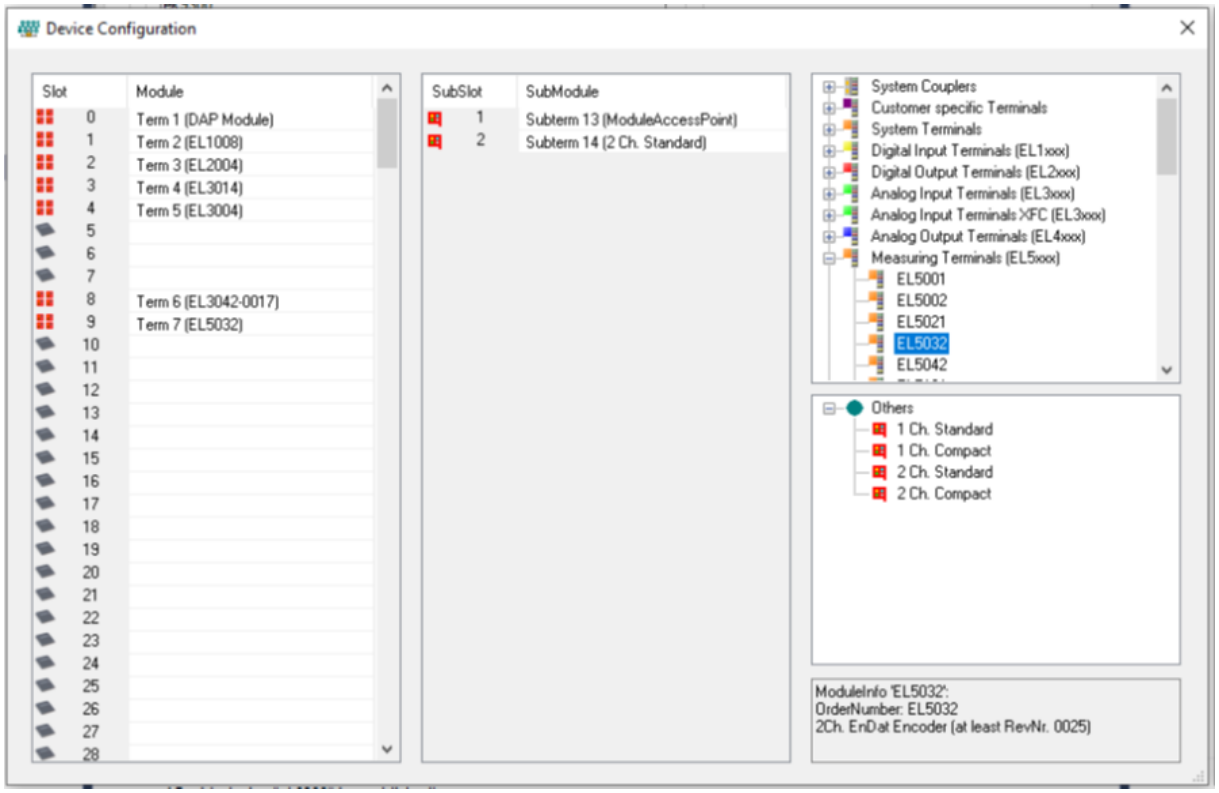
With TwinCAT Version 4024, the classic tree view for module/submodule population was replaced by a rack view.

In this view slots can be populated and removed without shifting the following slots. Empty slots can remain free and do not have to be provided with placeholders.

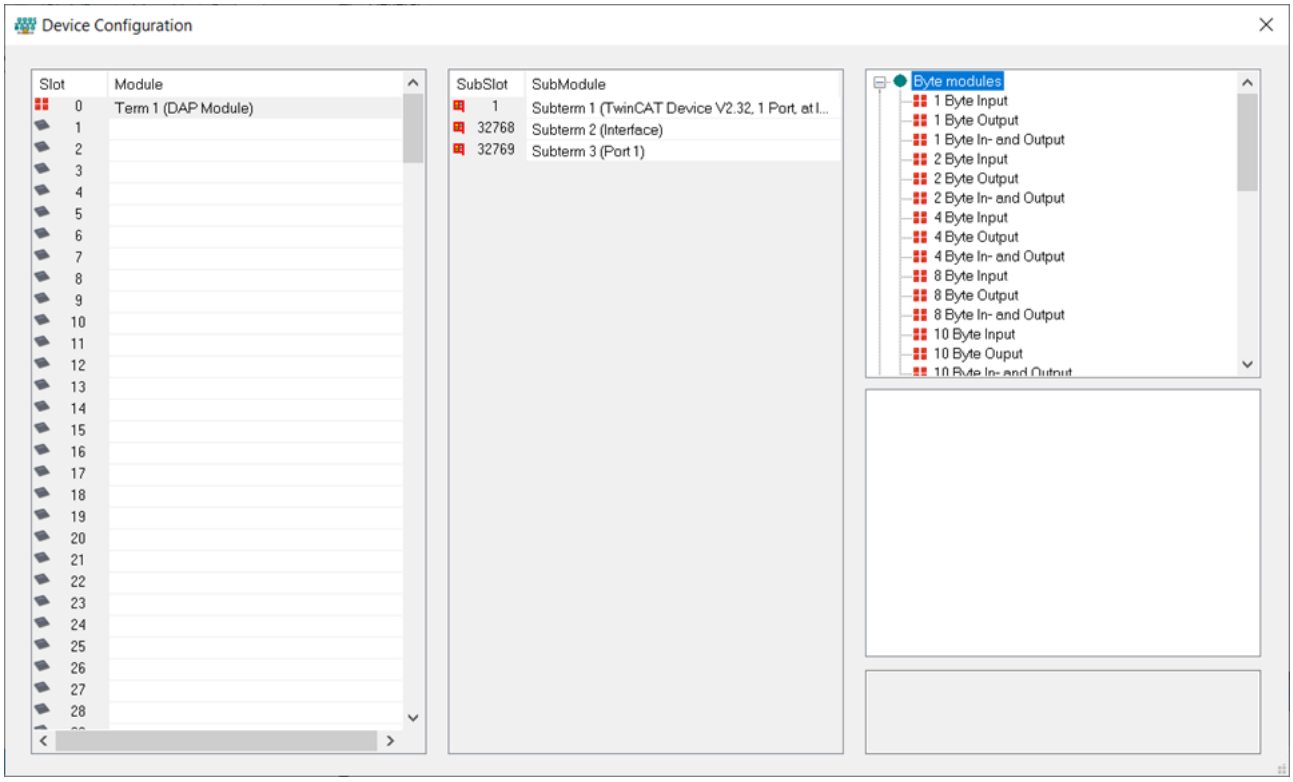
Furthermore, it is possible to generate the created module configuration in a GSDML file using the GSDML Generator. The generated file can then be integrated into the controller. The device configuration is thus fixed and does not have to be reprojected on the controller side.

For more information see chapter GSDML Generator.

The new rack view is illustrated below:



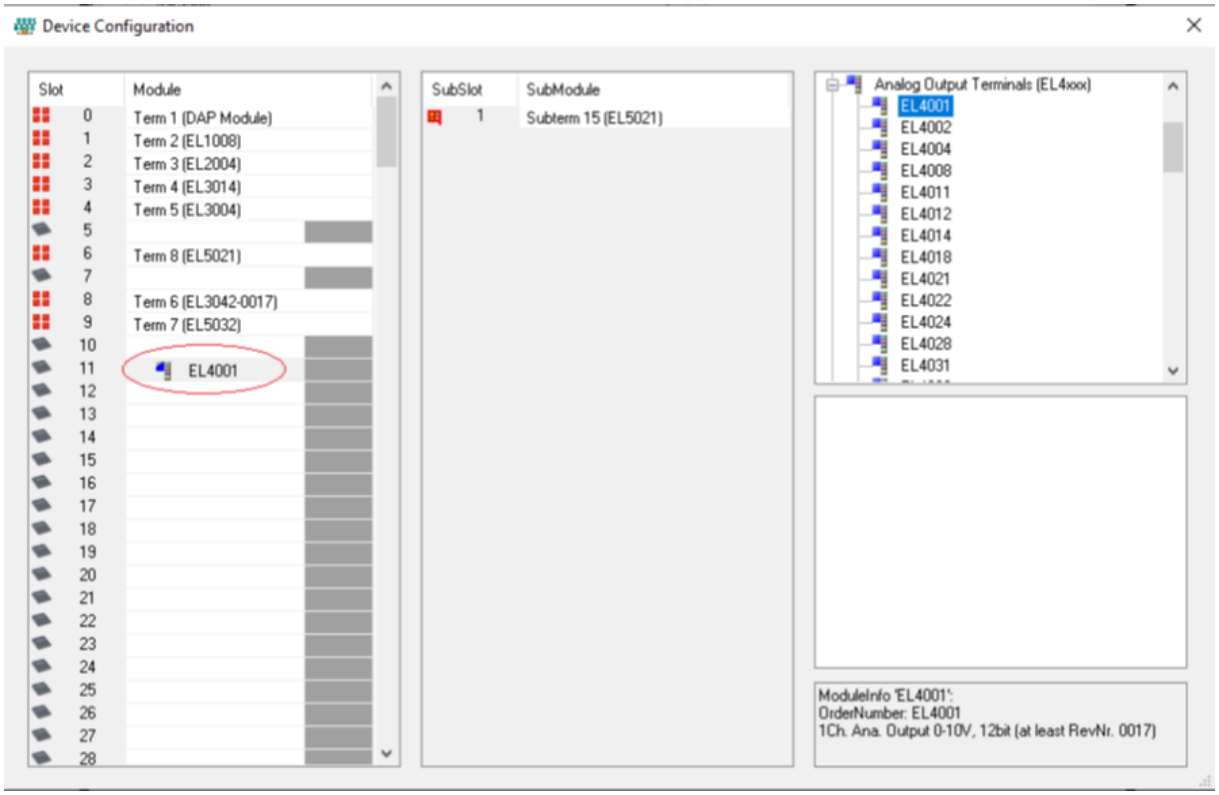

The list on the left shows the available device slots; here you can see which slots are occupied or free. Clicking on a slot updates the list in the center, showing the available subslots at the selected slot. If supported by the module, the submodules can then be populated here.



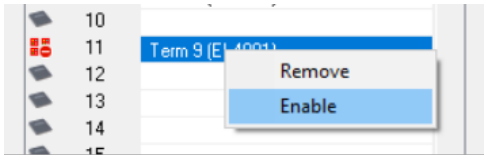
The list on the right shows the available modules and submodules. The list of submodules is always updated depending on the selected module.

The slots and subslots can be populated simply by double-clicking or via drag and drop.

- Double-clicking always triggers insertion at the next free and available slot from the cursor position.
- With drag and drop the available slots are selected, and a module can then be dragged to the required slot and released.



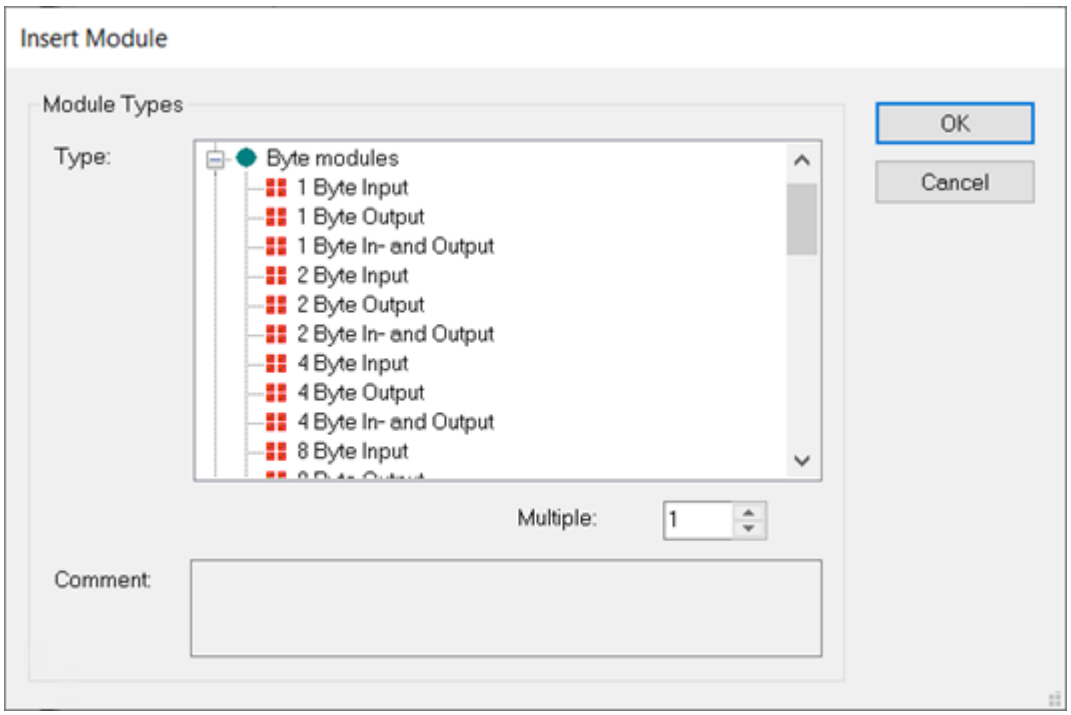

In the slot and subslot list, populated modules or submodules can be disabled via the context menu (right mouse click).



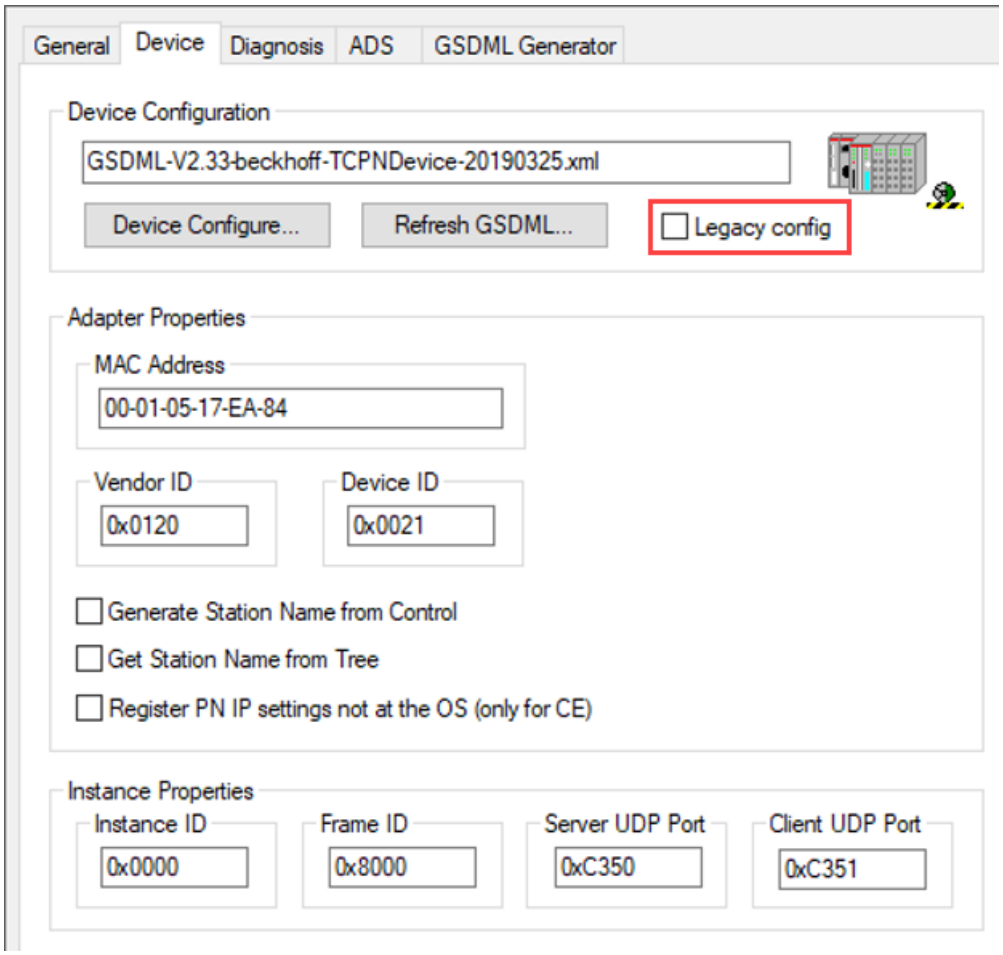

The icon indicates disabled status, which is also apparent in the module tree. Disabled objects are ignored during project planning, i.e. corresponding slots or subslots are considered empty.

### Classic tree view

It is possible to switch to classic tree view for module/submodule population, if required.




To do this, check **Legacy Config** on the **Device** tab of the attached box.

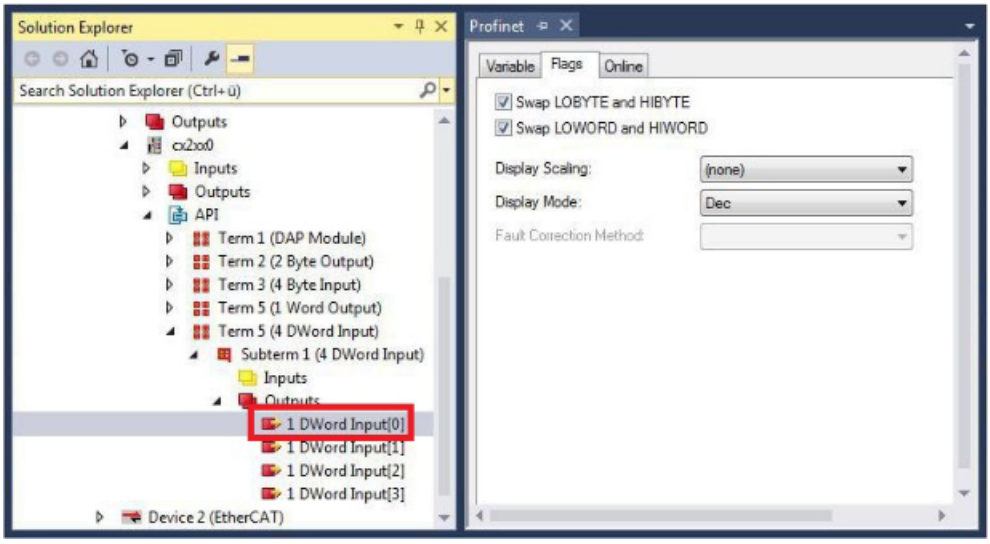



### 4.2.2.3 'Turning' process data

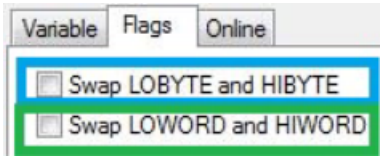
The process data are transferred in Intel format as standard. If the data is required in Motorola format, the data must be rotated accordingly. This step illustrates how to 'turn' the data in TwinCAT.

'Turn' the process data as follows:

1. Click the process data you want to rotate on the right in the tree view.
2. Click on the **Flags** tab




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

| Controller data | Data received by the device | | | |
|---|---|---|---|---|
| Original data | No option selected | Swap Byte (blue) | Swap Word (green) | Swap both (blue and green) |
| 0x01020304 | 0x01020304 | 0x02010403 | 0x03040102 | 0x04030201 |

# 4.3 PROFINET Features

## 4.3.1 Acyclic data

The ADS function blocks are used to send acyclic data. They access the PROFINET record data. So that acyclic data can be read or written, the PROFINET device must be in data exchange mode.

An *ADSReadWrite* is set.

### ADS settings

*AMSNetID*: The AMSNetID of the Profinet controller

*PORT*: Port number of the device (take this from the System Manager)

*Index GROUP*: `0x0000_F823`

*Index OFFSET*: `0x0000_0000`

DATA

```
typedef struct {
    WORD          RW;
        #define     PN_READ   0
        #define     PN_WRITE  1
    WORD          NrOfAR;
    DWORD         API;
    WORD          Slot;
    WORD          SubSlot;
    PNIO_RECORD   RecordData;
} PNIO_CONFIGRECORD
```

| nRW | nNr | nAPI | nSlot | SubSlot | nIndex | nLen | nTrans | nLenA |
|---|---|---|---|---|---|---|---|---|
| 2 bytes | 2 bytes | 4 bytes | 2 bytes | 2 bytes | 2 bytes | 2 bytes | 2 bytes | 2 bytes |

**Example:**

Send a read request for I&M function 0

| nRW | nNr | nAPI | nSlot | SubSlot | nIndex | nLen | nTrans | nLenA |
|---|---|---|---|---|---|---|---|---|
| 00 00 | 00 00 | 00 00 00 00 | 00 00 | 01 00 | F0 AF | 00 00 | 01 00 | 00 00 |

**NOTE**

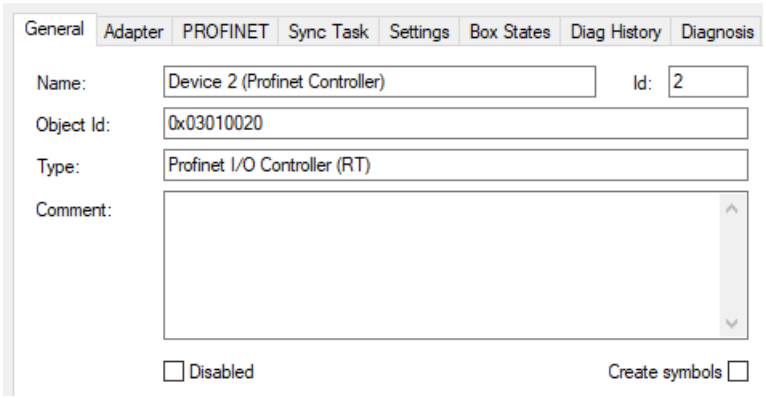
**Pay attention to the memory size**

Make sure that the receive data memory is large enough.

# 5 Settings and diagnostics

## 5.1 Settings on the PROFINET Controller protocol

### 5.1.1 General




**Name**

Identifier for the PROFINET device protocol object

**Id**

The device ID is set by the TwinCAT System Manager during configuration and cannot be configured by the user.

**Object Id**

Identification number of the PROFINET device protocol object in the TwinCAT object context.

**Type**

Shows the selected object type and its property.

**Comment**

Freely editable comment to describe the object used.

**Disabled**

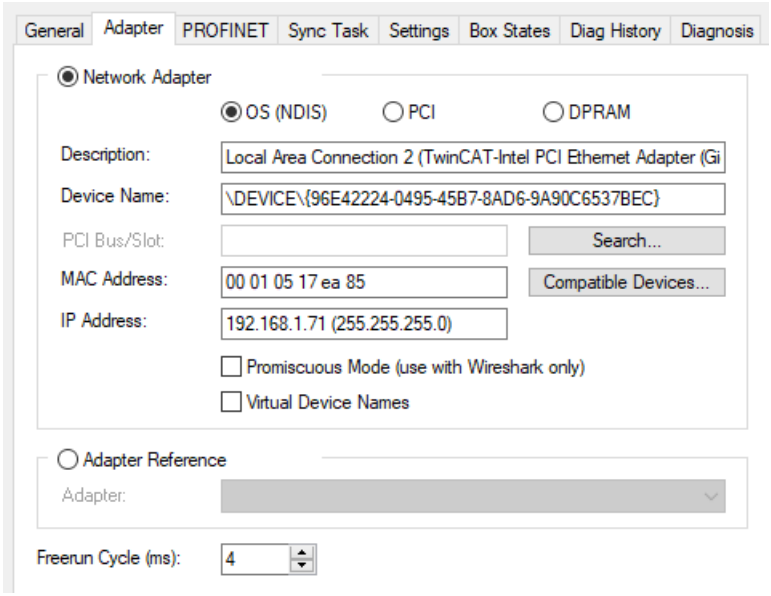
This option sets the PROFINET controller to inactive (transparent) for the current configuration. If this option is activated, the corresponding object is ignored in the IO configuration.

**Create symbols**

Creating variables as symbolic names.

### 5.1.2 Adapter

This dialog is used to specify and parameterize the network card to be used for communication with the PROFINET Controller.




**OS (NDIS)**

This option uses the operating system (OS) settings for installed network cards. The name of the network card is displayed under Description. Device Name contains the Device Manager path of the installed network card

**PCI**

This option controls the network card via the PCI bus address, which is specified in the PCI Bus/Slot field.

The PCIBus/Slot field is not enabled until the PCI option is selected

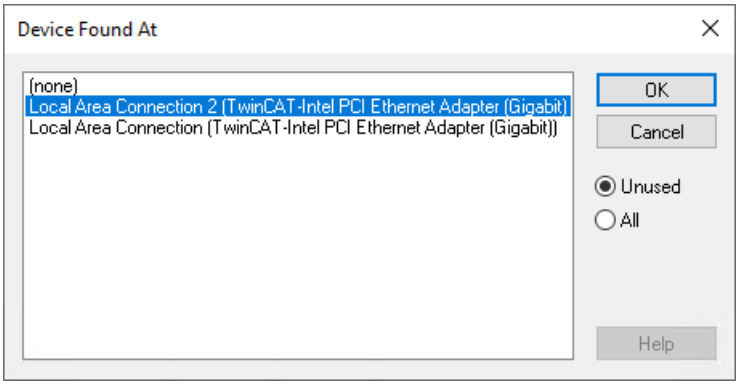
**DPRAM**

This option controls the network card via the DPRAM address, which is specified in the Address field.

The Address field is not enabled until the DPRAM option is selected.

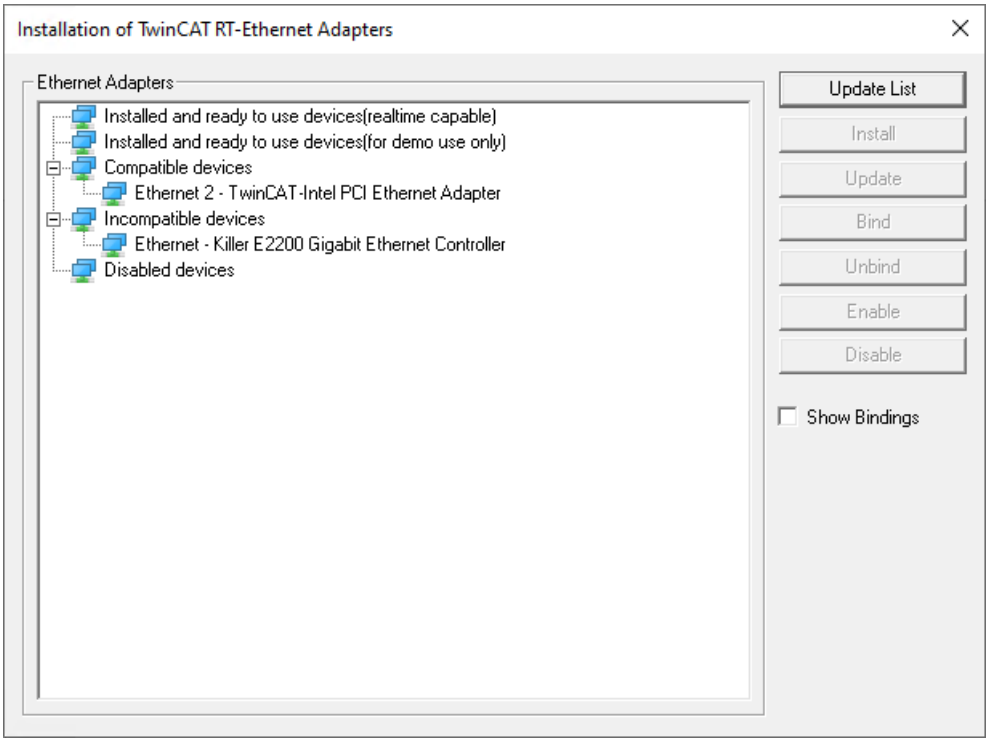
**Search...** button

This button opens a dialog in which all unused or all compatible devices (adapters) can be selected.




**Compatible Device…** button

This button opens the same dialog as "TWINCAT\ Show Real-time Ethernet Compatible Devices..." in the main menu. Use the dialog box to determine if compatible Ethernet adapters are available on the system.




**MAC Address**

MAC address of the Ethernet card (read-only).

**IP Address**

IP address of the card (read-only). The IP address is read from the operating system. It is unrelated to the PROFINET IP address that will be used later.

**Promiscuous Mode**

This is required in order to record Ethernet frames, and should normally be switched off.

**Virtual Device Names**

A virtual name is used for the network card.

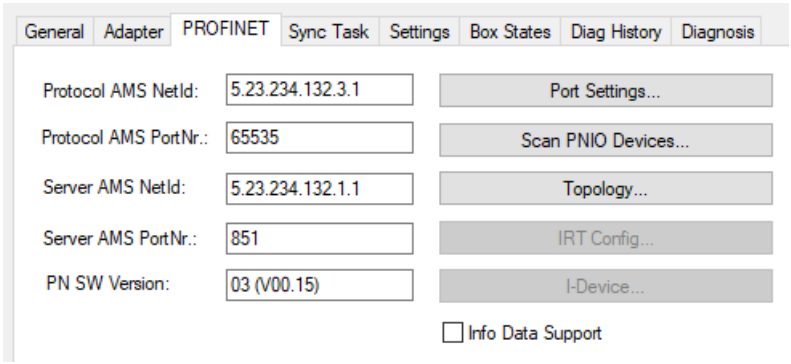
**Adapter Reference**

If the network adapter is referenced to another device, this option must be selected. This is used, for example, when using the "Multiple Protocol Handler".

**Free Cycle**

Cycle time in Config mode (no real-time). If TwinCAT is operated in FREERUN mode, care must be taken that the freerun cycle set is no longer than the PROFINET cycle!

## 5.1.3 PROFINET




**Protocol AMS NetId**

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. This is always fixed to 0xFFFF

**Server AMS NetId**

This is the NetID to which certain AMS messages (e.g. PN records within the index range 0x1000 - 0x1FFF) are forwarded by the PROFINET driver. Currently this is always the SystemNetId.

**AMS PortNo server**

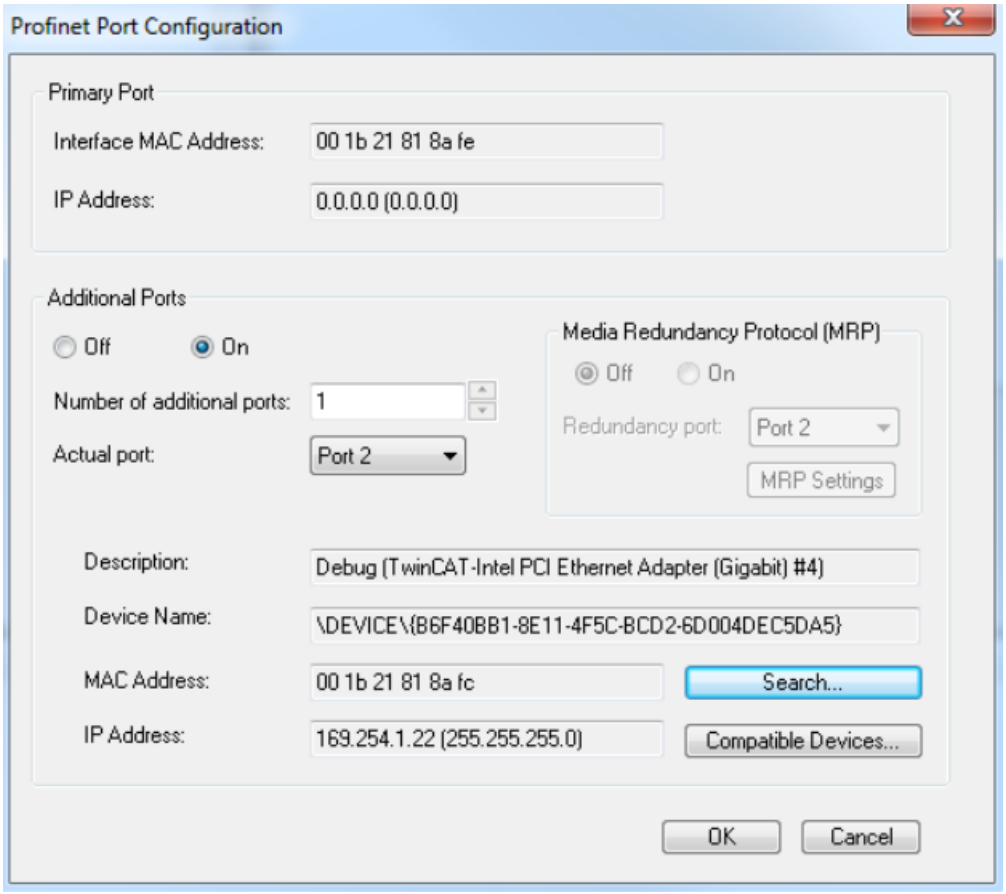
This is the PortNo to which certain AMS messages (e.g. PN records within the index range 0x1000 - 0x1FFF) are forwarded by the PROFINET driver. By default this is the PLC Port 802 of runtime system 1.

**PN SW version**

Firmware version of the device

**Port Settings**

This feature is currently only approved for the Real-Time Ethernet protocol (no EL663x). With this a second PROFINET port and an intelligent switch can thus be realized with a second network card (Intel chipset) . It is intended to repeat this feature x times; however, it is presently limited to one additional port.



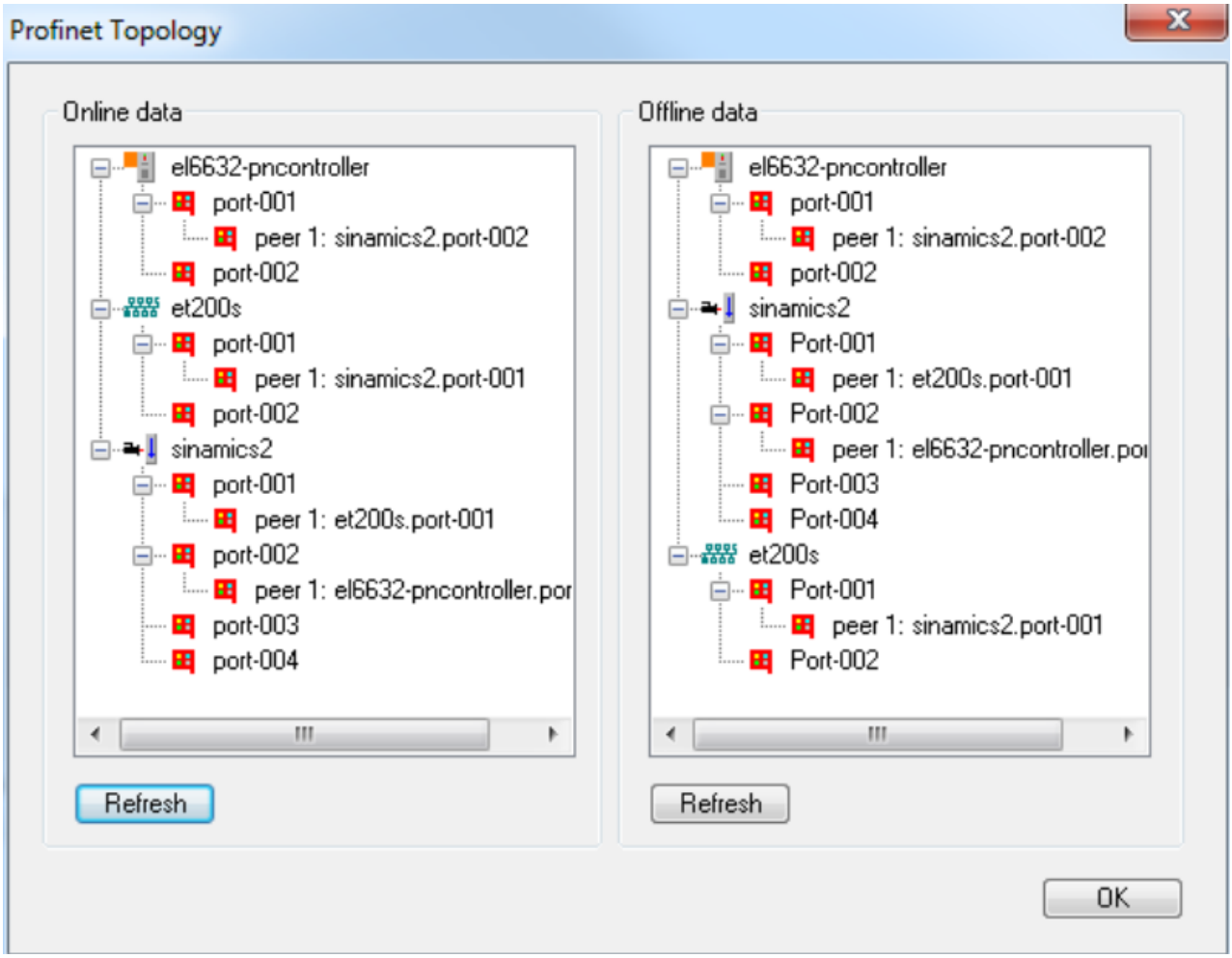

**Scan PNIO Devices**

Opens a search dialog for PROFINET devices; only available for the controller.

Appending via the Scan function [▶ 24]

**Topology**

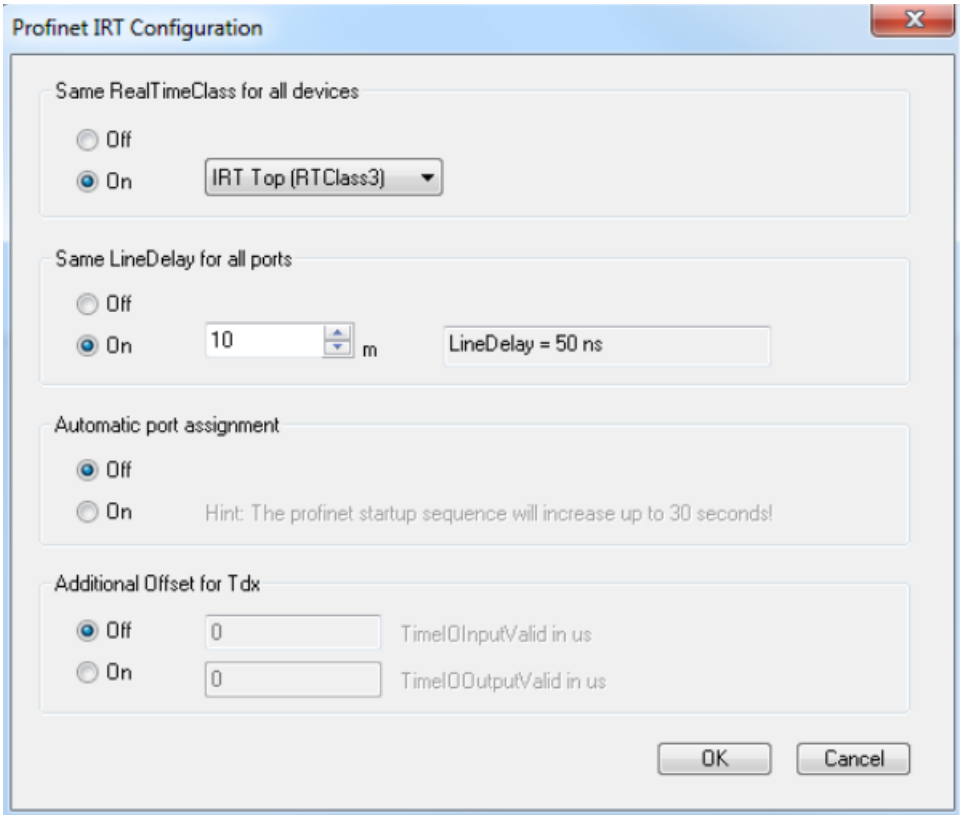
The offline topology can be compared with the online topology via this dialog.




It is quite possible for a device to have several partners on a port in the online view. This is the case, for example, if a switch is used in PROFINET that does not support LLDP (protocol for neighborhood ID). In offline view partners may have been assigned that do not exist in the project. This takes place if the reading of the port properties was activated during automatic scanning and appending. In this case the device has a ‘neighbor’ that is adopted into the project, but the associated device box is missing from the *.tsm file. When this project is enabled, the "neighbor" that is not present in the *.tsm file is ignored in the driver.

**IRT Config**

This menu is only enabled for an IRT-capable controller (currently only EL6632). A global setting can be made for all projected devices via this menu.




On the one hand the type of communication can be specified here. At present only RTClass1 (RT) and RTClass3 (IRT Top) are supported. In addition there is an option in this dialog to specify a general cable length (for IRT only). An approximate value or the max. cable length is sufficient here, because for the calculation of IRT communication this value tends to be lower (at 100 Mbaud and copper cable 5 ns/m). For optimization this feature can also be deactivated again later and the exact cable length can then be entered for each individual device (on the port submodules). Furthermore there is an option here to activate an 'automatic port assignment'. As a result of this the port connection set in the TwinCAT project is irrelevant. Before each restart of the PN communication the topology is read out and the IRT communication is calculated on the basis of this. The advantage of this is that possible cabling errors are minimized. In addition to that the ports can be simply replugged without having to change and reload the TwinCAT project. Only a restart of the PN communication is required (e.g. switch terminal to PREOP or disconnect the cable). As a result of this the start-up of the PROFINET communication can take up to 30 seconds longer. The reason for this is the TTL (TimeToLive) factor in the LLDP MIB. These are set by default to 20 seconds, i.e. only after this time can it be guaranteed that the port connection read is also the current one. Also, an additional offset for all Ti / To values can be specified in this menu.

**I-Device**

Opens a dialog for simultaneous parameterization of a controller AND device interface. Only available at CCAT.

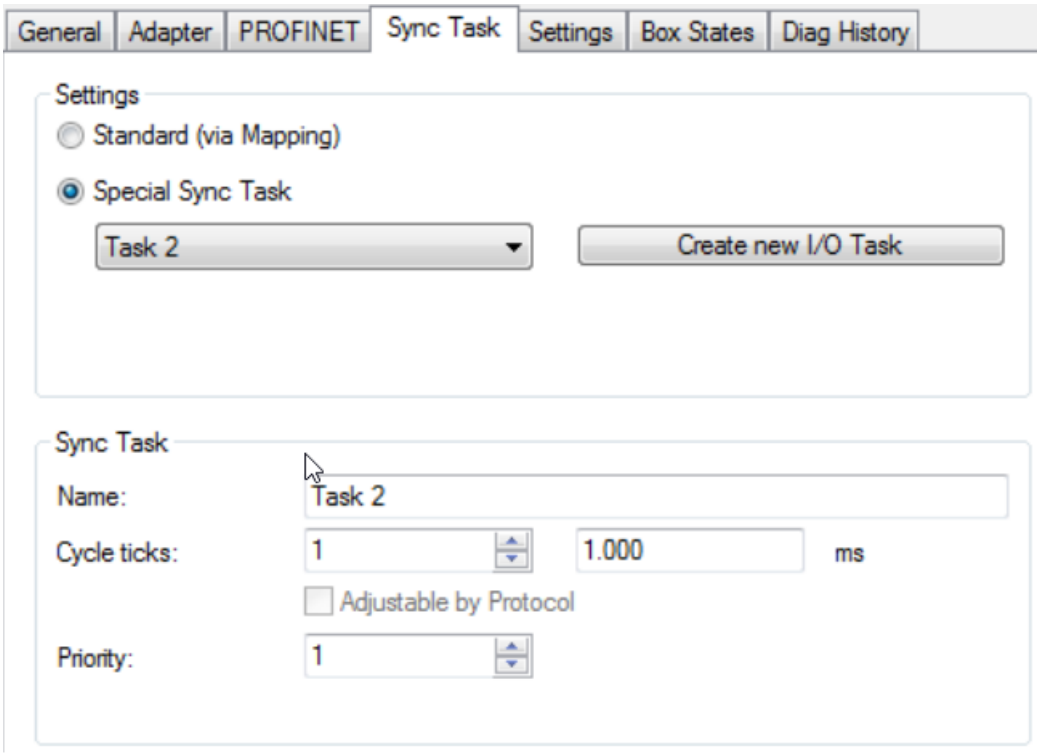
Creating an I-Device [▶ 20]

**Info Data Support**

If this option is activated, the AMSNETID is also available in the TwinCAT tree and can then be linked accordingly.

## 5.1.4 Sync Task

The PROFINET controller protocol must always be linked with a task. The protocol is also processed with the set task time. Theoretically the controller can also be jointly processed, for example, 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.



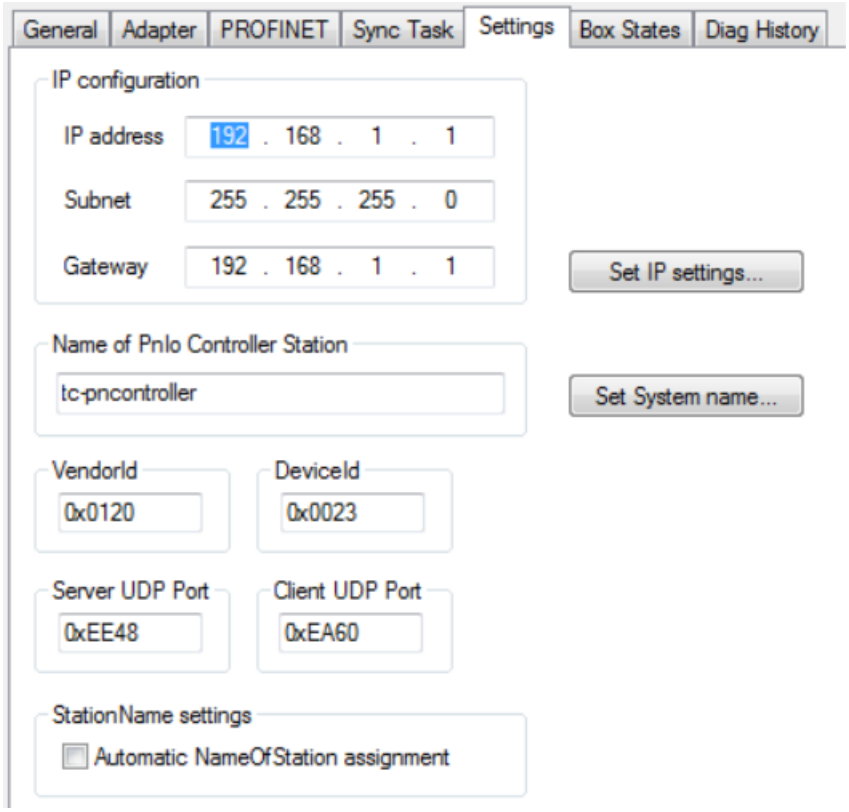

Make sure that the task cycle is in a PROFINET cycle. For PROFINET the basic cycle is 31.25 µs. The basic cycle is obtained by multiplying this cycle by the SendClockFactor (SCF). The SendClockFactor is usually set to 32 for RTClass1. For the Beckhoff PROFINET controller this is also the minimum PN cycle for RTClass1. The smallest resulting cycle time is 1 ms. Further reductions take place using a ReductionRatioFactor. It always corresponds to a multiple of the minimum PN cycle. For RTClass1, the smallest cycle must always be doubled **(permissible cycle times (for RTC1) with an SCF of 32 are 1, 2, 4, 8, ... 512 ms)**. In order to realize faster cycle times for RTClass3, the SCF must be reduced. This is presently at least 16 for a Beckhoff IRT controller (EL6632), which corresponds in turn to a basic cycle of 500 µs. When reducing the PROFINET cycle in this way, note that the time of the triggering task must also be adjusted accordingly.

**Using EL663x**

When using the EtherCAT PROFINET gateway terminal, please note that the process data always arrive one cycle late in the PLC, since one cycle is required to transfer the process data from PROFINET to EtherCAT

## 5.1.5 Settings

Specific settings that directly affect the controller can be made via the **Settings** tab.



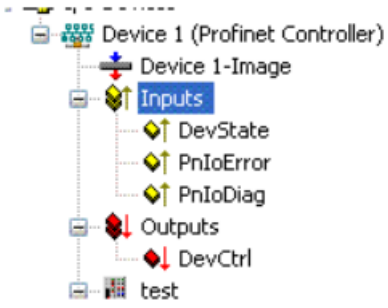

An IP setting can take place here. The selection of the address range need not correspond to the network card settings. The PROFINET communication spreads its own net, which can be selected here. The IP Settings shown in the image above are the default settings. If no changes are made here, the controller will use these settings. The same applies to the controller name (system name). To change both settings you will need to select the appropriate button. A check is made to ascertain that the input is correct (e.g. the format of the controller name must correspond to the PN spec.). This data is transferred permanently. When changing the subnet or gateway, the settings are also applied to any projected devices.
It is also possible to change these settings via a supervisor tool. In addition the VendorID and DeviceID of the controller can be read out in this dialog. The server and client UDP port employed can also be set here. The default settings should be adequate in most cases. Furthermore there is a possibility in this dialog to enable an automatic PROFINET start-up following a device exchange (including devices without removable media). The target topology must be specified to ensure correct function. On the basis of this information the controller can query the alias names of the individual devices. Every device that supports alias names generates such a name for each of its ports. This is composed of the neighborhood IDs (PortId.ChassisId). If this name is queried, the ‘new’ device answers. If VendorId and DeviceId are correct the device is named with the actual name and a normal PROFINET start-up can subsequently take place. With this mechanism a complete PROFINET system could also start up without having named an individual device beforehand.

When using the TF627x supplement, note that the IP addresses of the operating system and PROFINET are different.

## 5.2 PROFINET controller protocol diagnostics

### 5.2.1 Box States

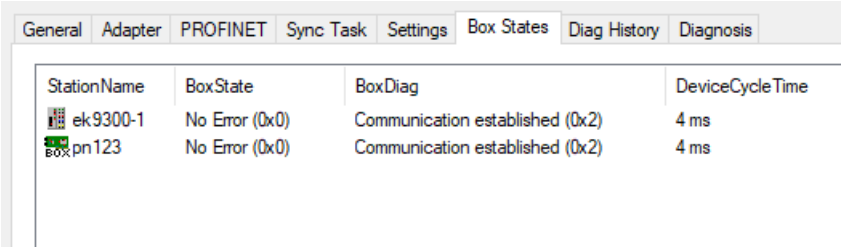
Directly below the PROFINET controller protocol there is a collective PROFINET error and a collective PROFINET status. Both show the number of devices for which a problem has occurred or for which a diagnosis is available. The error indicates possible problems with the connection establishment or reasons for a termination. The diagnosis provides status information about an existing connection.




PnIoError - number of PROFINET IO devices that have an error

PnIoDiag - number of PROFINET IO devices where a diagnosis is available

It is possible to check at a glance which device or box has a problem in the protocol under Box States.




Presently the following error messages are displayed under the “PnIoState”.

| Number | Text | Description | Remedy/Cause |
|---|---|---|---|
| 0 | No Error | No error | No error |
| 1 | PROFINET Device state machine is in boot mode | PROFINET DeviceStateMachine is still in the start-up phase | Not an error, wait |
| 2 | Device not found | Device does not reply to the Identify Request | Check the connection; device switched on and connected; was the device named correctly? |
| 3 | The station name 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 | IP conflict 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 alias name 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 assignment is not 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). This means that no automatic naming and thus no device start-up is possible. |
| 31 | Only for EtherCAT gateways: WC-State of cyclic EtherCAT frame is 1 | For EL6631 only: EtherCATWC state is set to 1 | Check the mode on the EtherCAT master & slave (OP?). |

As opposed to the state, more than one status can be displayed in the “BoxPnIoDiag”, i.e. the whole thing is bit-coded and up to 16 pieces of information can be displayed. The following statuses are currently displayed.

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 Problemindicator is set

0x1XXX = At least one OutputCR is invalid

0x2XXX = At least one OutputCR Provider is in stop

0x4XXX = At least one OutputCR Problemindicator 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. This applies to the input and output CRs (currently only one CR is possible, in future several). A PROFINET alarm is also displayed in "PnIoDiag"

**Readout via ADS**

The Box Status can be read out via an ADS Read.

ADS Read:

NetId = AMSNETID of the PROFINET controller

Port = BoxPort (0x1000 + BoxId)

Indexgroup = 0xF829

IndexOffset = 0

Length = sizeof(TPnIoDeviceDiagData);

where:typedef struct{

WORD pnioState;

WORD pnioDiag;

WORD NrOfInputCRs;

WORD NrOfOutputCRs;

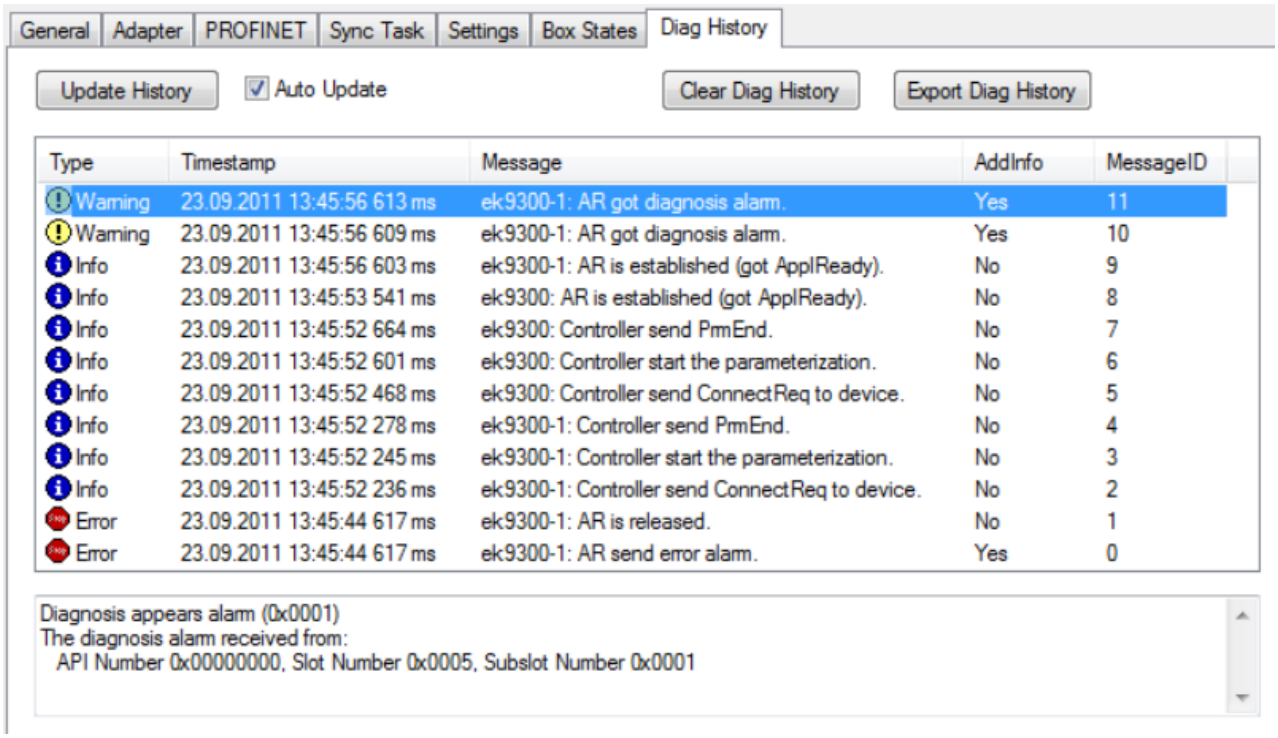
WORD reserved[8];} TPnIoDeviceDiagData, *PTPnIoDeviceDiagData;

**Readout via CoE (for EL663x)**

The Box Status can also be read out via CoE for the EL663x. The index 0xAyy0 (where yy is the Adaptor / Device number) and the subindex 0x001 must be taken for this.

## 5.2.2 Diagnostics history

Logged diagnosis messages from the controller protocol can be read out on the **Diag History** tab. The diagnosis buffer operates as a ring buffer with a current maximum size of 1000 entries.




The possible errors are grouped into three types:

**Info:** e.g. information on the connection establishment

**Warning**: e.g. PROFINET diagnosis alarms

**Error**: e.g. loss of connection

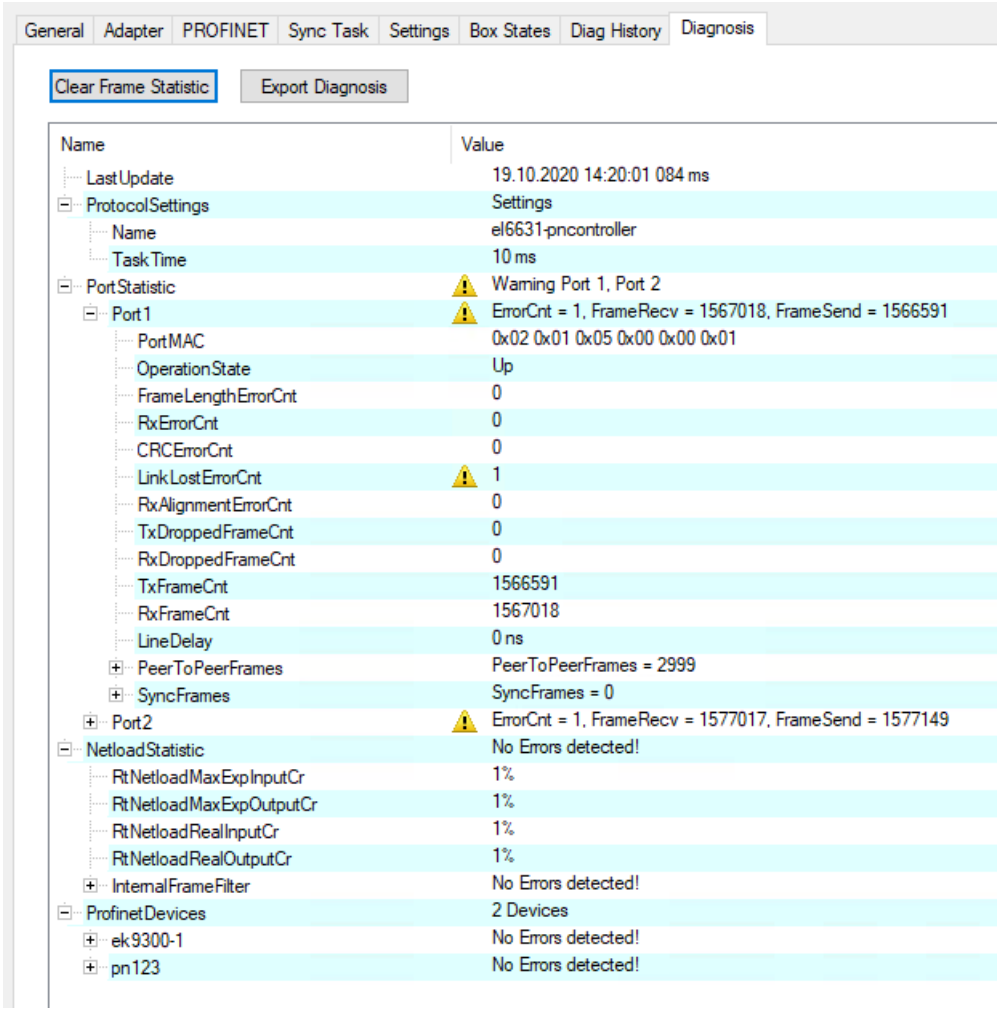
“AddInfo” indicates whether additional information about the event is available. If this is marked by “Yes”, the additional information can be fetched and displayed by clicking on the respective message. In the case of a diagnosis alarm ("Diagnosis appears"), the precise diagnosis information can be fetched at the corresponding level (device, API or module).

The complete diagnosis buffer is cleared by pressing the **Clear Diag History** button.

The displayed messages can be saved in a .TXT file by pressing the **Export Diag History** button.

## 5.2.3 Diagnosis

This tab contains the frame statistics; this list contains various information and statistics.

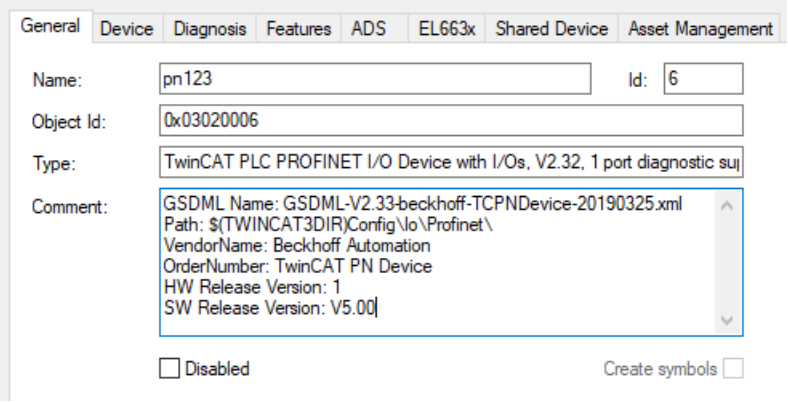



These can be reset or exported using the corresponding buttons; they are divided into the following sections.

1. Protocol settings: Include the name and cycle time of the PROFINET protocol.
2. Port statistics: Port-specific statistics and diagnostics.
3. Network load statistics: Percentage display of the expected bus load associated with the cyclic process data.
4. Also diagnostic counter of the internal network load filter to avoid possible frame bursts.
5. PROFINET devices: Diagnosis and statistics for the projected PROFINET devices.

# 5.3 Settings on the PROFINET device

## 5.3.1 General




**Name**

Identifier for the PROFINET device protocol object

**Id**

The device ID is set by the TwinCAT System Manager during configuration and cannot be configured by the user.

**Object Id**

Identification number of the PROFINET device protocol object in the TwinCAT object context

**Type**

Displays the selected/integrated GSDML and its property.

**Comment**

Displays further information on the selected/integrated GSDML, if available. Also freely editable commentary.

**Disabled**

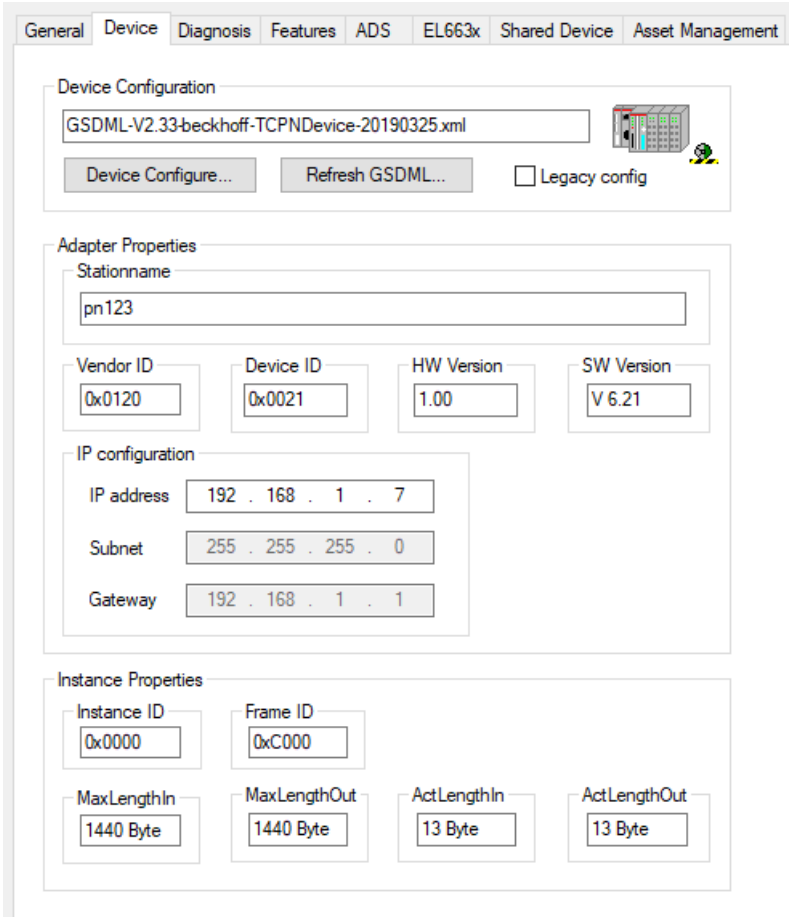
This option sets the PROFINET device to inactive (transparent) for the current configuration. If this option is activated, the corresponding object is ignored in the IO configuration.

**Create symbols**

Creating variables as symbolic names

## 5.3.2 Project planning of the PROFINET device

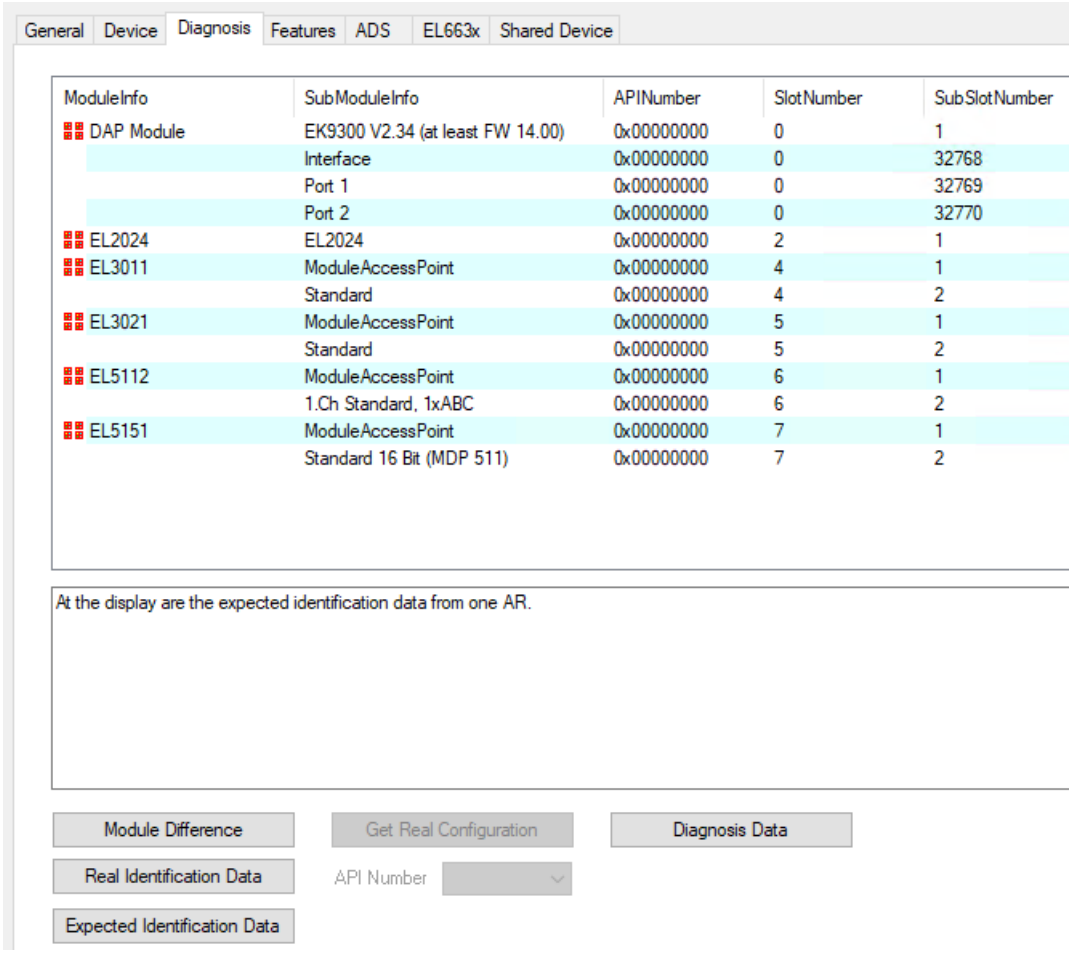
When establishing a PROFINET connection the controller always assigns an IP address to the device from its own address space (if the device does not yet have one or if it has a different one). In TwinCAT the next higher address is taken for a device by default (starting from the controller adaptor class); the subnet and gateway are the same as those of the controller. Before the actual assignment of the IP address to the device by the controller, an ARP is used to test for a possible address conflict or to check whether the device already has this IP address. If there is a conflict, e.g. that the IP address is already assigned in the network, the IO driver determines this and outputs a corresponding message in the logger window. If there is no reply to the ARP, this means that no device (the projected device included) is using this IP configuration, which in turn results in the controller assigning the IP settings to the device via a DCP_SET. Setting is skipped if it is determined via the ARP that the device sought already has the projected IP address. If the IP addresses of the PROFINET device and the operating system are identical, this can lead to unexpected behavior such as errors in the structure of the AR.



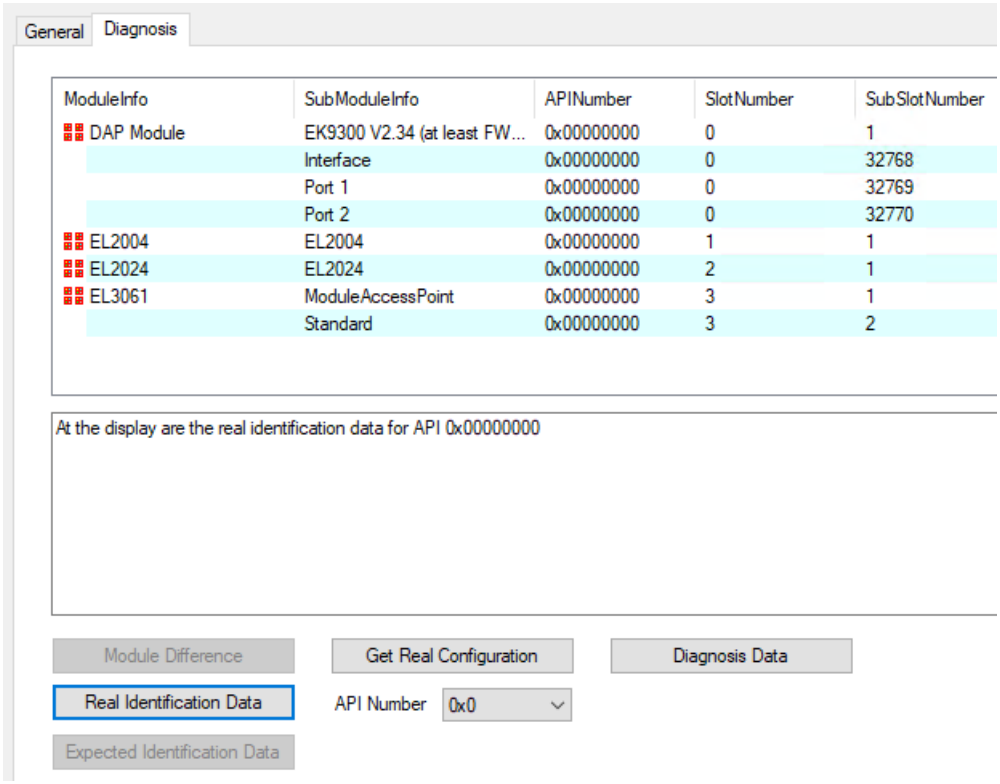

In this window you can also change the **InstanceID** and the **FrameID**. However, the default settings are adequate for most applications. The Instance ID is incorporated into the formation of the UUID object. A change should therefore be made only in exceptional cases. When changing the FrameID, the RTClass used must be taken into account (e.g. for RTClass1 unicast0xC000 - 0xFAFF). If the device is on an IRT controller and all devices have been switched automatically to RTClass3, the FrameID is managed automatically and there is no input option (marked by "Fast Config"). The current process data length can also be checked in this menu. **MaxLength** indicates which process data size is supported by the corresponding device, **ActLength** indicates the current process data length (including IOPS and IOCS). The corresponding error message appears if the maximum lengths are exceeded on appending further modules/ submodules.

## 5.3.3 Comparison of nominal and actual population

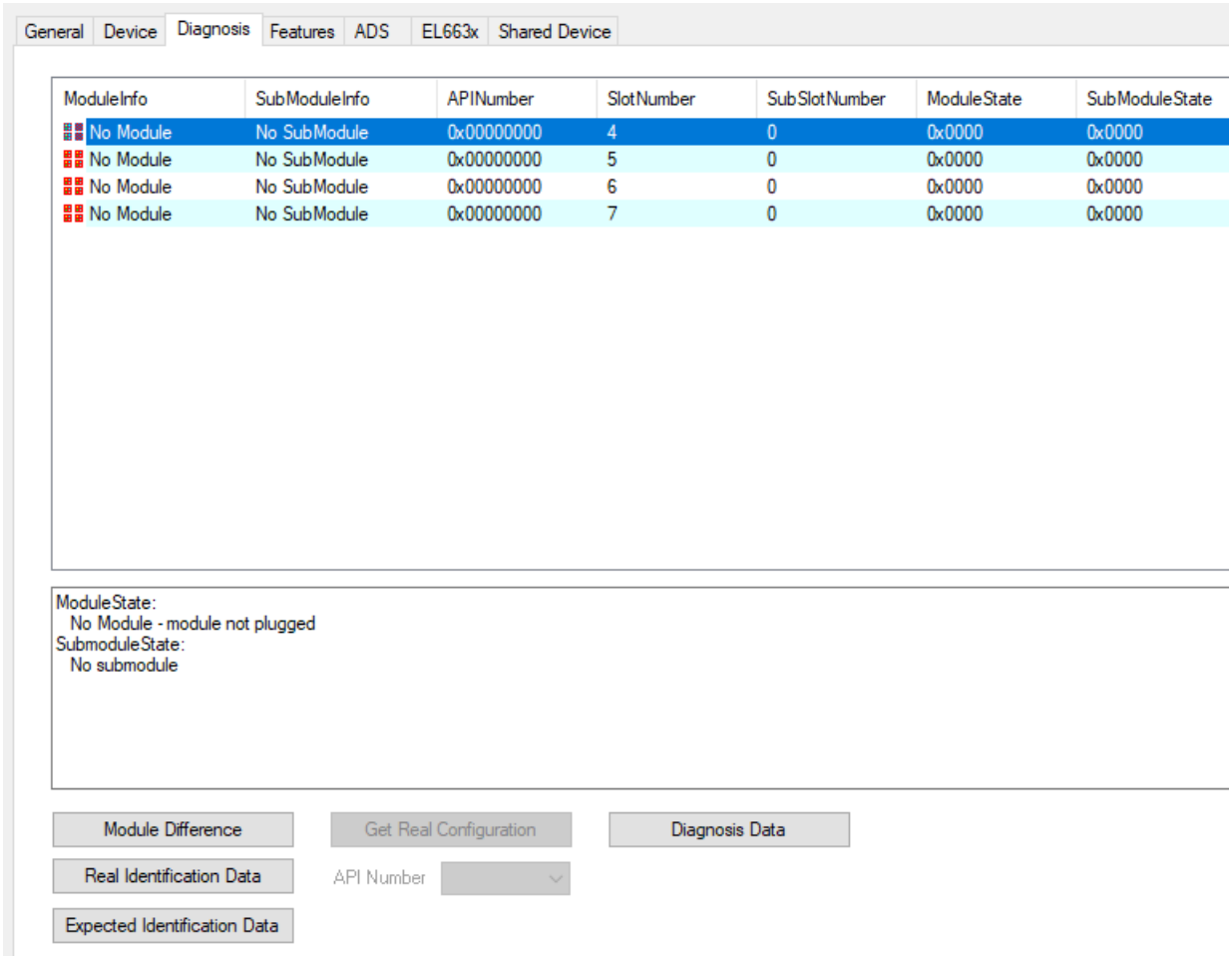
If a connection exists, the project planning can be checked on the **Diagnosis** tab. **Real Identification Data** indicates the existing modules within an AR at this level, **Expected Identification Data** indicates the expected modules (i.e. those projected in the controller). **Module Difference** shows the differences detected by the device during the target/actual comparison.



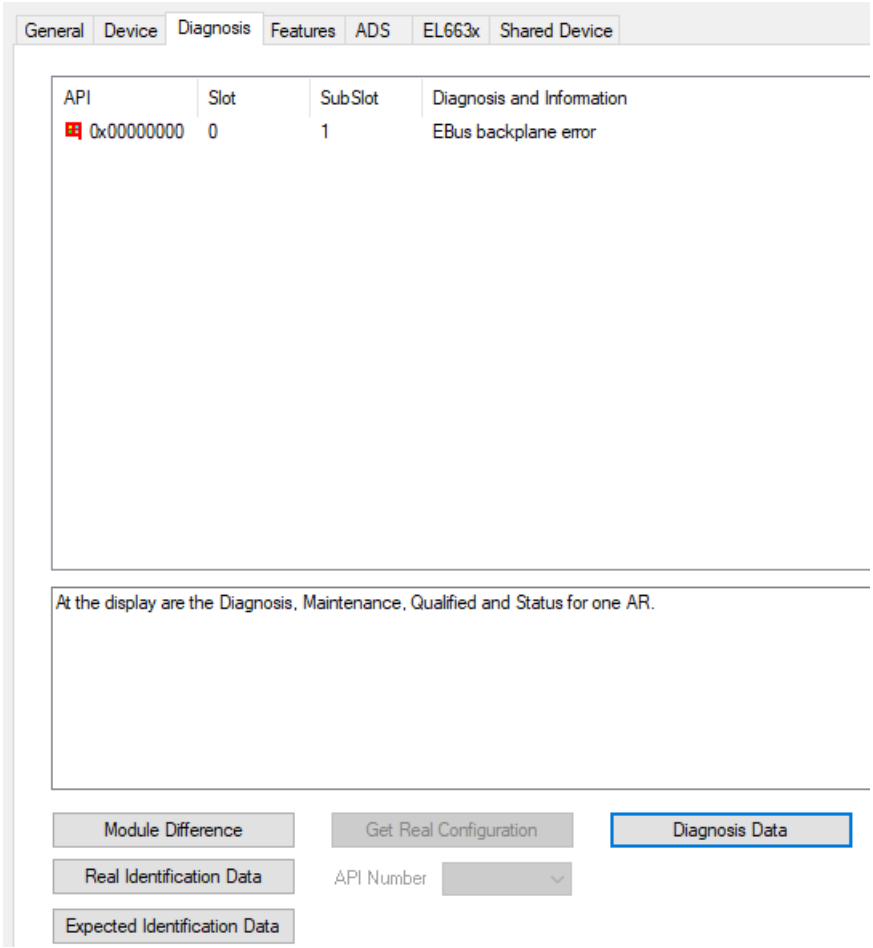

On the **Diagnosis** tab within the API the corresponding API can be selected from which information is to be obtained. If, for example, the PROFINET device is a drive, then this usually supports the Profidrive profile, which is identified in turn via API0x3A00. If the **Real Identification Data** is to be read from this API, for example, then this access takes place via the Profidrive profile.



The **Get Real Configuration** button becomes active within an API (except for drives). Here you can transfer the read-in data set into the current project. Note that modules that have already been created will be overwritten when doing this. This means that the links are lost, even in the case of previously correctly created modules. When displaying the module differences, additional information is displayed by marking the message.



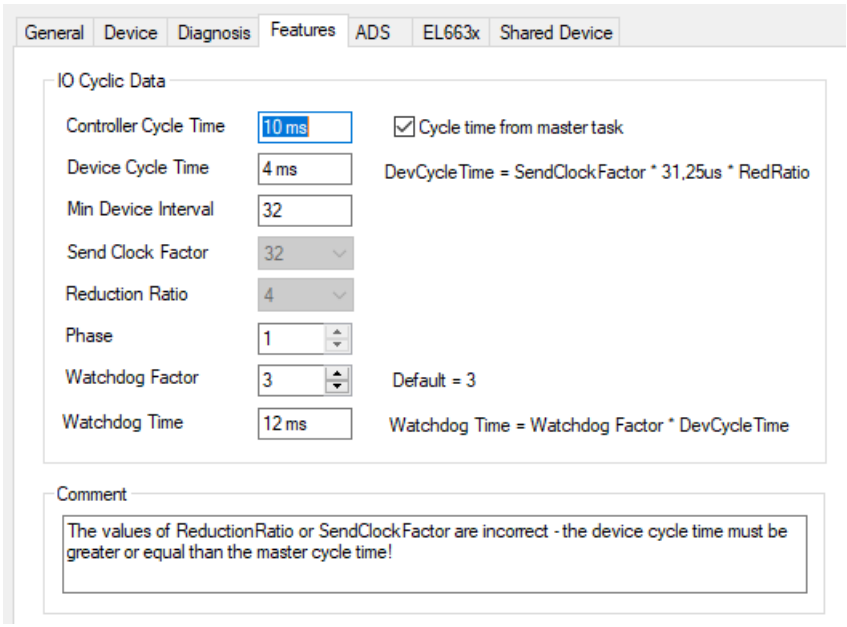

The available diagnosis can be read out by pressing the **Diagnosis Data** button. At device level all available diagnosis data for the existing AR is read out here.




Only two diagnosis parameters are displayed in the list, others are marked with "...". If the individual message is clicked, all available diagnosis information is displayed in the window below.

## 5.3.4 Features

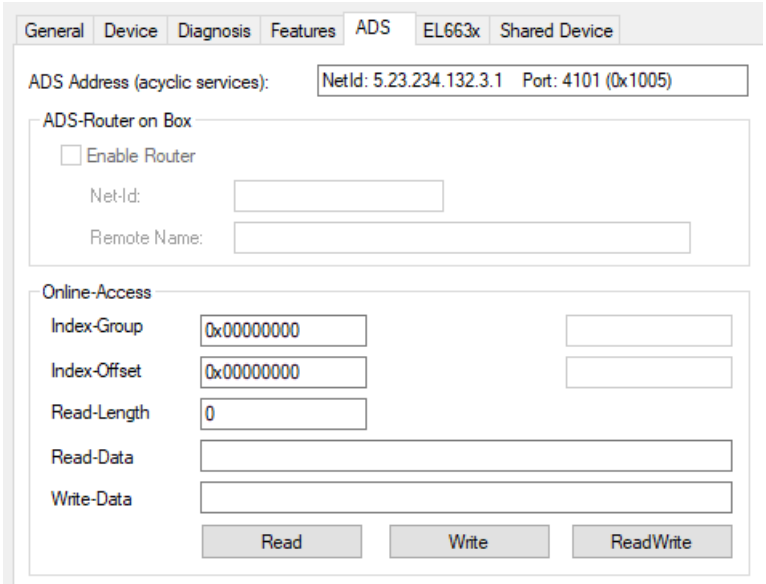
Various settings for the cycle time can be made on the **Features** tab. The cycle time of the controller must always correspond to a power of two for RTCLass1, starting at 1 ms (1, 2, 4, 8...). If an incorrect base time has been selected, this is indicated by a corresponding message. For RTClass3 the 1 ms base time can be divided again and again by two (down to min. 31.25 µs). The device cycle time can be changed via the exponents. The minimum is always the Controller Cycle Time, unless a larger minimum cycle time than that of the controller is defined in the GSDML. The maximum for RTClass1 is 512 ms. The **SendClockFactor** is fixed here as a time base to the value 32 (31.25 µs * 32 = 1 ms). The **Reduction Ratio** also refers to this, i.e. a factor of 4 means a cycle time of 4 ms. The transmission point can be shifted again within a cycle via the phase; i.e. where RR = 4 the phase can be 1 - 4. However, this value is only of importance in the case of a synchronized transmission.




The PROFINET **Watchdog Factor** can also be adjusted here. Each device monitors the input of the cyclic data based on this factor. If the factor is set to the default value (3) this means that, with an RR of 4, three cycles require 12 ms. Hence, a device reacts after 12 ms to missing telegrams (e.g. with an alarm and/or disconnection of the AR). The limits and values are recalculated each time when adjusting the individual factors.

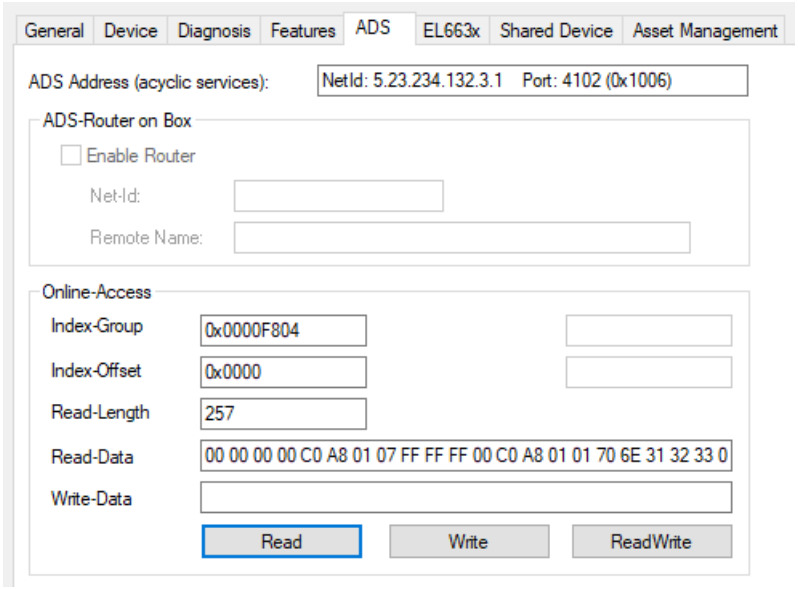
## 5.3.5 ADS

ADS messages can be sent directly from here. The NetId and the port are taken directly from the corresponding adapter.



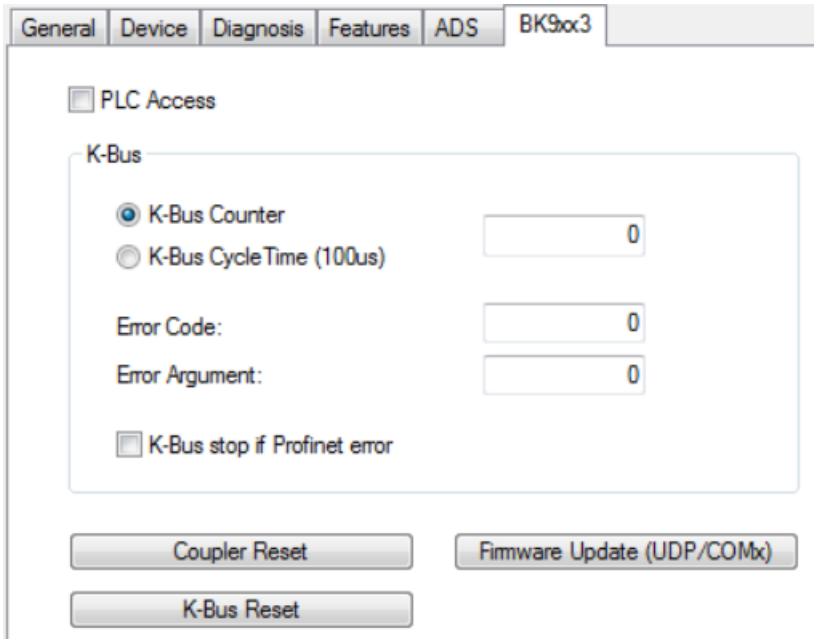

A large number of PROFINET functions can be triggered directly via the correct settings for index group and index offset. This includes, for example, the setting of alarms or record data.

Example: Read out the PROFINET name and the IP settings




## 5.3.6 BK9xx3

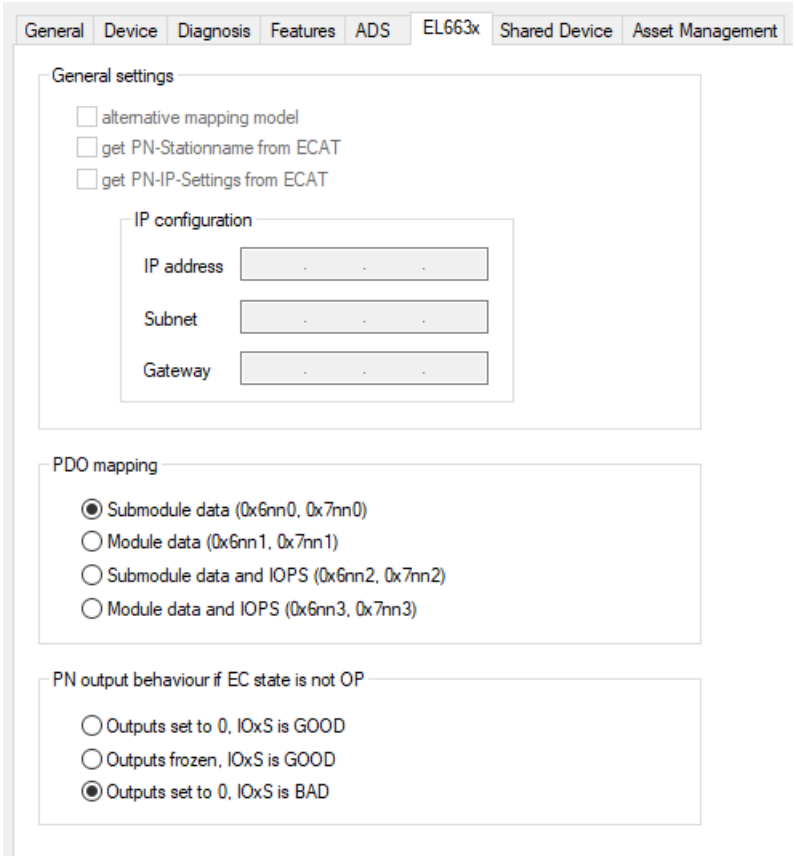
In the case of the Beckhoff K-bus Coupler (at present BK9103 or BK9053) that is not connected to an EL663x, an additional menu appears here.




This allows easy access to the cyclic process data in the DAP of the Bus Coupler. In addition, a firmware update from the System Manager to the Bus Coupler can be carried out via this menu. If the update takes place by IP it is important to ensure that the IP address is obtained via the DIP switches. If this is not the case the connection breaks off during the update, since the memory area of the IP settings is also formatted and rewritten.

## 5.3.7 EL663x

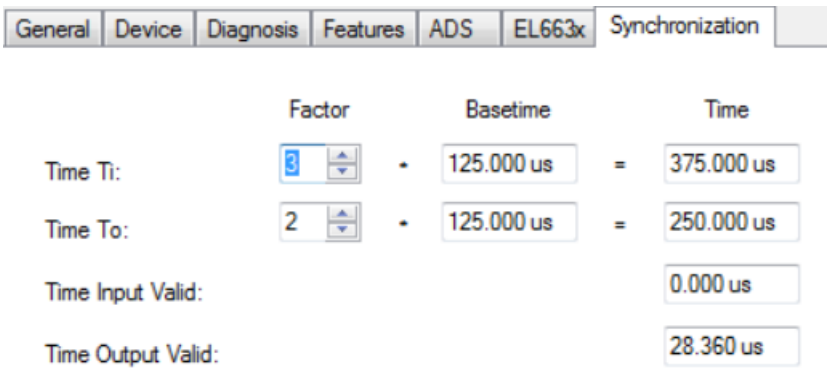
If the controller protocol is operated via an EL663x, then an additional menu appears on the devices.




Currently only the PDO mapping can be selected for the controller. This sets the form in which the PROFINET process data are mapped to the EtherCAT-side PDOs.

## 5.3.8 IRT Controller

If the device is operated with an IRT-capable controller, an additional **Synchronization** menu appears.

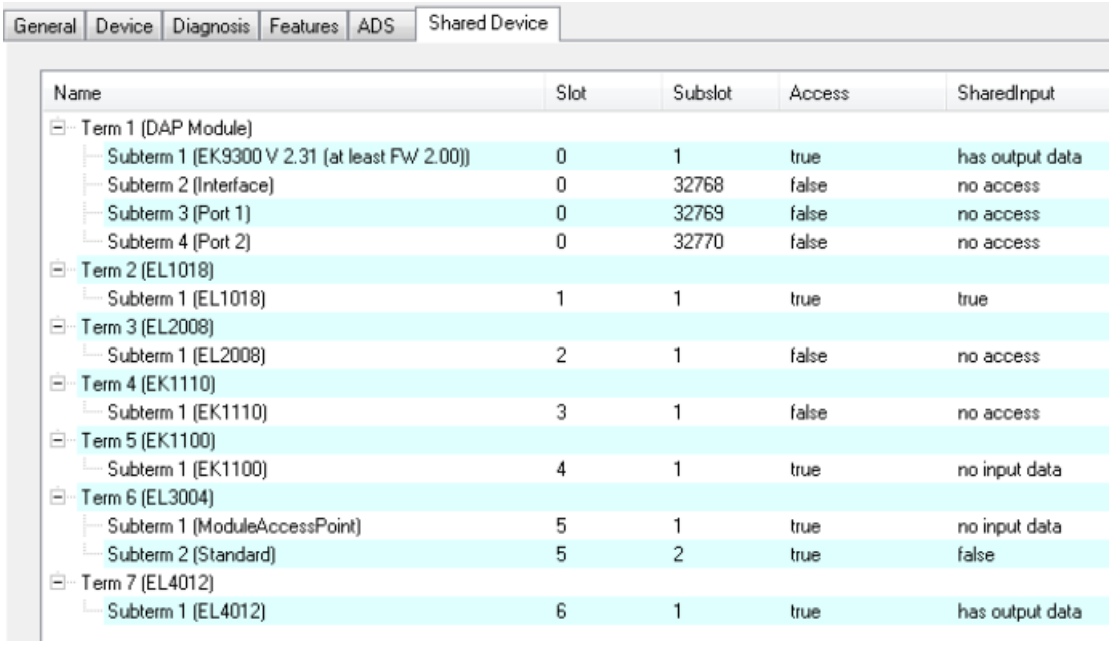



Here it is possible to specify the Ti and To factors (**Time Ti, Time To**) for IRT-capable devices. This means the time during which the data in the device are valid within a cycle, or should be set to valid. The prerequisite is that this feature is also supported. The GSMDL supplies the information about this. There is always a basic cycle here (**base time**). A statement about the minimum possible time comes via the GSDML on the basis of a minimum factor. The upper limit of the factor is limited by the cycle time employed. The shortest possible time in which the data could be valid over PROFINET (always in reference to the cycle) is displayed via the **Time Input Valid** or **Time Output Valid** parameter.

## 5.3.9 Shared Device

The Shared Device feature is available from TwinCAT 2 Build 22.50 or TwinCAT 3 Build 4019.

The dialog appears if the device supports **Shared Device**. The information for this comes from the GSDML.




| Name | Slot | Subslot | Access | SharedInput |
|---|---|---|---|---|
| Term 1 (DAP Module) | | | | |
| Subterm 1 (EK9300 V 2.31 (at least FW 2.00)) | 0 | 1 | true | has output data |
| Subterm 2 (Interface) | 0 | 32768 | false | no access |
| Subterm 3 (Port 1) | 0 | 32769 | false | no access |
| Subterm 4 (Port 2) | 0 | 32770 | false | no access |
| Term 2 (EL1018) | | | | |
| Subterm 1 (EL1018) | 1 | 1 | true | true |
| Term 3 (EL2008) | | | | |
| Subterm 1 (EL2008) | 2 | 1 | false | no access |
| Term 4 (EK1110) | | | | |
| Subterm 1 (EK1110) | 3 | 1 | false | no access |
| Term 5 (EK1100) | | | | |
| Subterm 1 (EK1100) | 4 | 1 | true | no input data |
| Term 6 (EL3004) | | | | |
| Subterm 1 (ModuleAccessPoint) | 5 | 1 | true | no input data |
| Subterm 2 (Standard) | 5 | 2 | true | false |
| Term 7 (EL4012) | | | | |
| Subterm 1 (EL4012) | 6 | 1 | true | has output data |

There is an option here to allow or forbid the controller to access the individual submodules. By default the controller may access all submodules; if SharedInput is supported it is switched off.

The text messages for SharedInput have the following meanings:

"not supported" - SharedInput is not supported by the device (info from the GSDML)

"has output data" - the submodule has outputs - activation of SharedInput not possible

"no input data" - the submodule has no inputs (and also no outputs)

"no access" - access is blocked

"true" or "false" - set value for SharedInput

The settings can be changed by double-clicking on the individual submodules. If the access to a port or interface submodule is changed, then it is changed for all ports or interfaces.

## 5.3.10 Module and submodules

### 5.3.10.1 Submodule level

PROFINET currently distinguishes between 4 types of submodule.

**Virtual submodules:** The virtual submodules are always permanently connected to a module. This means that when a module is inserted, the co-defined virtual submodules are also always inserted on the specified subslot. This kind of submodule is presently the commonest method.

**Real submodules:** Here there is a possibility to select the pluggable submodules from a submodule list and to append them to the module. The necessary information is procured from the GSDML. In TwinCAT a module can be selected from such a list with the right mouse button (provided this is supported by the device).

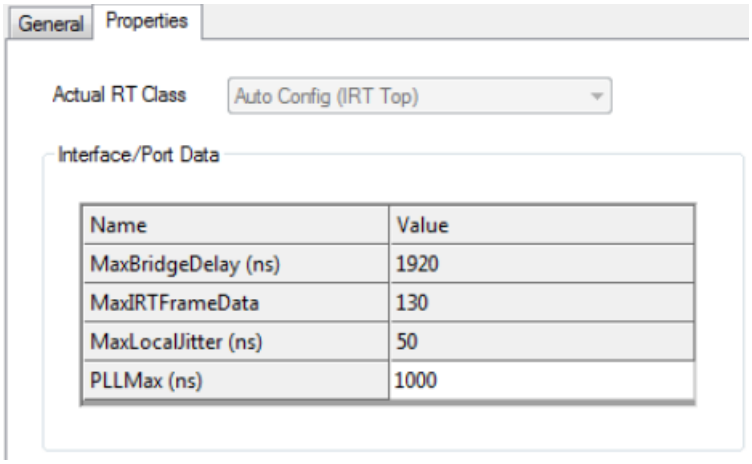
**Interface submodules:** Device-specific properties are defined in the interfaces submodules. These can be, for example, additionally supported protocols, timing properties, supported MIBs etc.

**Port submodules:** The physical properties of a network port are reproduced in such a submodule.

### 5.3.10.2 Interface submodule

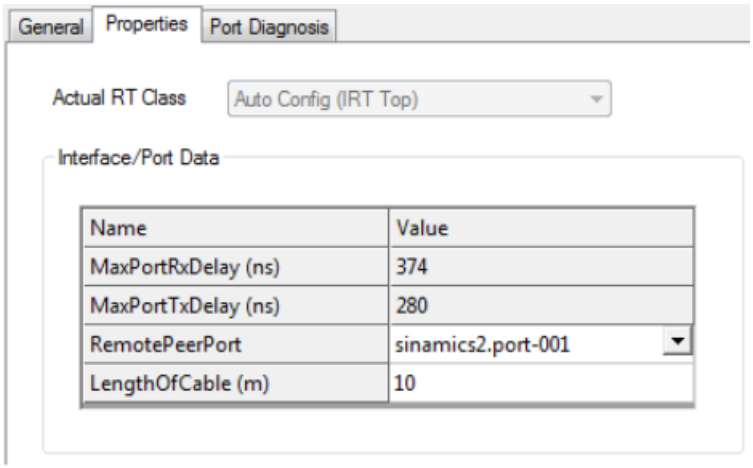
The type of communication can always be set on the interface submodel (at present RTClass1 or RTClass3). The only exception is in the case that a generally valid RTClass was set via the **Auto Config...** menu.

If communication takes place over RTClass3, then the PLL window can additionally be set at the interface.

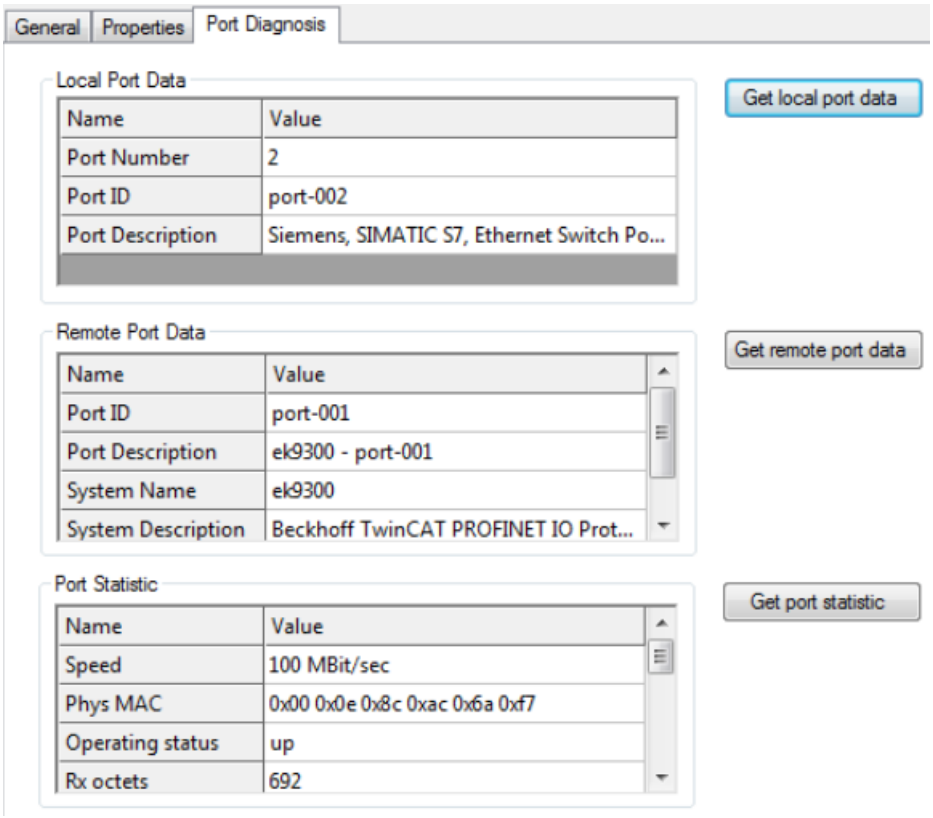



### 5.3.10.3 Port Submodule

In the **Properties** tab you can make port-specific settings. The menu of possible settings depends on the RTClass used.




Here you can read out some port properties.

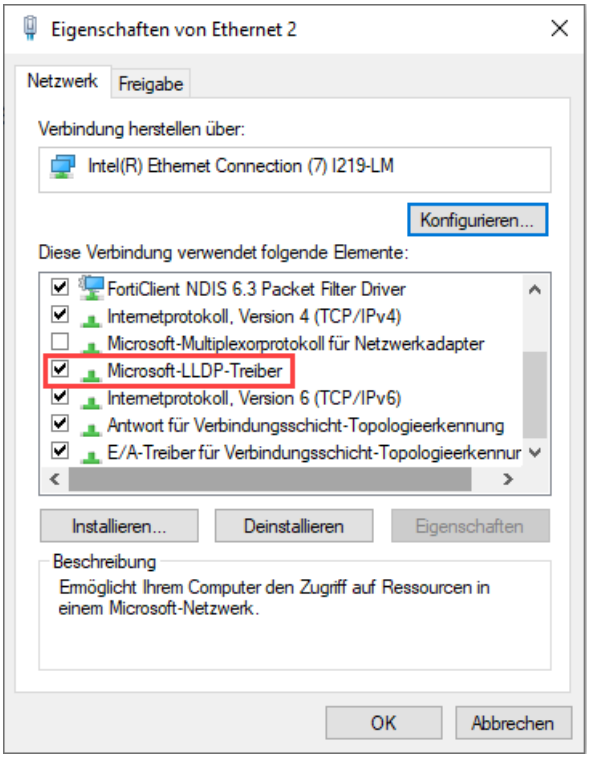



The information here is subdivided into local port information and remote port properties. In PROFINET the LLDP protocol (IEEE Std 802.1AB) is mandatory from conformance class A (CCA). The devices exchange neighborhood IDs via this protocol, so that each port is known to its neighbor. Furthermore, the Simple Network Management Protocol (SNMP) can be used as an aid at this point. On opening the **Port Diagnosis** tab, TwinCAT acts as a Network Management Station (NMS) and collects the required device information via SNMP. The previous images shows that the local port 1 of the BK9053 is linked to port 2 of the BK9103. For correct topology recognition it is important that only devices are present in the segment that also support the LLDP protocol (this is also applies to switches!).

### Undesirable behavior

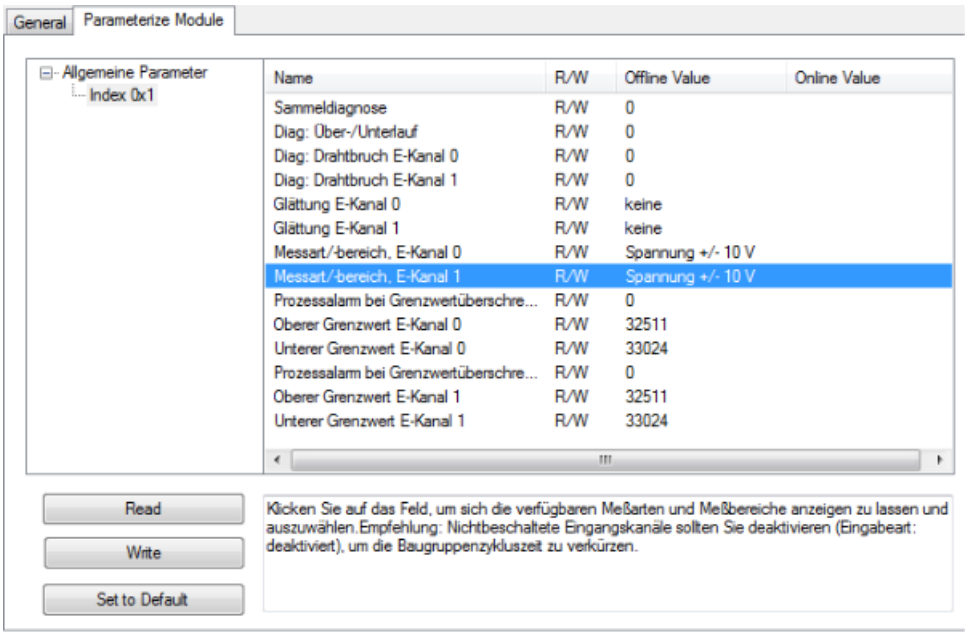
Unwanted behavior may occur in Windows 10. This is reflected in inconsistencies in topology detection.

Inconsistencies in topology detection.
Windows 10 provides an LLDP driver which is active by default. The PROFINET device also contains an LLDP driver. The system then sends two LLDP telegrams from one port to the remaining nodes on the bus. These remaining devices also expect two ports due to two LLDP telegrams, which do not exist.
To prevent this behavior, disable the Windows LLDP driver. To do this, uncheck **Microsoft-LLDP-Driver** (see red rectangle).




### 5.3.10.4 Real / virtual submodules

If these submodules have parameterization data they will be displayed as shown in the illustration below.




Selection can be made here between the individual indices. The data can be read and/or written depending on the access method. The online values are updated when reading back. If a single index is selected, clicking **Set to Default** will set all values within an index to default; selecting individual values will reset only those values. The writable values can be changed by double-clicking on the corresponding line.

# 5.4 PROFINET device diagnostics

## 5.4.1 Diagnosis at module level

At module level there is an option on the Diagnosis tab to compare the nominal and actual data for the respective module. In addition the existing diagnosis for the module can be read out.

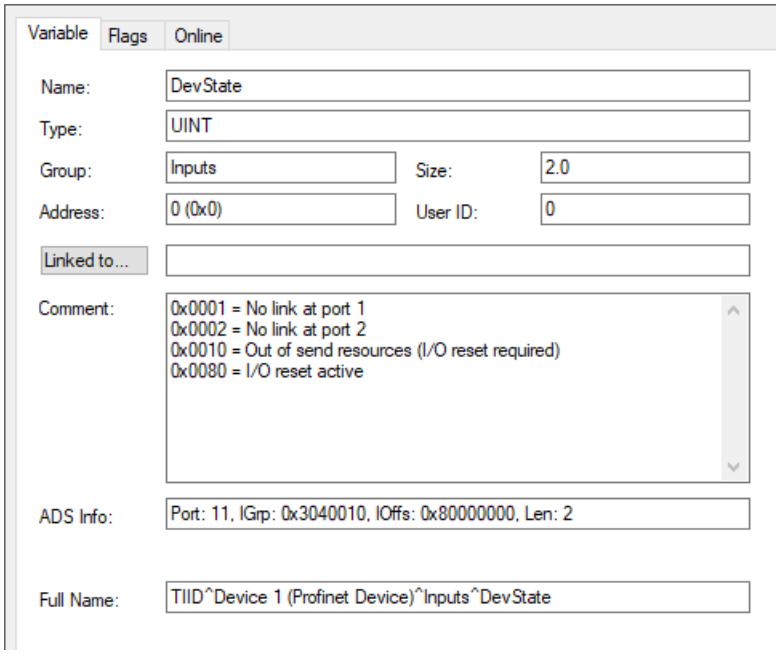
Comparison of nominal and actual population [▶ 51]

## 5.4.2 Diagnosis at submodule level

In general the submodules have the same diagnostic properties as the modules, i.e. in this case also it is currently only possible to read out the nominal and actual configuration in TwinCAT. The order of the subslot numbers is not necessarily the same as the order in the TwinCAT project. For example, in the DAP the starting point is always the Interface Submodule (ISM), but the subslot number of the ISM is defined in the GSDML and starts at 0x8000. There are 16 possible interfaces (0x8x00), each with up to 256 ports (0x80xx). An ISM is followed by the port submodules with the subslot number referred to above.

## 5.4.3 Cyclic diagnostics

Cyclic process data can be found directly below the PROFINET controller protocol. This data is only exchanged between the PROFINET driver and TwinCAT 3. It provides general information about the status of the device and the configured boxes.

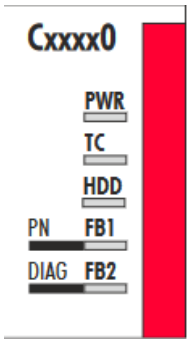



The process data **DevState**. contains information about the physical communication status of the device, such as the link status or whether the sender resources are still adequate.

The output process data **DevCtrl** currently has no function.

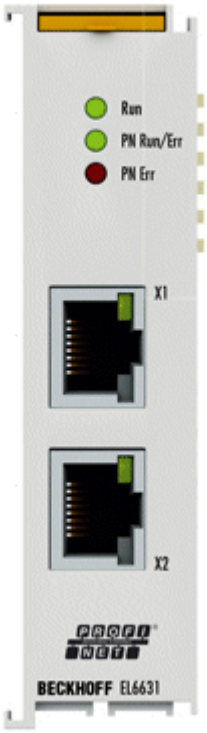
# 5.5 Diagnostic LEDs

## 5.5.1 Option interface, -M930 / -B930

| Display | LED | PROFINET status | | Meaning |
|---|---|---|---|---|
| | | green | red | |
|  | PN | off | 200 ms flashing | Power on, startup phase |
| | | 200 ms flashing | off | No PROFINET name |
| | | 1s off, 200 ms on | off | No IP address |
| | | on | off | RUN |
| | 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. |
| | | 1s off, 200 ms on | off | Problem when establishing a connection, or the actual and target configurations are different. |
| | | 200 ms on | 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 projected, 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.

## 5.5.2 EL6631 – LEDs



### LEDs for EtherCAT diagnosis

| LED | | Display | Description |
|---|---|---|---|
| RUN | green | off | State of the EtherCAT State Machine:<br>**INIT** = initialization of the terminal;<br>**BOOTSTRAP** = function for firmware updates of the terminal |
| | | flashing<br>200 ms | State of the EtherCAT State Machine:<br>**PREOP** = function for mailbox communication and different standard-settings set |
| | | off (1 s)<br>on (200 ms) | State of the EtherCAT State Machine:<br>**SAFEOP** = verification of the sync manager channels and the distributed clocks.<br>Outputs remain in safe state |
| | | on | State of the EtherCAT State Machine:<br>**OP** = normal operating state; mailbox and process data communication is possible |

### LED diagnosis PROFINET RUN/Err

| Color green | Color red | Meaning |
|---|---|---|
| off | flashing<br>200 ms | Terminal starts |
| flashing<br>200 ms | off | No name |
| 1 s off, 200 ms on | off | No IP address |
| on | off | EL terminal is parameterized |

### LED diagnosis PROFINET Err

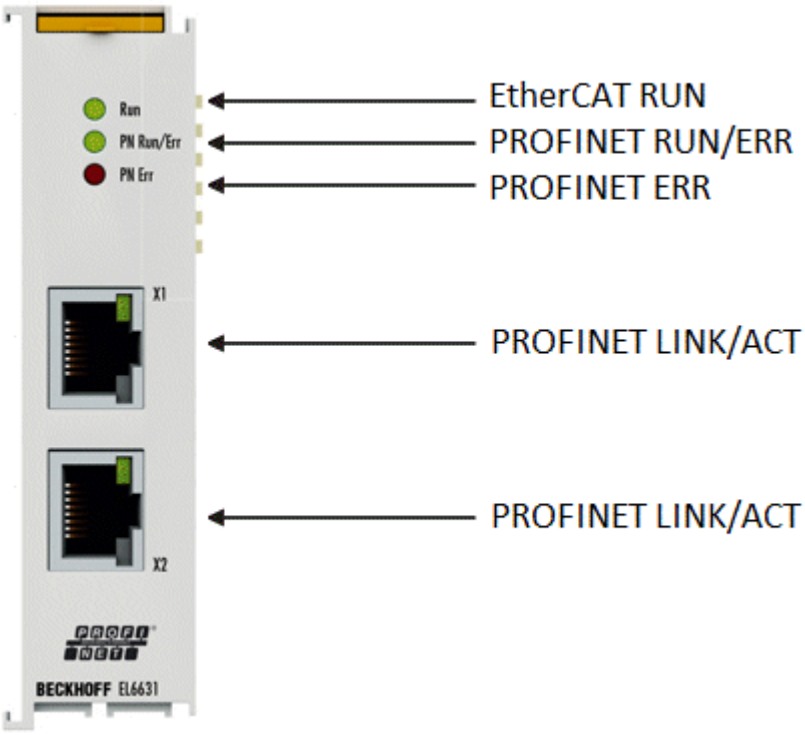
| Colours green | Colours red | Meaning |
|---|---|---|
| off | flashing<br>500 ms | no AR established with any device |
| off | 1 s off, 200 ms on | At least one device has not established an AR |
| 1 s off, 200 ms on | off | At least one device has signaled an error, e.g. that there is a module difference or that the error bit for an IO-CR is set (Problem Indicator) |
| flashing<br>200 ms | off | At least one device is signaling that its status is ‘Stop’ (Provider State Stop) |
| on | off | All PROFINET devices are in data exchange mode |

If there are several different errors, then the error that is located at the top of (or higher in) the table is always displayed.

### LEDs starting up

| Run | PN Run/Err | PN Err | Meaning |
|---|---|---|---|
| off | off | off | No electrical voltage connected to E-bus. The EL6631 must be exchanged if EtherCAT terminals behind it function. |
| off | off | red on | EL terminal is starting up; after approx. 10 seconds, the LED should go out. If this does not happen, the EL6631 module must be exchanged. |

## 5.5.3 EL6631-0010 – LEDs




**LEDs for EtherCAT diagnosis**

| LED | | Display | Description |
|---|---|---|---|
| RUN | green | off | State of the EtherCAT State Machine:<br>**INIT** = initialization of the terminal;<br>**BOOTSTRAP** = function for terminal firmware updates |
| | | flashing<br>200 ms | State of the EtherCAT State Machine:<br>**PREOP** = function for mailbox communication and different standard-settings set |
| | | off (1 s)<br>on (200 ms) | State of the EtherCAT State Machine:<br>**SAFEOP** = verification of the sync manager channels and the distributed clocks.<br>Outputs remain in safe state |
| | | on | State of the EtherCAT State Machine:<br>**OP** = normal operating state; mailbox and process data communication is possible |

**LED diagnosis PROFINET RUN/Err**

| Colors green | Colors red | Meaning |
|---|---|---|
| on | off | EL terminal is parameterized |
| off (1 s)<br>on (200 ms) | off | EL6631-0010 does not have an IP address |
| flashing<br>200 ms | off | EL6631-0010 still has not received a PROFINET name |
| off | flashing<br>200 ms | Terminal starts |

### LED diagnosis PROFINET Err

| Colors green | Colors red | Meaning |
|---|---|---|
| on | off | EL terminal is exchanging data |
| flashing<br>200 ms | off | EL terminal is exchanging data, but the provider status is stopped |
| off (1 s)<br>on (200 ms) | off | EL terminal is exchanging data, but the modules are different |
| off | flashing<br>500 ms | No AR established,<br>establishment of connection has not been initialized |
| flashing<br>500 ms | flashing<br>500 ms | Identify EL terminal through PROFINET "flashing" |

### LEDs starting up

| Run | PN Run/Err | PN Err | Meaning |
|---|---|---|---|
| off | off | off | No electrical voltage connected to E-bus. The EL6631-0010 must be exchanged if EtherCAT terminals behind it need to function. |
| off | off | red on | EL terminal is starting up; after approx. 10 seconds, the LED should go out. If this does not happen, the EL6631-0010 module must be exchanged. |

# 6 Appendix

## 6.1 Troubleshooting

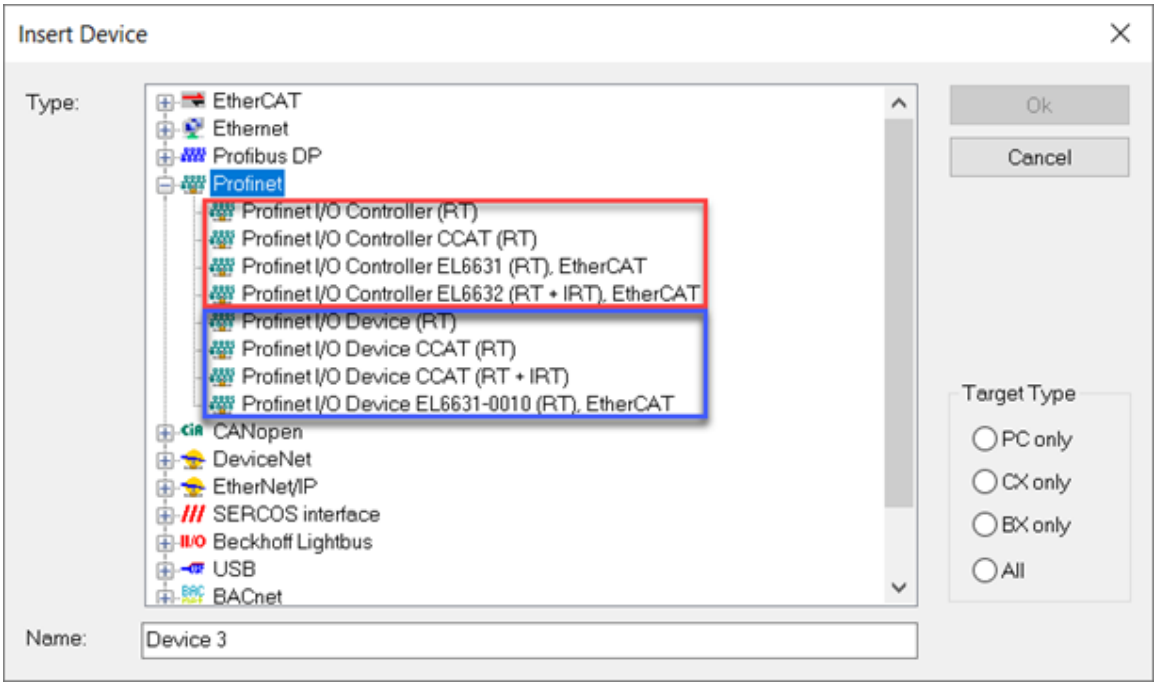
The following section provides information about settings in the PROFINET system configuration. Failure to comply with this can lead to undesirable behavior. Here you will find diagnosis information.

**Device description file (GSDML) / DAP (DeviceAccessPoint)**

- Is the GSDML available on the system?
- Do the versions of both systems match?
  - It is recommended to use the same GSDML/DAP versions on both systems.
  - Is the latest version used?
- Is the GSDML in the correct path?
  - TwinCAT 2: TwinCAT2: C:\TwinCAT\Io\ProfiNet
  - TwinCAT 3: C:\TwinCAT\3.1\Config\Io\Profinet
- Is the correct GSDML being used?
  - Version
  - It may be necessary to contact the vendor/manufacturer or search for the appropriate GSDML on the vendor's website.

**Has the correct PROFINET adapter been created?**

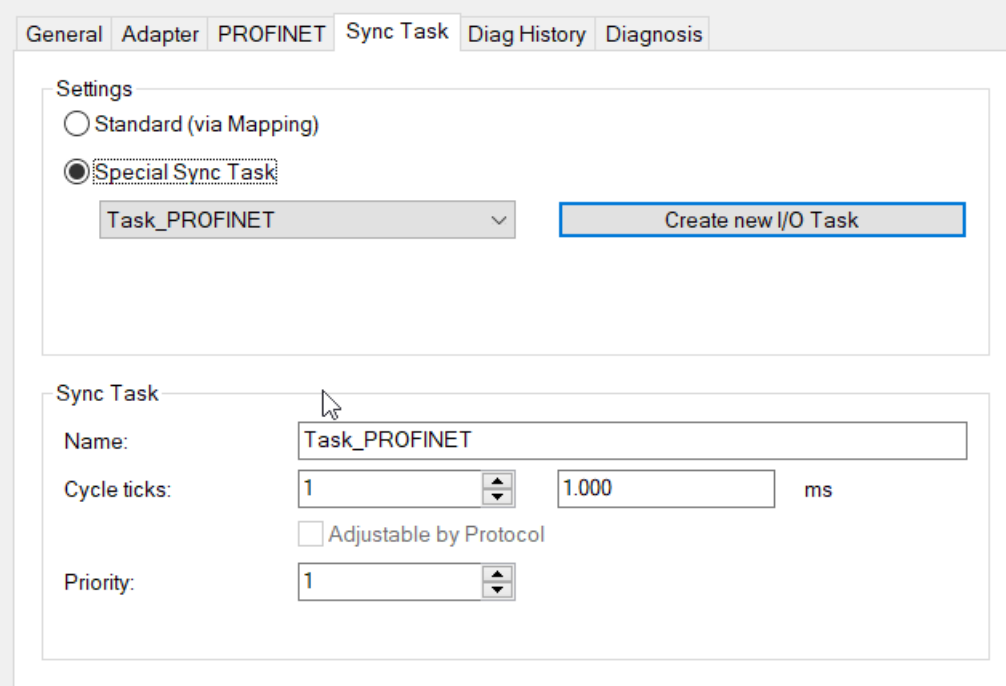
- Controller or device (master or slave)?




  - Example, wrong / correct

**Task configuration**

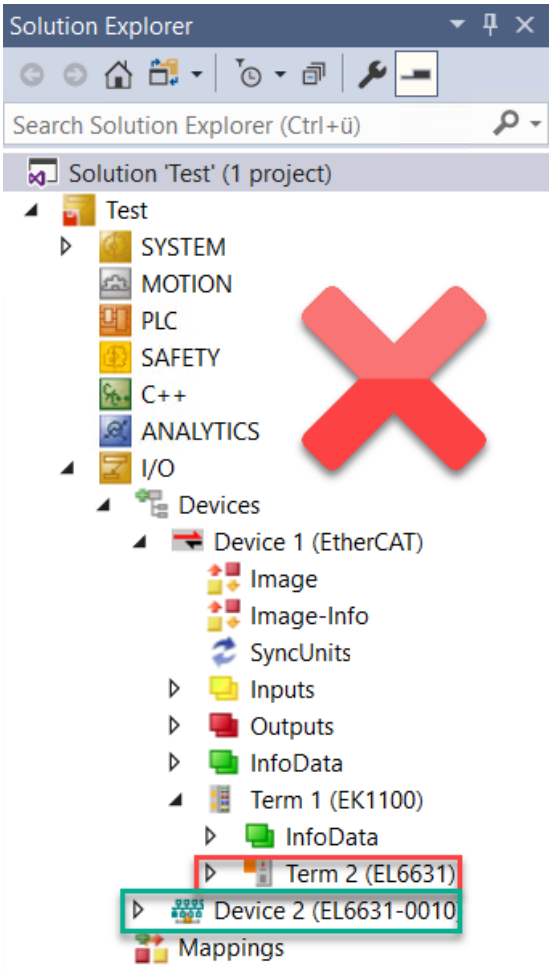
- Was a free running task created?
  - Or was a special Sync Task used?
- Cycle time to base 2?
  - 1 ms, 2 ms, 4 ms, 8 ms, ….

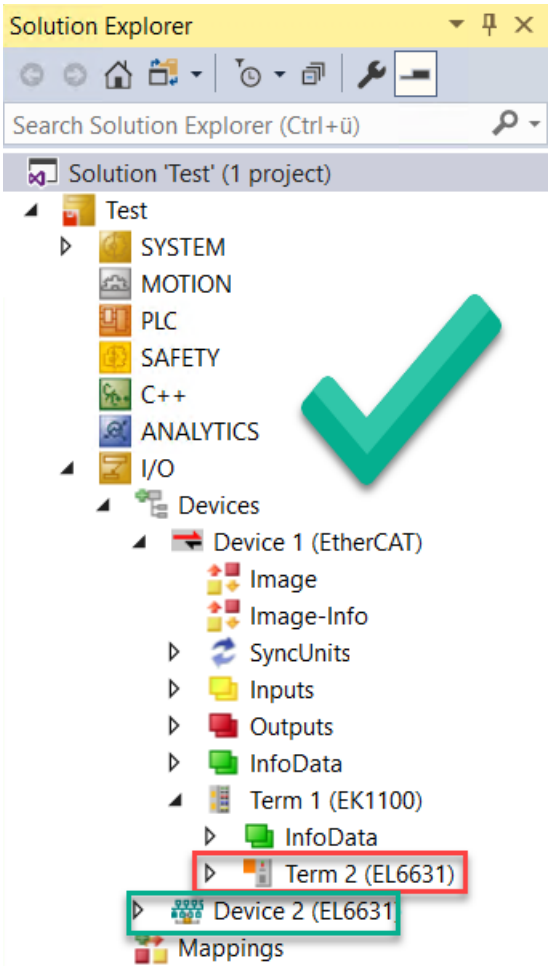



- Further information in chapter Sync Task

**EtherCAT Terminals EL663x-00x0**

- Was the correct terminal used?
  - EL663x-0000 cannot be used as device
  - EL6631-0010 cannot be used as controller




- EtherCAT diagnostics
  - EtherCAT status = operational (OP)
  - WcState = 0 (data valid)

**BoxStates of the PROFINET devices**

- Communication established?

## 6.2 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: https://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

**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 5246 963 157 |
| Fax: | +49 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 5246 963 460 |
| Fax: | +49 5246 963 479 |
| e-mail: | service@beckhoff.com |

**Beckhoff Headquarters**

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

| | |
|---|---|
| Phone: | +49 5246 963 0 |
| Fax: | +49 5246 963 198 |
| e-mail: | info@beckhoff.com |
| web: | https://www.beckhoff.com |

More Information:
**www.beckhoff.com/tf6271**

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