Documentation

EL6631-0010

PROFINET DEVICE Supplement

Version: 3.1.0
Date: 2019-10-21

BECKHOFF

Table of contents

BECKHOFF

1   Foreword

1.1   Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these 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 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 instructions

In this documentation the following instructions are used.
These instructions must be read carefully and followed without fail!

⚠ **DANGER**
Serious risk of injury!
Failure to follow this safety instruction directly endangers the life and health of persons.

⚠ **WARNING**
Risk of injury!
Failure to follow this safety instruction endangers the life and health of persons.

⚠ **CAUTION**
Personal injuries!
Failure to follow this safety instruction can lead to injuries to persons.

NOTE
Damage to environment/equipment or data loss
Failure to follow this instruction can lead to environmental damage, equipment damage or data loss.

i   **Tip or pointer**

This symbol indicates information that contributes to better understanding.

1.3 Documentation issue status

Version	Comment
3.1.0	• Update structure • Update revision status
3.0.1	• Update chapter "Technical data" • Update structure • Update revision status
3.0.0	• Migration • Update structure • Update revision status
2.0.0	• Update structure, addenda Technical Data
1.1.0	• Chapter "SharedDevice" added, EC LED description amended, tree structure amended
1.0.0	• First version

Firmware and hardware versions

The software version of the Profinet device supplement can be taken from the corresponding TwinCAT build number.

For the EL6631-0010, the firmware and hardware versions (delivery state) can be taken from the serial number printed on the side of the terminal.

1.4 Version identification of EtherCAT devices

Designation

A Beckhoff EtherCAT device has a 14-digit designation, made up of

- family key
- type
- version
- revision

Example	Family	Type	Version	Revision
EL3314-0000-0016	EL terminal (12 mm, non-pluggable connection level)	3314 (4-channel thermocouple terminal)	0000 (basic type)	0016
ES3602-0010-0017	ES terminal (12 mm, pluggable connection level)	3602 (2-channel voltage measurement)	0010 (high-precision version)	0017
CU2008-0000-0000	CU device	2008 (8-port fast ethernet switch)	0000 (basic type)	0000

Notes

- The elements mentioned above result in the **technical designation**. EL3314-0000-0016 is used in the example below.
- EL3314-0000 is the order identifier, in the case of "-0000" usually abbreviated to EL3314. "-0016" is the EtherCAT revision.
- The **order identifier** is made up of
  - family key (EL, EP, CU, ES, KL, CX, etc.)
  - type (3314)
  - version (-0000)

- The **revision** -0016 shows the technical progress, such as the extension of features with regard to the EtherCAT communication, and is managed by Beckhoff.
  In principle, a device with a higher revision can replace a device with a lower revision, unless specified otherwise, e.g. in the documentation.
  Associated and synonymous with each revision there is usually a description (ESI, EtherCAT Slave Information) in the form of an XML file, which is available for download from the Beckhoff web site.
  From 2014/01 the revision is shown on the outside of the IP20 terminals, see Fig. *"EL5021 EL terminal, standard IP20 IO device with batch number and revision ID (since 2014/01)"*.

- The type, version and revision are read as decimal numbers, even if they are technically saved in hexadecimal.

Identification number

Beckhoff EtherCAT devices from the different lines have different kinds of identification numbers:

Production lot/batch number/serial number/date code/D number

The serial number for Beckhoff IO devices is usually the 8-digit number printed on the device or on a sticker. The serial number indicates the configuration in delivery state and therefore refers to a whole production batch, without distinguishing the individual modules of a batch.

Structure of the serial number: **KK YY FF HH**

KK - week of production (CW, calendar week)
YY - year of production
FF - firmware version
HH - hardware version

Example with
Ser. no.: 12063A02:   12 - production week 12 06 - production year 2006 3A - firmware version 3A 02 - hardware version 02

Exceptions can occur in the **IP67 area**, where the following syntax can be used (see respective device documentation):

Syntax: D ww yy x y z u

D - prefix designation
ww - calendar week
yy - year
x - firmware version of the bus PCB
y - hardware version of the bus PCB
z - firmware version of the I/O PCB
u - hardware version of the I/O PCB

Example: D.22081501 calendar week 22 of the year 2008 firmware version of bus PCB: 1 hardware version of bus PCB: 5 firmware version of I/O PCB: 0 (no firmware necessary for this PCB) hardware version of I/O PCB: 1

Unique serial number/ID, ID number

In addition, in some series each individual module has its own unique serial number.

See also the further documentation in the area

- IP67: EtherCAT Box

- Safety: TwinSafe

- Terminals with factory calibration certificate and other measuring terminals

Examples of markings

Fig. 1: EL5021 EL terminal, standard IP20 IO device with serial/ batch number and revision ID (since 2014/01)

Fig. 2: EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number

Fig. 3: CU2016 switch with serial/ batch number
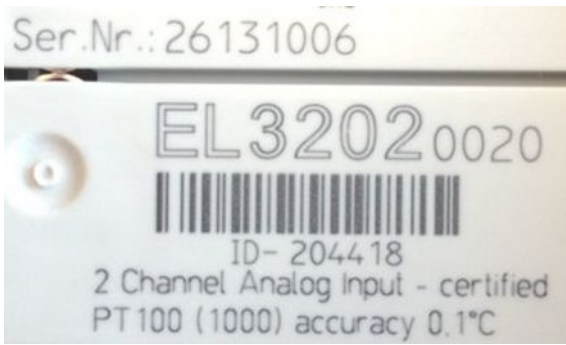
Fig. 4: EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418
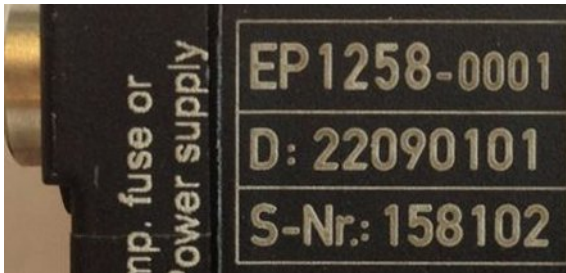
Fig. 5: EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102
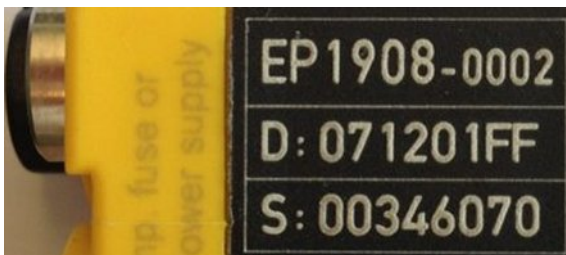

Fig. 6: EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070

Fig. 7: EL2904 IP20 safety terminal with batch number/ date code 50110302 and unique serial number 00331701

Fig. 8: ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201

1.4.1   Beckhoff Identification Code (BIC)

The Beckhoff Identification Code (BIC) is increasingly being applied to Beckhoff products to uniquely identify the product. The BIC is represented as a Data Matrix Code (DMC, code scheme ECC200), the content is based on the ANSI standard MH10.8.2-2016.
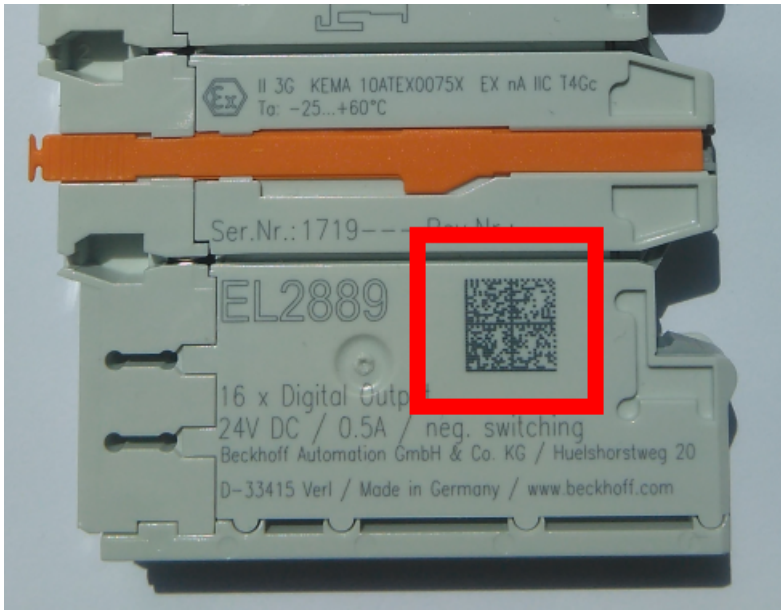
Fig. 9: BIC as data matrix code (DMC, code scheme ECC200)

The BIC will be introduced step by step across all product groups.

Depending on the product, it can be found in the following places:

- on the packaging unit
- directly on the product (if space suffices)
- on the packaging unit and the product

The BIC is machine-readable and contains information that can also be used by the customer for handling and product management.

Each piece of information can be uniquely identified using the so-called data identifier (ANSI MH10.8.2-2016). The data identifier is followed by a character string. Both together have a maximum length according to the table below. If the information is shorter, spaces are added to it. The data under positions 1 to 4 are always available.

The following information is contained:

Item no.	Type of information	Explanation	Data identifier	Number of digits incl. data identifier	Example
1	Beckhoff order number	**Beckhoff order number**	1P	8	1P072222
2	Beckhoff Traceability Number (BTN**)**	**Unique serial number, see note below**	S	12	SBTNk4p562d7
3	Article description	**Beckhoff article description, e.g. EL1008**	1K	32	1KEL1809
4	Quantity	**Quantity in packaging unit, e.g. 1, 10, etc.**	Q	6	Q1
5	Batch number	Optional: Year and week of production	2P	14	2P401503180016
6	ID/serial number	Optional: Present-day serial number system, e.g. with safety products	51S	12	51S678294104
7	Variant number	Optional: Product variant number on the basis of standard products	30P	32	30PF971, 2*K183
...					

Further types of information and data identifiers are used by Beckhoff and serve internal processes.

Structure of the BIC

Example of composite information from item 1 to 4 and 6. The data identifiers are marked in red for better display:

BTN

An important component of the BIC is the Beckhoff Traceability Number (BTN, item no. 2). The BTN is a unique serial number consisting of eight characters that will replace all other serial number systems at Beckhoff in the long term (e.g. batch designations on IO components, previous serial number range for safety products, etc.). The BTN will also be introduced step by step, so it may happen that the BTN is not yet coded in the BIC.

NOTE
This information has been carefully prepared. However, the procedure described is constantly being further developed. We reserve the right to revise and change procedures and documentation at any time and without prior notice. No claims for changes can be made from the information, illustrations and descriptions in this information.

BECKHOFF

2 TwinCAT supplement

2.1 TwinCAT 2.10

2.1.1 PROFINET device integration under TwinCAT 2.10

ℹ **Installation**

An installation version is no longer available for TwinCAT 2.10. Use TwinCAT V2.11 or contact Beckhoff Support [▶ 68].

1. Integration of the PROFINET protocol

First of all, a real time Ethernet device must be added. This interface must be assigned to an adaptor class. This is composed of the MAC and IP address of the network card.
Hence, e.g., if a CX9000 is to be parameterized, the target system must first be accessed remotely; the corresponding Ethernet port can subsequently be selected.

Thereafter, a PROFINET protocol is added by means of the right mouse button. Precisely one TwinCAT device protocol can be appended!
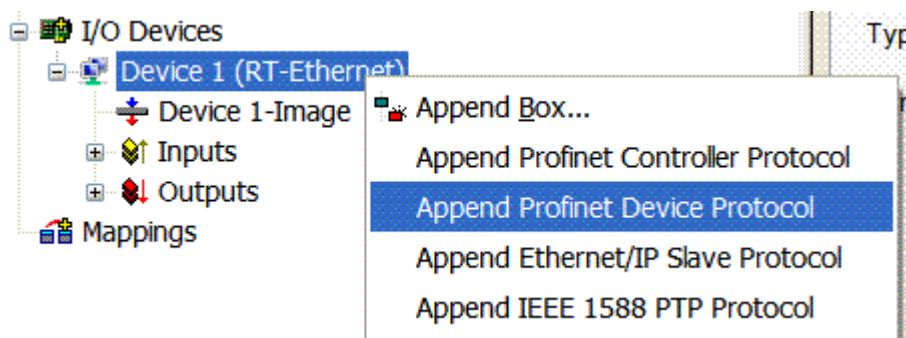

Fig. 10: Inserting a Profinet device protocol

The NetID used can be found under the "Protocol" tab. It is, e.g., necessary for an ADS access. Besides that, there is a possibility in the "Sync Task" tab to append a free-running task for PROFINET communication (this is recommended). This allows the RealTime communication to run independently of a PLC task. For performance reasons, however (e.g. in the case of a CX9000), the Sync Task can be dispensed with and standard mapping takes place, i.e. a PLC task, for example, must be running in order for PROFINET communication to be possible.
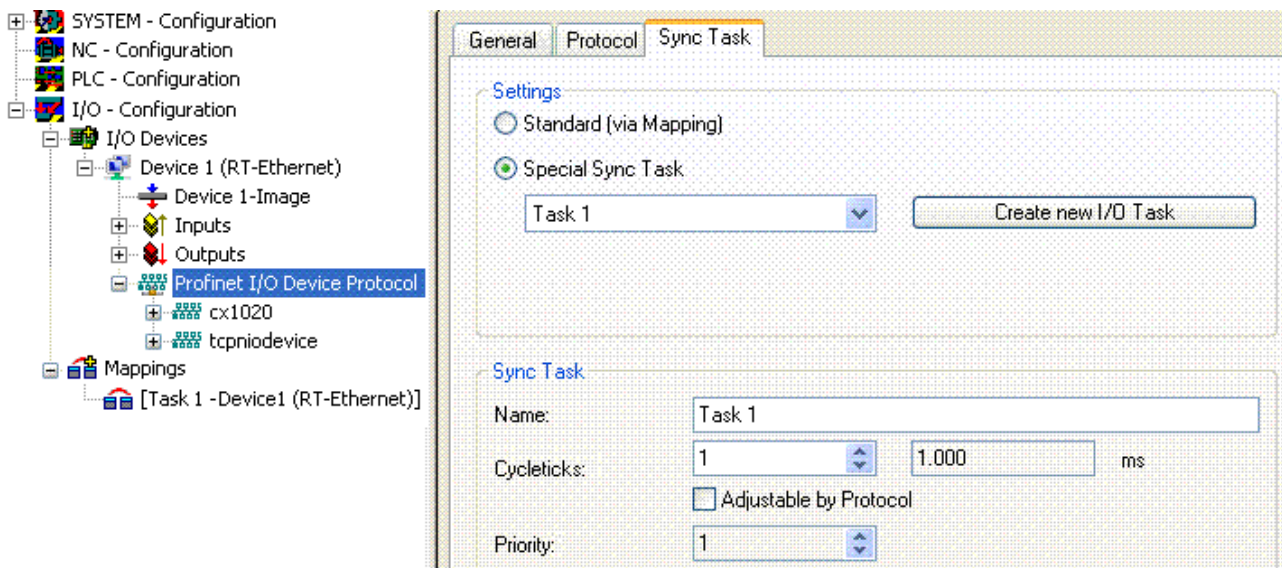
Fig. 11: "Sync Task" tab

Next, a box is integrated in the form of a GSDML (right click on "PROFINET I/O Device Protocol"). In TwinCAT it is also possible to configure several virtual PROFINET devices. Each device is again to be viewed as its own adaptor, i.e. each virtual device is assigned a default MAC address from a reserved range. To avoid possible address conflicts, the MAC address of virtual devices can also be changed. The name which the device was given in the strand (or in the "General" tab), is also taken as a PROFINET station name following a restart. On the basis of different MAC addresses and names, a PROFINET controller can now distinguish several devices in the strand. In turn, the individual virtual devices can then be linked with terminals from the E-bus, e.g. via a PLC task. As a result of this, e.g., several EL6601 terminals in the E-bus can be configured as individual PROFINET devices!

In PROFINET, different user profiles can be defined within a device (API - Application Process Instance). In Twin CAT, precisely one API is permitted / supported per device!

In accordance with PROFINET, the DAP (Device Access Point) is always defined on slot 0. Device-specific data is defined in it. In accordance with PROFINET, several DAPs can be defined; however, in TwinCAT there is currently precisely one!

2. Process data

The process data can be inserted from slot 1. This takes place by integrating the desired module in the API. Each module (slot) has at present precisely one sub-module (sub-slot). The configuration of the modules and hence the creation of the process data image on the PROFINET side takes place by means of specifying data types / data widths (e.g. byte, word, dword, float 32).

The designation of the inputs and outputs always takes place in TwinCAT from the TwinCAT point of view, i.e. the inputs are from the PROFINET point of view (from the controller) the outputs! Therefore the data of the PROFINET I/O devices is represented as follows:
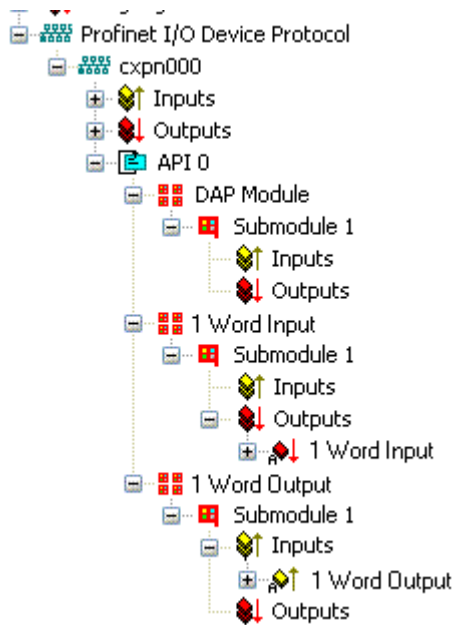
Fig. 12: Representation of the inputs/outputs of the PROFINET I/O device

This representation applies only to the TwinCAT PROFINET device, in the case of a TwinCAT PROFINET controller, the view will correlate again!

3. General settings

In the "PROFINET Device" tab, the Instance ID and the PROFINET Frame ID as well as the Client and Server UDP port can be changed. However, the default settings suffice for normal operation!
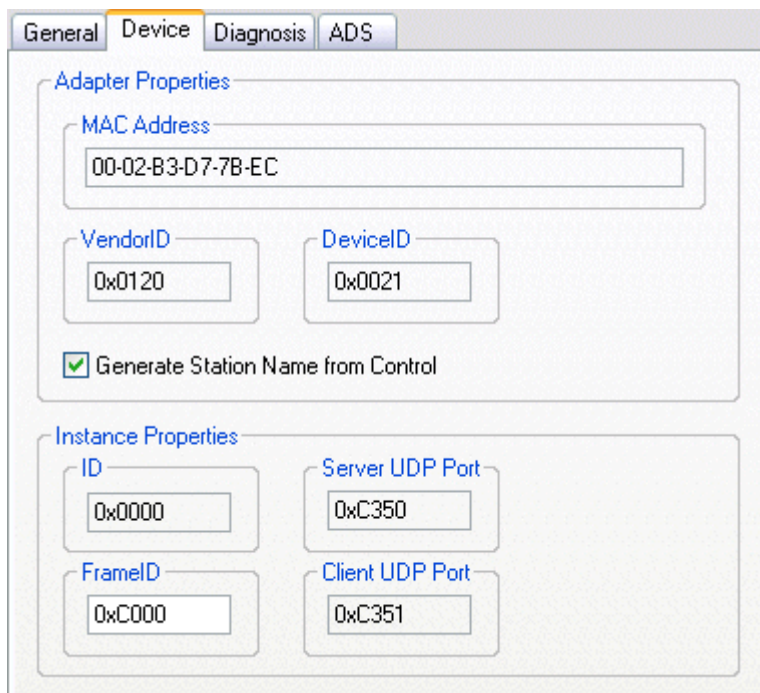

Fig. 13: "Device" tab

There is one further way of assigning the PROFINET station name. This is comparable with the DIP switches in the BK9103 and can be carried out via a PLC task. For activation, "Generate Station Name from Control" must be activated. For clarification, 000 will be appended to the previous tree name (default: "tcpniodevice").

i ● **Tree name**

This tree name no longer corresponds to the PROFINET station name! The Ctrl WORD of the PROFINET protocol is used to help assign a name. This means that the number that is entered (range of values 0 - 255) is appended to the existing station name. In addition, the Ctrl WORD must be linked to a task. It is subsequently necessary to restart TwinCAT. If, e.g., the task now specifies a value of 11 for the linked Ctrl WORD, its previous station name will change from, for example, "tcpniodevice" to "tcpniodevice011". The current tree name remains "tcpniodevice000".

The current station names and the IP configuration used can be checked via the "PROFINET diagnosis" tab.

4. Cycle times

The device can be operated in the minimum cycle time of 1 ms defined by PROFINET!

If TwinCAT is operated in FREERUN mode, care must be taken that the freerun cycle set is no longer than the PROFINET cycle!
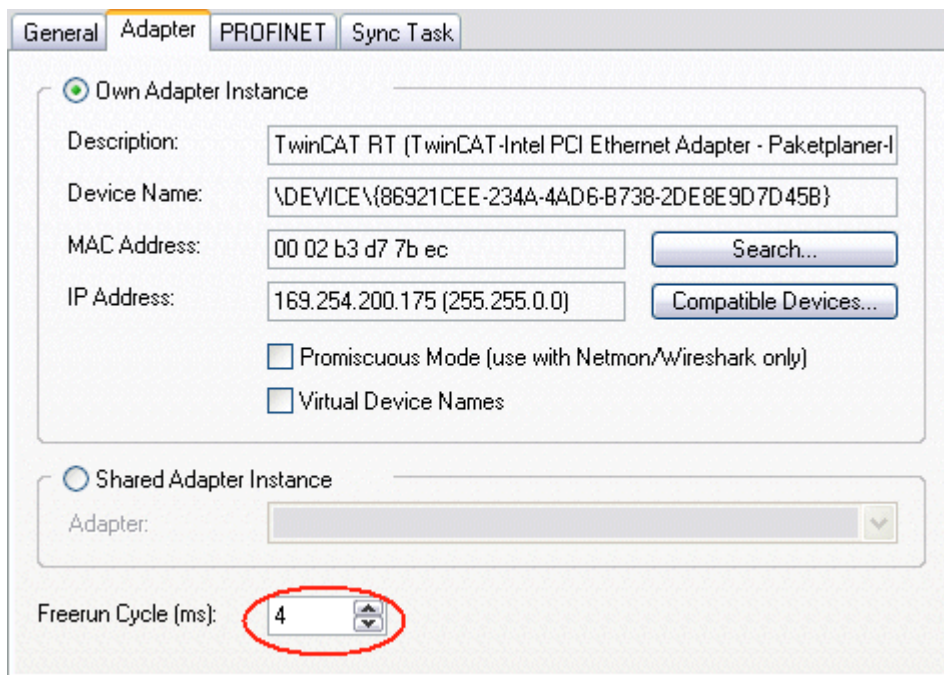

Fig. 14: "Adapter" tab

If TwinCAT is to be operated in RUN mode, a task must be created. In the simplest case, this is the aforementioned SyncTask. The call time for the task must not be longer than the PROFINET cycle time! If a second task is created, e.g. for the PLC, this can also run more slowly.
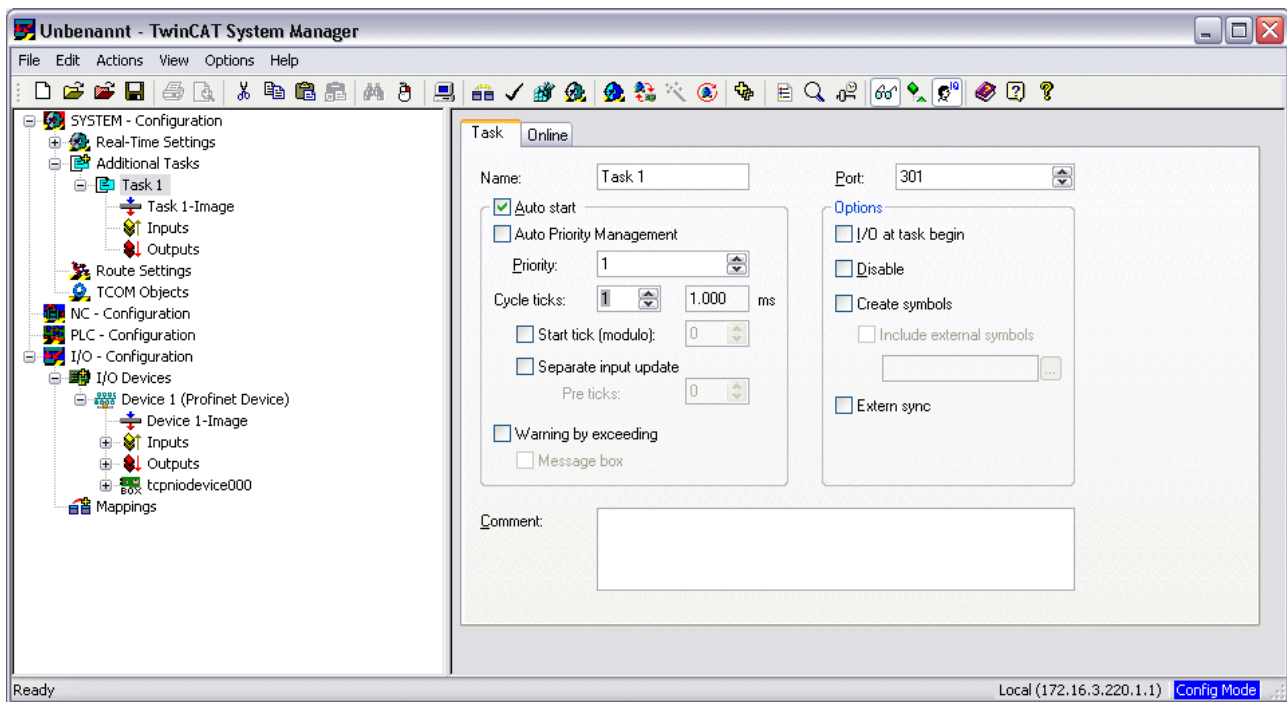
BECKHOFF

Fig. 15: Creating a task

In the case of less powerful devices (e.g. CX1000, CX9000 with E-bus), the CPU load may reach its limits. However, the performance can be improved again by means of the following considerations/measures:

- Is it possible to work with just one PLC task?
- Does the cycle time have to be 1 ms?
- If two tasks are used, can the PLC task time be shortened?
- Will several virtual PROFINET devices by required?

2.2 TwinCAT 2.11

2.2.1 Technical data - PROFINET RT

When using TwinCAT 2.11, all the target systems, for instance the CX Controllers, must have TwinCAT version 2.11 installed.

Technical data	Supplement
PROFINET Version	RT Class 1 ConformanceClassB
Number of device interfaces)[1]	8
Topology	variable
Quantity of user data	per device, maximum one Ethernet frame length)[2] 1500 bytes of user data, inc. IOPS and IOCS
Cycle time	≥ 1 ms

)[1] see the chapter on virtual PROFINET devices
)[2] Depending on the cycle time, the PROFINET cycle time, and the CPU being used

ⓘ Conditions required for operation

The following points must be observed when using the PROFINET supplement:
- Only Ethernet cards with Intel chipset allowed.
- RealTime Ethernet driver must be installed.
- No other RealTime protocols must be connected through this interface.
- The real-time capability can only be guaranteed in the transmit direction; in the receive direction, the possibility of incorrect use means that it cannot be guaranteed. This might, for example, be the copying of large quantities of data through this interface.
  It is recommended that the PROFINET network is separated from other networks.

2.2.2 PROFINET device integration under TwinCAT 2.11

This description applies to the PROFINET DEVICE Supplement.

ⓘ Installation

For a correct installation at least TwinCAT 2.11 build 1546 should be installed. With older TwinCAT versions we recommend an update. In other cases please contact Beckhoff Support [▶ 68].

1. Integration of the PROFINET protocol

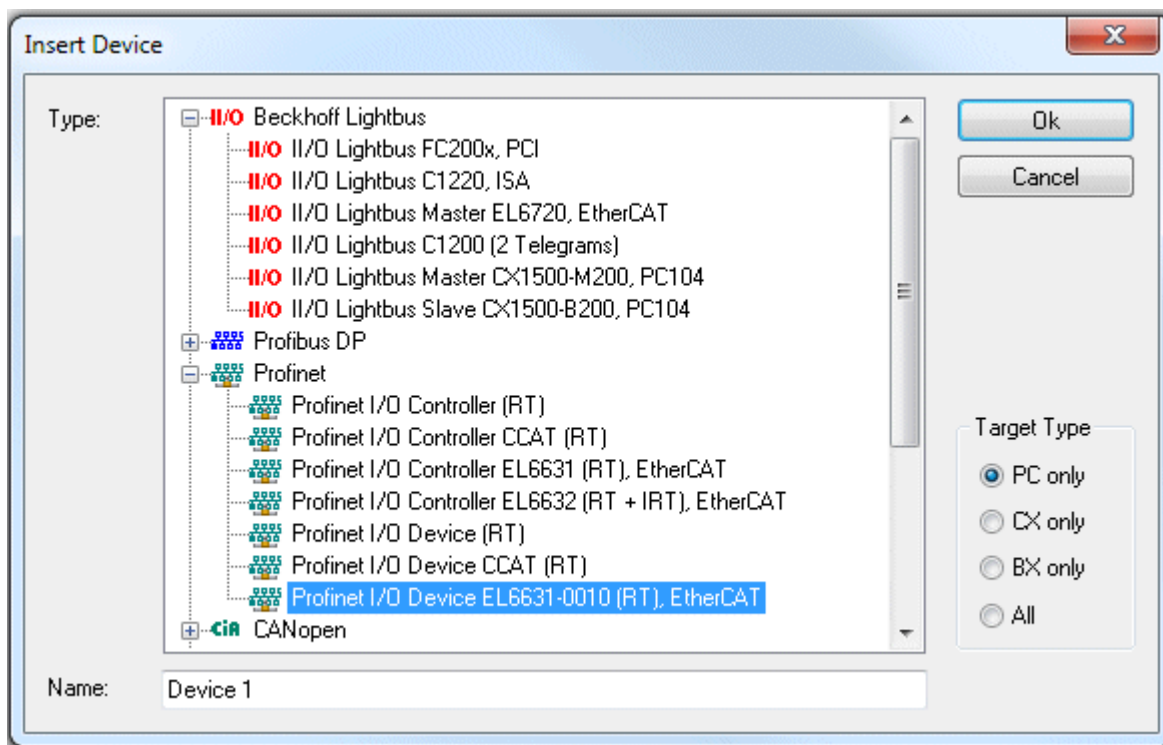A PROFINET device must first be added.

Fig. 16: "Insert Device" dialog

ⓘ Network card

If no card is found, either the TwinCAT is not in Config mode, or the real-time Ethernet driver has not been installed properly.

This interface must be assigned to an adaptor class. This is composed of the MAC and IP address of the network card. Look for the corresponding network card by using "Search".
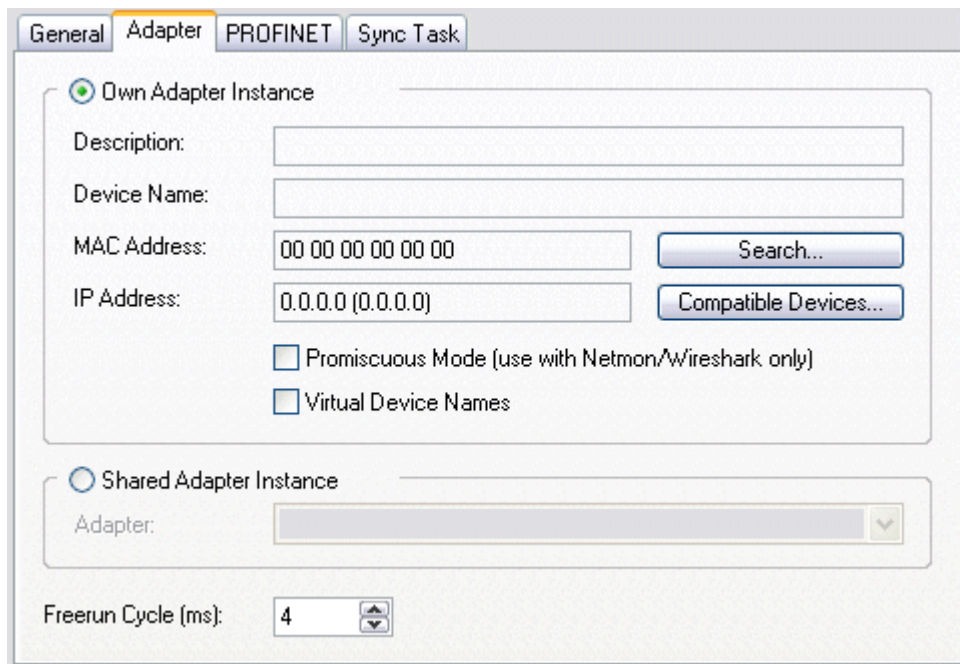
BECKHOFF

Fig. 17: Search button, network card selection

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, and has nothing to do with the PROFINET IP address that will be used later.

> **● IP address of the network card**
>
> Please note that the IP address of the PROFINET device and the IP address of the Ethernet card are not identical! The PROFINET driver works with a virtual IP address that must not be identical to the IP address of the operating system!

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.

Shared Adapter Interface:
Is included for compatibility reasons, but cannot be used with PROFINET.

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!

If, e.g., a CX9000 is to be parameterized, the target system must first be accessed remotely; the corresponding Ethernet port can subsequently be selected.

The NetID used can be found under the "Protocol" tab. It is, e.g., necessary for an ADS access. Besides that, there is a possibility in the "Sync Task" tab to append a free-running task for PROFINET communication (this is recommended). This allows the RealTime communication to run independently of a PLC task. For performance reasons, however (e.g. in the case of a CX9000), the Sync Tasks can be dispensed with and standard mapping takes place, i.e. a PLC task, for example, must be running in order for PROFINET communication to be possible.

A PROFINET protocol is then added by means of the right mouse button. Precisely one TwinCAT device protocol can be appended!
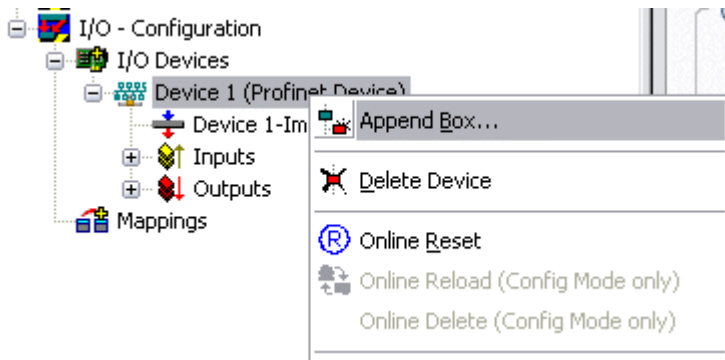
Fig. 18: Inserting a PROFINET protocol (TwinCAT device protocol)

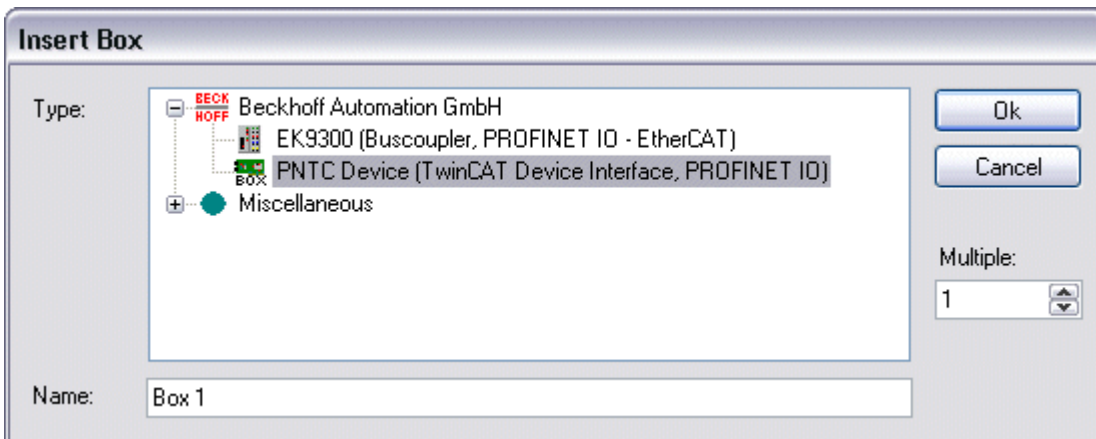A Box in the form of a GSDML is then integrated (right mouse button on "PROFINET Device").

Fig. 19: "Insert Box" dialog

Integration with any Ethernet interface without port diagnosis according to Version 2.0 without LLDP.
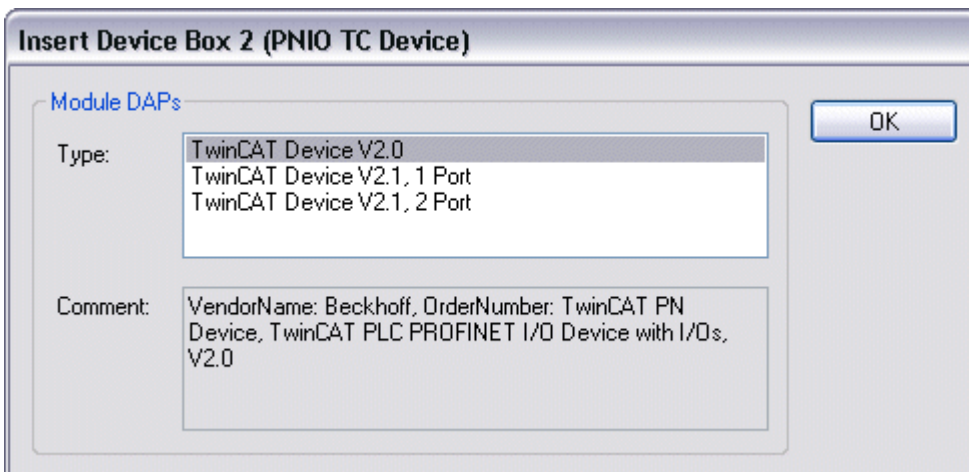

Fig. 20: Integrating any Ethernet interface without port diagnosis

Integration with a simple Ethernet interface (without switch, for example the CX1010 or FC90xx) with port diagnosis according to Version 2.1.
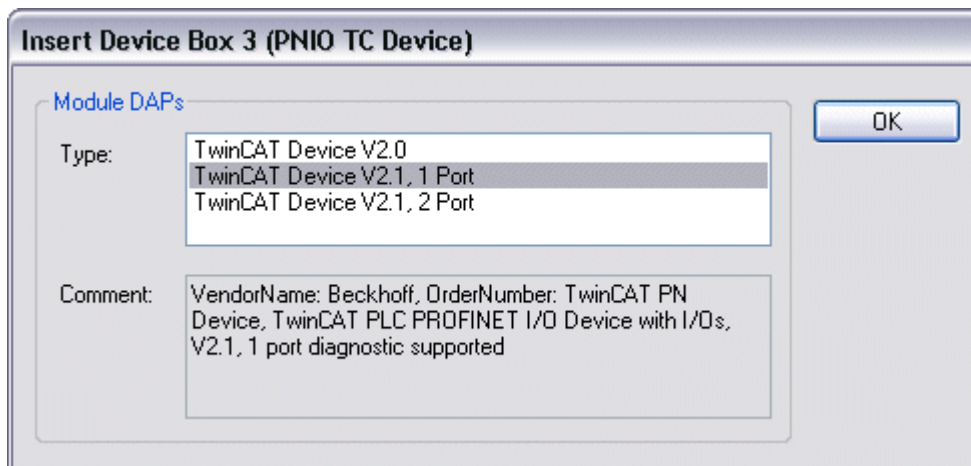
Fig. 21: Integrating a simple Ethernet interface with port diagnosis

Integration with a switched Ethernet interface (the CX1020, for example) with port diagnosis according to Version 2.1.
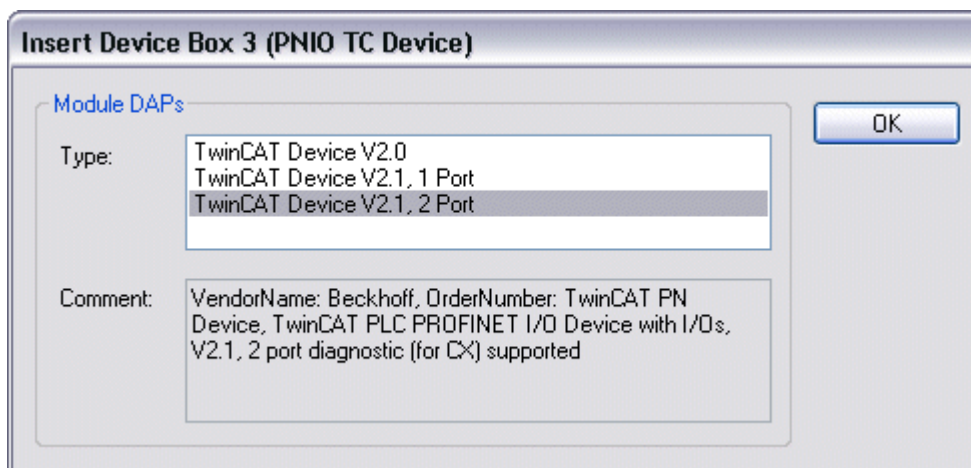

Fig. 22: Integration with a switched Ethernet interface with port diagnosis

Virtual PROFINET device

It is also possible in TwinCAT to configure a number of virtual PROFINET devices (a maximum of 7 in the Supplement, and 1 in the EL6631-0010). Each device is again to be viewed as its own adaptor, i.e. each virtual device is assigned a default MAC address from a reserved range. To avoid possible address conflicts, the last byte of the MAC address of virtual devices can also be changed. The name which the device was given in the strand (or in the "General" tab) is also taken as a PROFINET station name following a restart. On the basis of different MAC addresses and names, a PROFINET controller can now distinguish several devices in the strand.

It is therefore possible to use 8 devices per Ethernet interface in the PROFINET Device Supplement, and to communicate with 8 different PROFINET controllers.

In PROFINET, different user profiles can be defined within a device (API - Application Process Instance). Precisely one API is permitted / supported for each device in TwinCAT.

In accordance with PROFINET, the DAP (Device Access Point) is always defined on slot 0. Device-specific data is defined in it. In accordance with PROFINET, several DAPs can be defined; however, in TwinCAT there is currently precisely one!

2. Process data

The process data can be inserted from slot 1. This takes place by integrating the desired module in the API. Each module (slot) has at present precisely one sub-module (sub-slot). The configuration of the modules and hence the creation of the process data image on the PROFINET side takes place by means of specifying data types / data widths (e.g. byte, word, dword, float 32).

The designation of the inputs and outputs always takes place in TwinCAT from the TwinCAT point of view, i.e. the inputs are from the PROFINET point of view (from the controller) the outputs! Therefore the data of the PROFINET I/O devices is represented as follows:
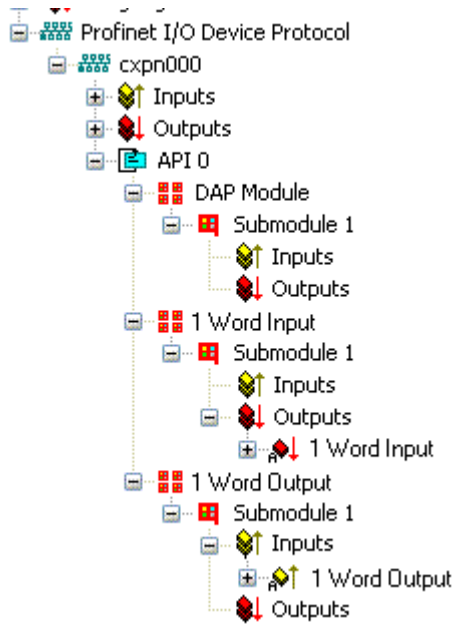

Fig. 23: Representation of the input/output data of the PROFINET I/O device

This representation applies only to the TwinCAT PROFINET device, in the case of a TwinCAT PROFINET controller, the view will correlate again!

3. General settings

The instance ID can be changed on the "PROFINET Devices" tab. However, the default settings suffice for normal operation!
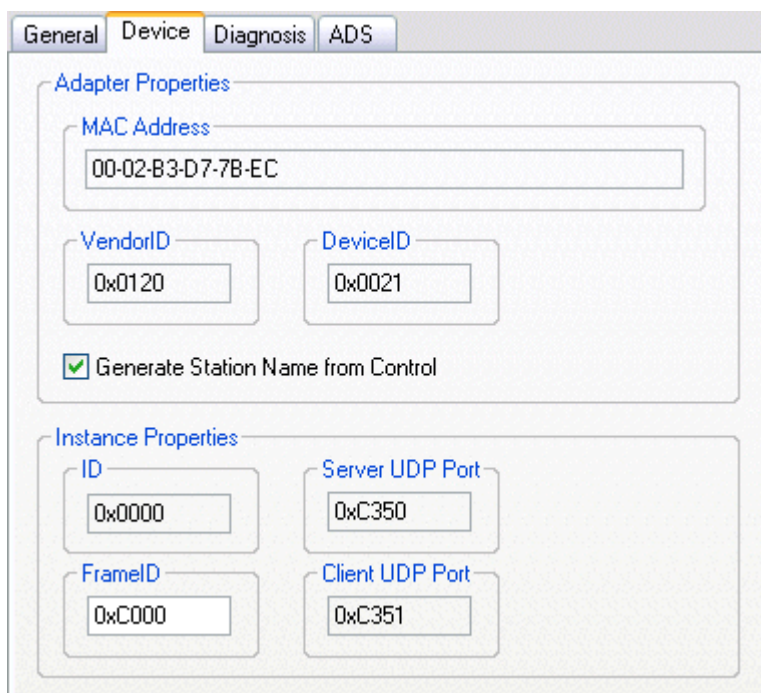

Fig. 24: "Device" tab

There is one further way of assigning the PROFINET station name. This is comparable with the DIP switches in the BK9103 and can be carried out via a PLC task. For activation, "Generate Station Name from Control" must be activated. For clarification, 000 will be appended to the previous tree name (default: "tcpniodevice").

ⓘ **Tree name**

This tree name no longer corresponds to the PROFINET station name! The Ctrl WORD of the PROFINET protocol is used to help assign a name. This means that the number that is entered (range of values 0 - 255) is appended to the existing station name. In addition, the Ctrl WORD must be linked to a task. It is subsequently necessary to restart TwinCAT. If, e.g., the task now specifies a value of 11 for the linked Ctrl WORD, its previous station name will change from, for example, "tcpniodevice" to "tcpniodevice011". The current tree name remains "tcpniodevice000".

The current station names and the IP configuration used can be checked via the "PROFINET diagnosis" tab.

4. Cycle times

The device can be operated in the minimum cycle time of 1 ms defined by PROFINET!

Fig. 25: "Adapter" tab, setting the cycle time

If TwinCAT is to be operated in RUN mode, a task must be created. In the simplest case, this is the aforementioned SyncTask. The call time for the task must not be longer than the PROFINET cycle time! If a second task is created, e.g. for the PLC, this can also run more slowly.
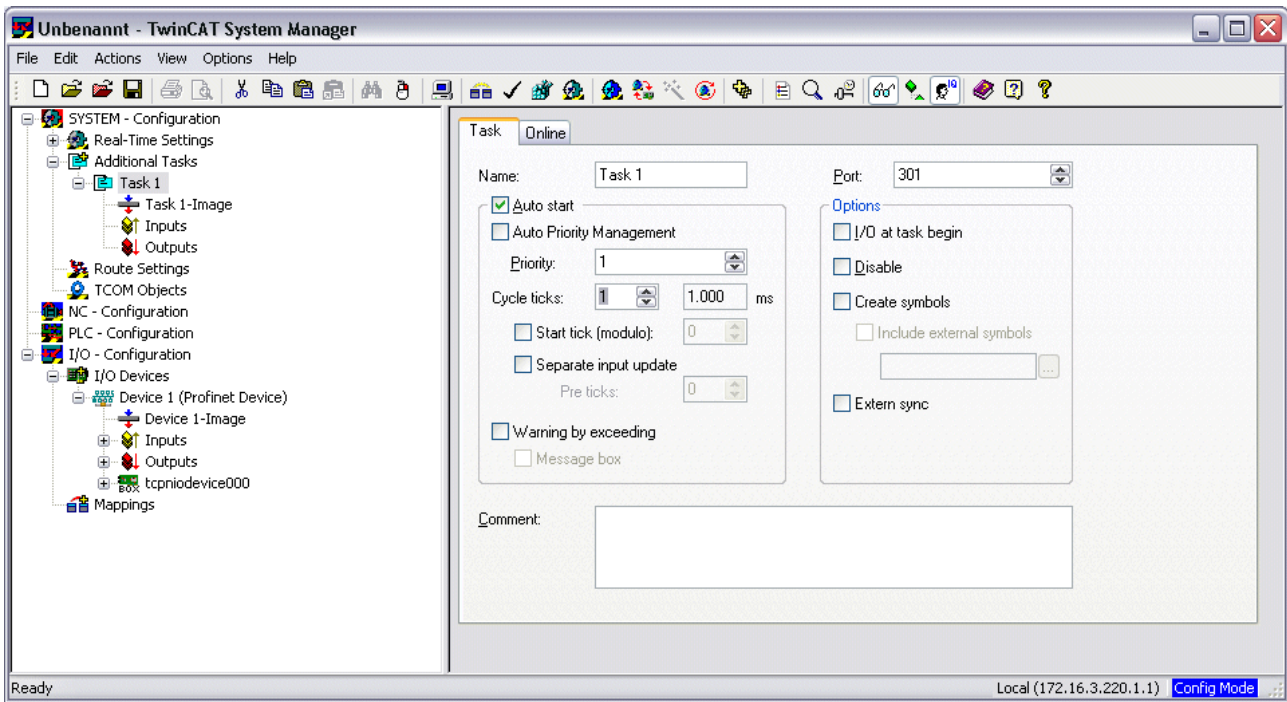
Fig. 26: Creating a task in the TwinCAT tree

In the case of less powerful devices (e.g. CX1000, CX9000 with E-bus), the CPU load may reach its limits. However, the performance can be improved again by means of the following considerations/measures:

- Is it possible to work with just one PLC task?
- Does the cycle time have to be 1 ms?
- If two tasks are used, can the PLC task time be shortened?
- Will several virtual PROFINET devices by required?

3   EL6631-0010

3.1   Product overview

3.1.1   PROFINET EL6631-0010 - Introduction
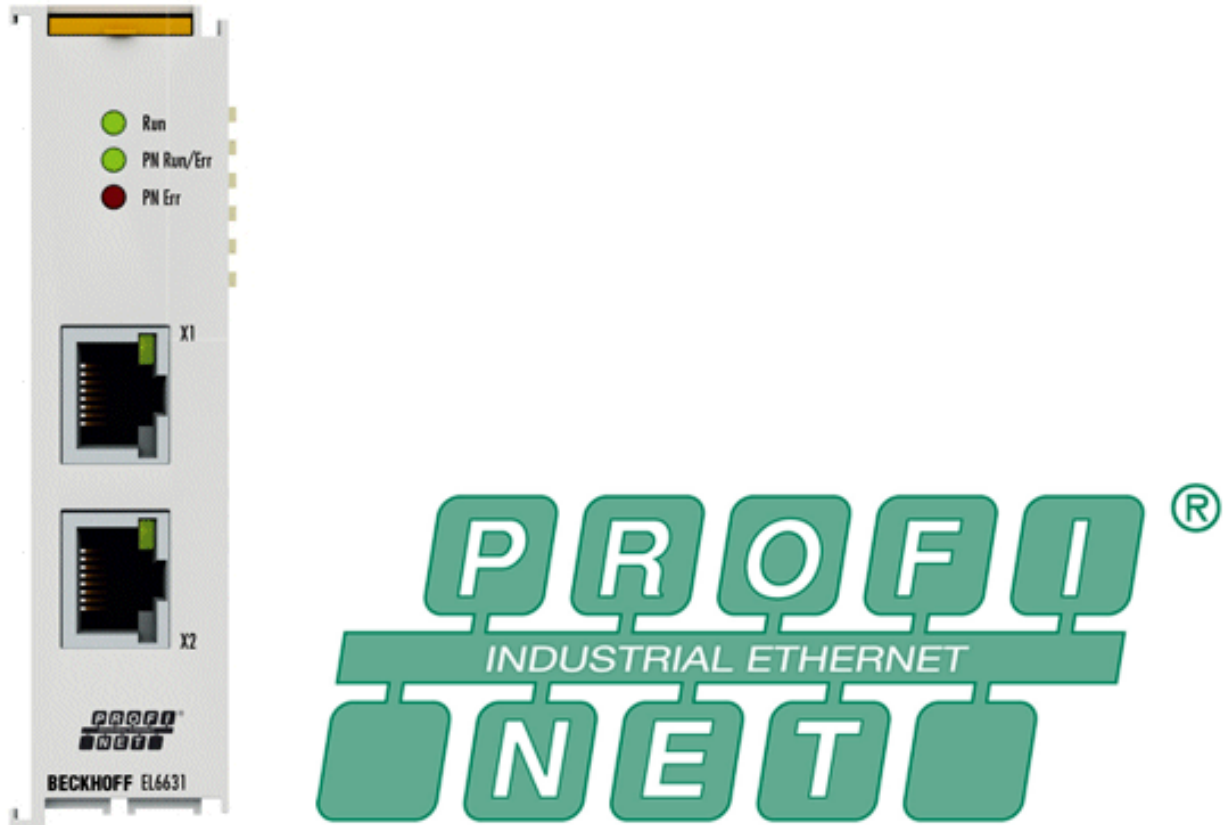
Fig. 27: EL6631

The EL6631-0010 PROFINET IO Device (Slave) Terminal enables simple exchange of data between EtherCAT and PROFINET IO. It is a device in the EtherCAT strand, which can consist of up to 65,535 devices. The EL6631-0010 has a 3-port switch. Two ports are fed to the outside on RJ-45 sockets. This allows the I/O stations to be structured as a line topology, as a result of which the wiring is simplified. The maximum distance between two devices is 100 m.

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

3.1.2   EL6631-0010 - Technical data

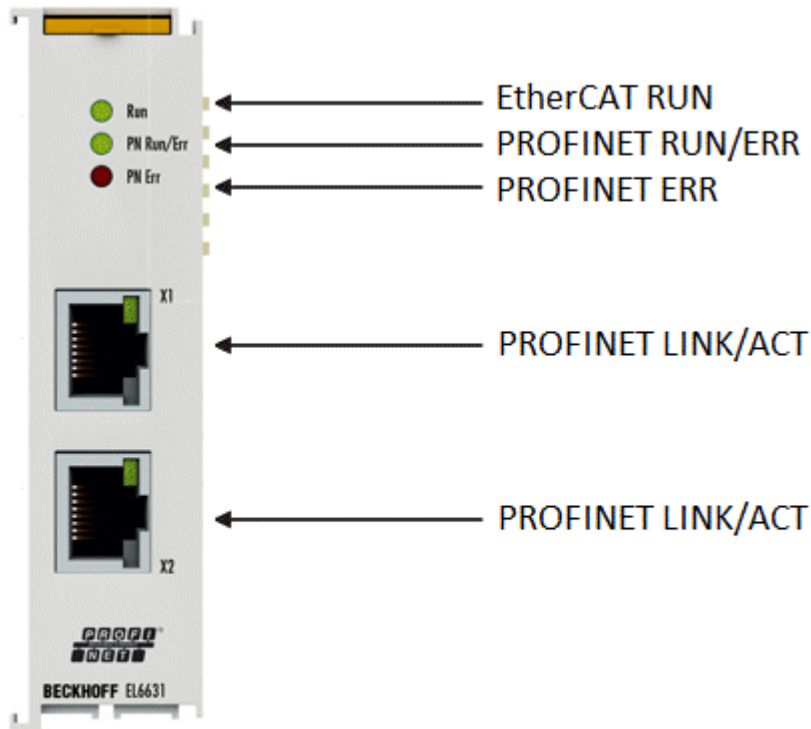| Technical data	EL6631-0010
Bus system	PROFINET RT Device
Number of Ethernet ports	2
Ethernet interface	10BASE-T/100BASE-TX Ethernet with 2 x RJ45
Cable length	up to 100 m twisted pair
Data transfer rate	10/100 Mbit/s, IEEE 802.3u Auto negotiation, half or full duplex at 10 and 100 Mbit/s possible, automatic settings
Diagnostics	Status LEDs
Power supply	via the E-bus
Current consumption via E-bus	typ. 400 mA
Electrical isolation	500 V (E-Bus/Ethernet)
Bit width in process image	variable (max. 2 KB for inputs and outputs)
Configuration	via the TwinCAT System Manager
Weight	approx. 75 g
Permissible ambient temperature range during operation	0 °C ... + 55 °C (aligned in horizontal installation position) 0 °C ... + 45 °C (all other installation positions, see Note [▶ 34])
Permissible ambient temperature range during storage	-25 °C ... + 85 °C
Permissible relative humidity	95 %, no condensation
Dimensions (W x H x D)	approx. 26 mm x 100 mm x 52 mm (width aligned: 23 mm)
Mounting [▶ 30]	on 35 mm mounting rail conforms to EN 60715
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
Protection class	IP20
Installation position	see notice [▶ 34]
Approval	CE ATEX [▶ 36] cULus [▶ 37]

3.1.3   EL6631-0010 - LEDs

Fig. 28: Tc_EL6631_LEDs

LEDs for EtherCAT diagnosis

LED		Display	Description
RUN	green	off	State of the EtherCAT State Machine: **INIT** = initialization of the terminal; **BOOTSTRAP** = function for terminal firmware updates
		flashing 200 ms	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different standard-settings set
		off (1 s) on (200 ms)	State of the EtherCAT State Machine: **SAFEOP** = verification of the sync manager channels and the distributed clocks. Outputs remain in safe state
		on	State of the EtherCAT State Machine: **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) on (200 ms)	off	EL6631-0010 does not have an IP address
flashing 200 ms	off	EL6631-0010 still has not received a PROFINET name
off	flashing 200 ms	Terminal starts

LED diagnosis PROFINET Err

Colors green	Colors red	Meaning
on	off	EL terminal is exchanging data
flashing 200 ms	off	EL terminal is exchanging data, but the provider status is stopped
off (1 s) on (200 ms)	off	EL terminal is exchanging data, but the modules are different
off	flashing 500 ms	No AR established, establishment of connection has not been initialized
flashing 500 ms	flashing 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.

3.2   Mounting and wiring

3.2.1   Instructions for ESD protection

| *NOTE* |
|---|
| **Destruction of the devices by electrostatic discharge possible!**
The devices contain components at risk from electrostatic discharge caused by improper handling.
• Please ensure you are electrostatically discharged and avoid touching the contacts of the device directly.
• Avoid contact with highly insulating materials (synthetic fibers, plastic film etc.).
• Surroundings (working place, packaging and personnel) should by grounded probably, when handling with the devices.
• Each assembly must be terminated at the right hand end with an <u>EL9011</u> or <u>EL9012</u> bus end cap, to ensure the protection class and ESD protection. |
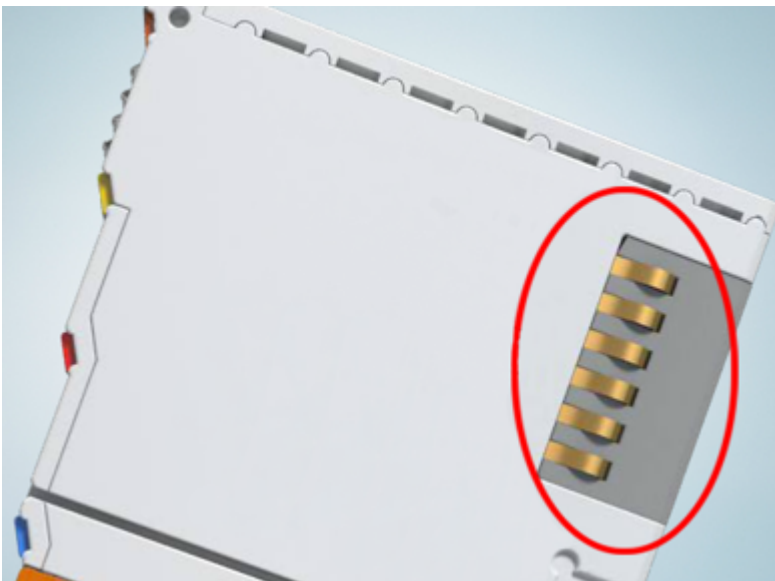
Fig. 29: Spring contacts of the Beckhoff I/O components

3.2.2   Recommended mounting rails

Terminal Modules und EtherCAT Modules of KMxxxx and EMxxxx series, same as the terminals of the EL66xx and EL67xx series can be snapped onto the following recommended mounting rails:

• DIN Rail TH 35-7.5 with 1 mm material thickness (according to EN 60715)
• DIN Rail TH 35-15 with 1,5 mm material thickness

● Pay attention to the material thickness of the DIN Rail

i Terminal Modules und EtherCAT Modules of KMxxxx and EMxxxx series, same as the terminals of the EL66xx and EL67xx series does not fit to the DIN Rail TH 35-15 with 2,2 to 2,5 mm material thickness (according to EN 60715)!

3.2.3   Mounting and demounting - terminals with traction lever unlocking

The terminal modules are fastened to the assembly surface with the aid of a 35 mm mounting rail (e.g. mounting rail TH 35-15).

i   **Fixing of mounting rails**

The locking mechanism of the terminals and couplers extends to the profile of the mounting rail. At the installation, the locking mechanism of the components must not come into conflict with the fixing bolts of the mounting rail. To mount the recommended mounting rails under the terminals and couplers, you should use flat mounting connections (e.g. countersunk screws or blind rivets).
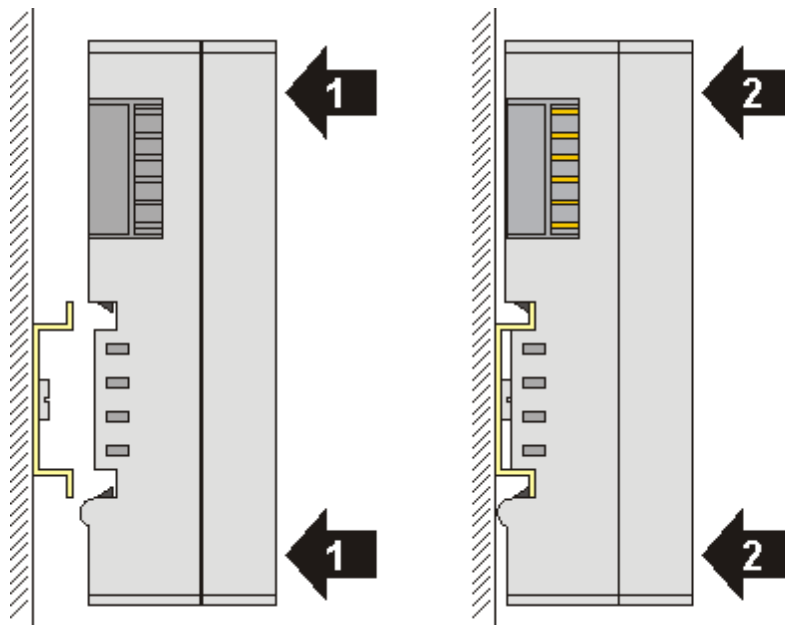
⚠ WARNING
Risk of electric shock and damage of device!
Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the Bus Terminals!

Mounting

- Fit the mounting rail to the planned assembly location.

and press (1) the terminal module against the mounting rail until it latches in place on the mounting rail (2).

- Attach the cables.

Demounting

- Remove all the cables. Thanks to the KM/EM connector, it is not necessary to remove all the cables separately for this, but for each KM/EM connector simply undo 2 screws so that you can pull them off (fixed wiring)!
- Lever the unlatching hook on the left-hand side of the terminal module upwards with a screwdriver (3). As you do this
  - an internal mechanism pulls the two latching lugs (3a) from the top hat rail back into the terminal module,
  - the unlatching hook moves forwards (3b) and engages
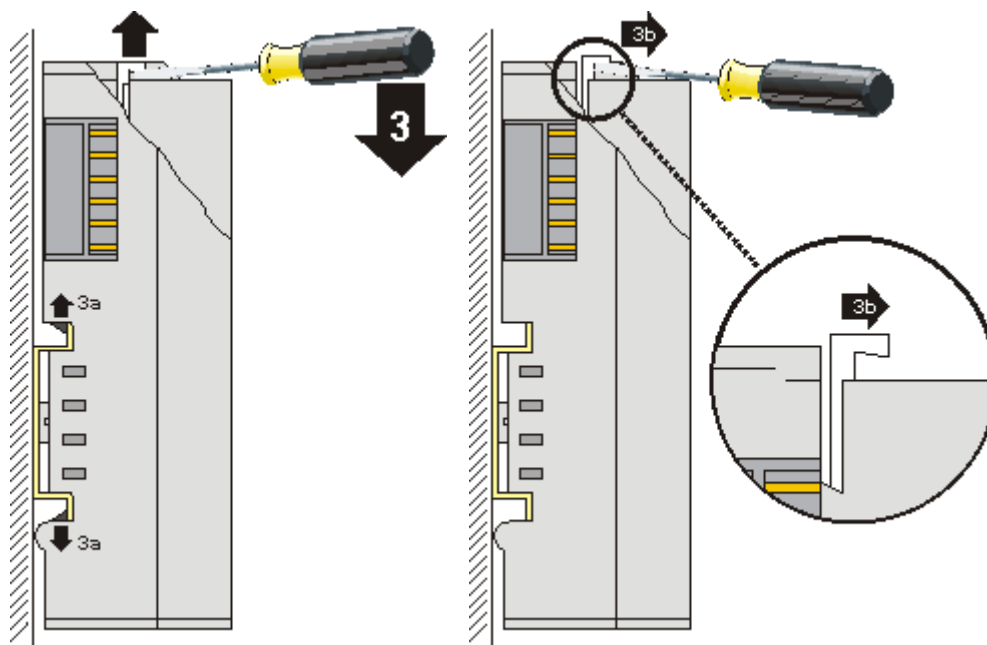
- In the case 32 and 64 channel terminal modules (KMxxx4 and KMxxx8 or EMxxx4 and EMxxx8) you now lever the second unlatching hook on the right-hand side of the terminal module upwards in the same way.
- Pull (4) the terminal module away from the mounting surface.
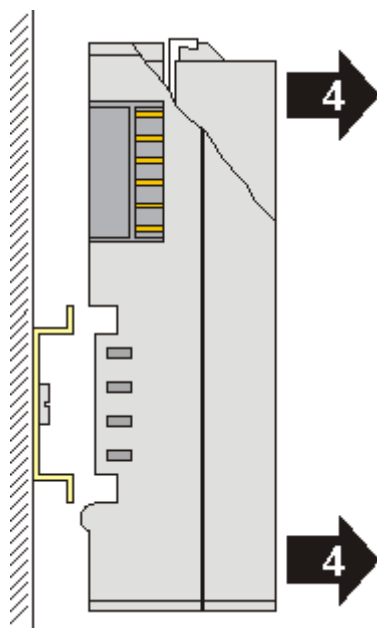
3.2.4   Mounting and demounting - terminals with front unlocking

The terminal modules are fastened to the assembly surface with the aid of a 35 mm mounting rail (e.g. mounting rail TH 35-15).
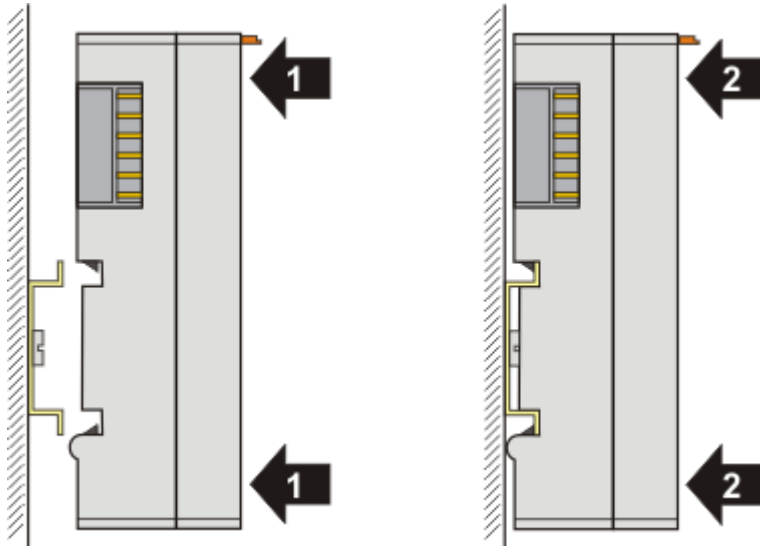
Fixing of mounting rails

The locking mechanism of the terminals and couplers extends to the profile of the mounting rail. At the installation, the locking mechanism of the components must not come into conflict with the fixing bolts of the mounting rail. To mount the recommended mounting rails under the terminals and couplers, you should use flat mounting connections (e.g. countersunk screws or blind rivets).

⚠ **WARNING**
Risk of electric shock and damage of device!
Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the Bus Terminals!

Mounting

- Fit the mounting rail to the planned assembly location.

and press (1) the terminal module against the mounting rail until it latches in place on the mounting rail (2).

- Attach the cables.

Demounting

- Remove all the cables.
- Lever the unlatching hook back with thumb and forefinger (3). An internal mechanism pulls the two latching lugs (3a) from the top hat rail back into the terminal module.
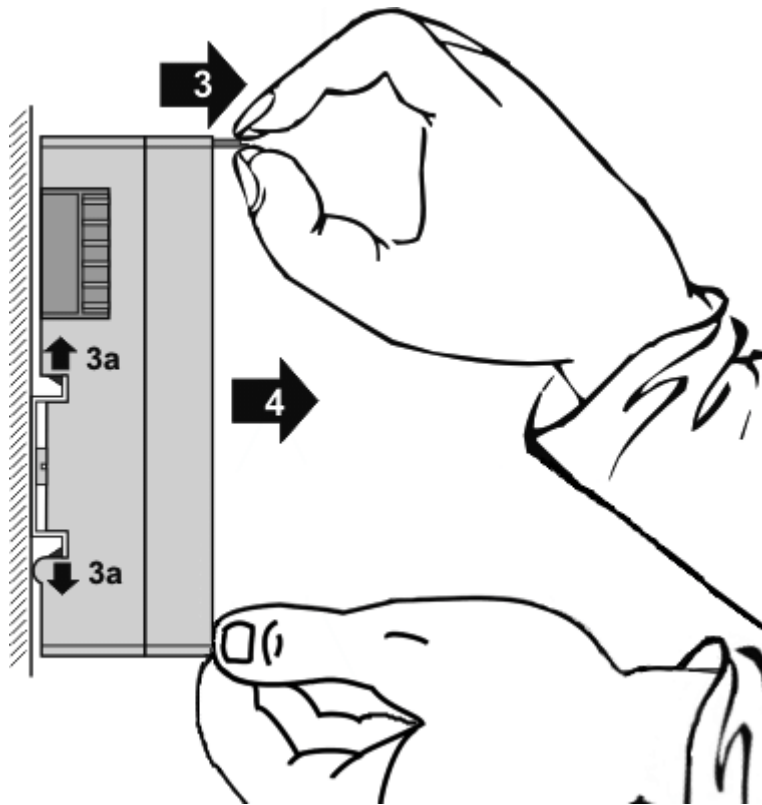
- Pull (4) the terminal module away from the mounting surface.
  Avoid canting of the module; you should stabilize the module with the other hand, if required.

3.2.5 Installation positions

NOTE
Constraints regarding installation position and operating temperature range
Please refer to the technical data for a terminal to ascertain whether any restrictions regarding the installation position and/or the operating temperature range have been specified. When installing high power dissipation terminals ensure that an adequate spacing is maintained between other components above and below the terminal in order to guarantee adequate ventilation!

Optimum installation position (standard)

The optimum installation position requires the mounting rail to be installed horizontally and the connection surfaces of the EL/KL terminals to face forward (see Fig. *"Recommended distances for standard installation position"*). The terminals are ventilated from below, which enables optimum cooling of the electronics through convection. "From below" is relative to the acceleration of gravity.
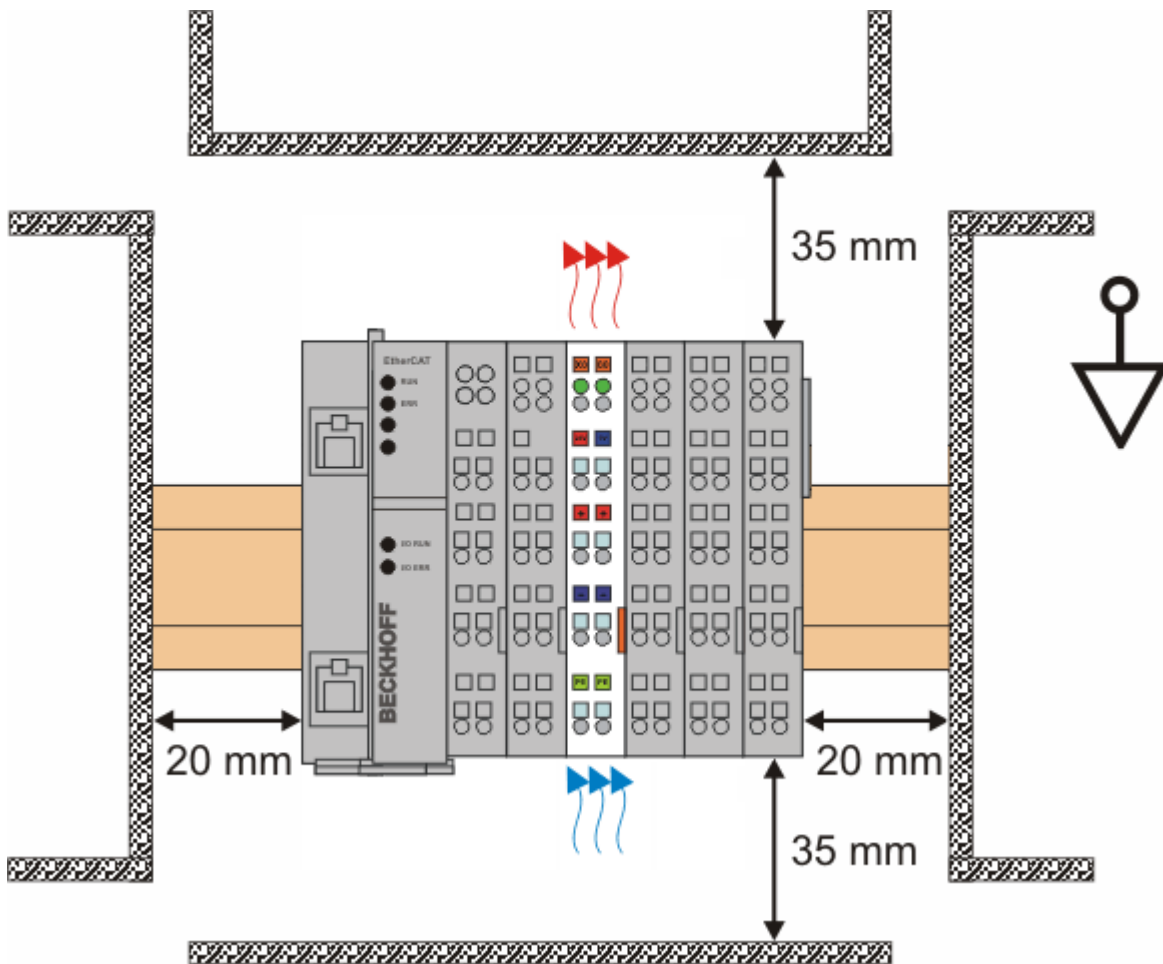
Fig. 30: Recommended distances for standard installation position

Compliance with the distances shown in Fig. *"Recommended distances for standard installation position"* is recommended.

Other installation positions

All other installation positions are characterized by different spatial arrangement of the mounting rail - see Fig *"Other installation positions".*

The minimum distances to ambient specified above also apply to these installation positions.
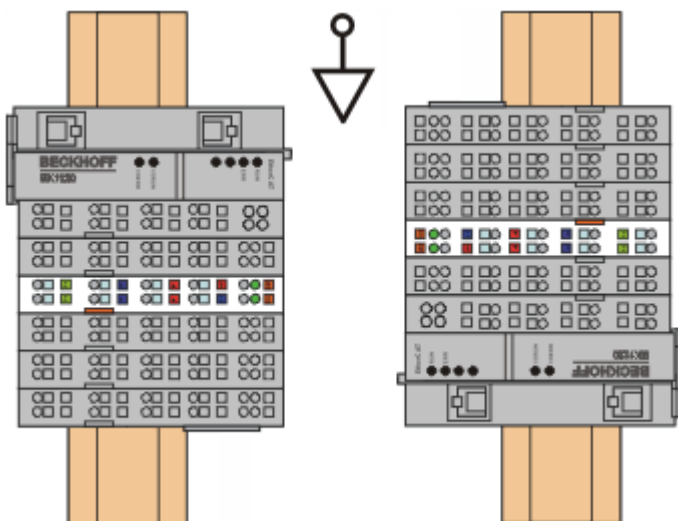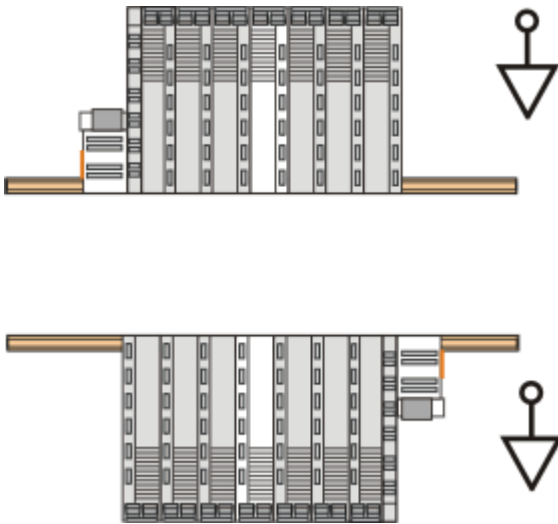
Fig. 31: Other installation positions

3.2.6 ATEX - Special conditions (standard temperature range)

⚠ WARNING
Observe the special conditions for the intended use of Beckhoff fieldbus components with standard temperature range in potentially explosive areas (directive 2014/34/EU)! • The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60079-15! The environmental conditions during use are thereby to be taken into account! • If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values! • Observe the permissible ambient temperature range of 0 to 55°C for the use of Beckhoff fieldbus components standard temperature range in potentially explosive areas! • Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages! • The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured! • The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured! • The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured! • Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

Standards

The fundamental health and safety requirements are fulfilled by compliance with the following standards:

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010

Marking

The Beckhoff fieldbus components with standard temperature range certified according to the ATEX directive for potentially explosive areas bear one of the following markings:

II 3G   KEMA 10ATEX0075 X Ex nA IIC T4 Gc   Ta: 0 … +55°C

or

II 3G   KEMA 10ATEX0075 X Ex nC IIC T4 Gc   Ta: 0 … +55°C

3.2.7   ATEX Documentation

> ℹ **Notes about operation of the Beckhoff terminal systems in potentially explosive areas (ATEX)**
>
> Pay also attention to the continuative documentation
>
> Notes about operation of the Beckhoff terminal systems in potentially explosive areas (ATEX)
>
> that is available in the download area of the Beckhoff homepage http:\\www.beckhoff.com!

3.2.8   UL notice

c(UL)us	**Application** Beckhoff EtherCAT modules are intended for use with Beckhoff's UL Listed EtherCAT System only.
c(UL)us	**Examination** For cULus examination, the Beckhoff I/O System has only been investigated for risk of fire and electrical shock (in accordance with UL508 and CSA C22.2 No. 142).
c(UL)us	**For devices with Ethernet connectors** Not for connection to telecommunication circuits.

Basic principles

UL certification according to UL508. Devices with this kind of certification are marked by this sign:

3.3   Commissioning

3.3.1   Technical data - PROFINET RT

Technical data	EL6631-0010
PROFINET Version	RT Class 1 ConformanceClassB
Number of device interfaces)[1]	2
Topology	variable
Quantity of user data	1 Kbyte In/Out
Cycle time	≥ 1 ms

)[1] see the chapter on virtual PROFINET devices

● **Switch functionality**

ℹ The internal switch is deactivated when switching to bootstrap mode and is not available during a firmware update.

3.3.2 PROFINET device (EL6631-0010) integration under TwinCAT 2.11

This description applies to the PROFINET DEVICE EL6631-0010.

1. Integration of the EL6631-0010 into the EtherCAT network

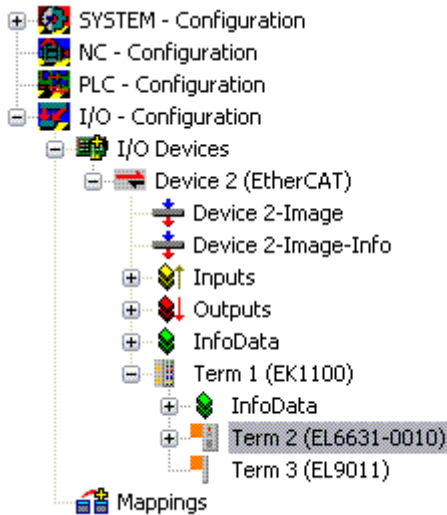Insert the EL6631-0010 into your TwinCAT project.

Fig. 32: Integration of the EL6631-0010 in the TwinCAT project

2. Integration of the PROFINET protocol

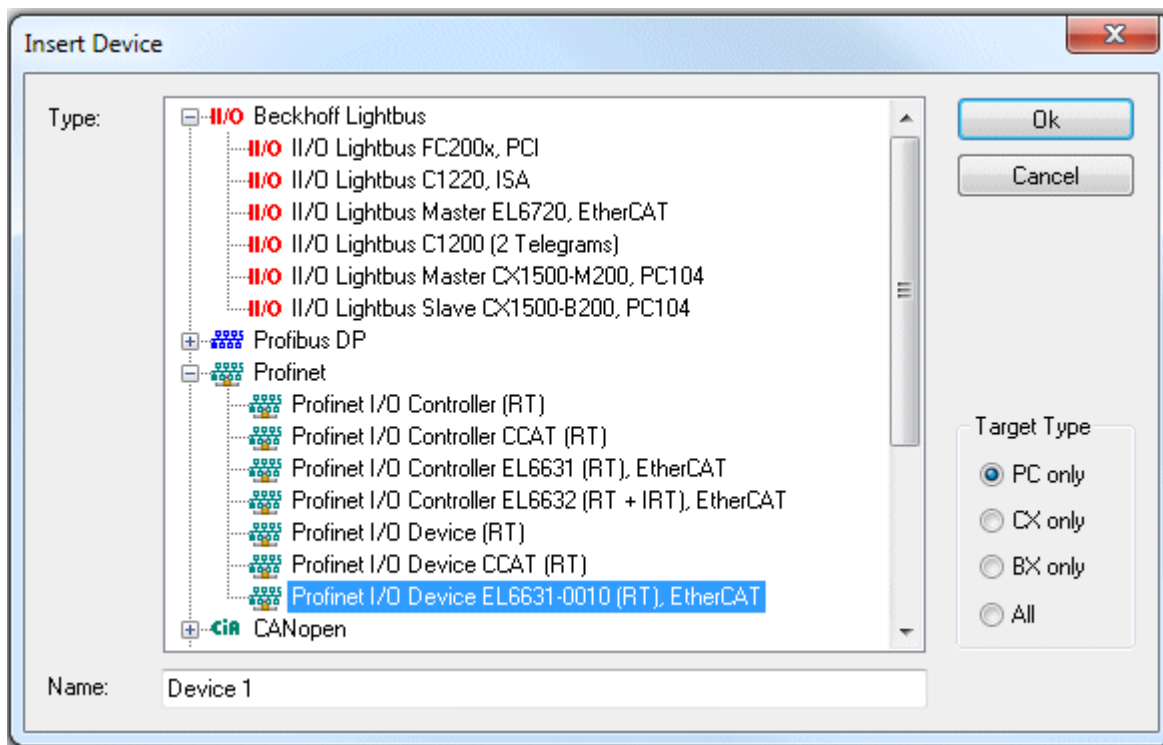A PROFINET device must first be added - EL6631-0010.

Fig. 33: "Insert Device" dialog

This interface must be assigned to an adaptor class. This is composed of the MAC and IP address of the network card. Use "Search" to look for the corresponding EtherCAT terminal. If there is only one terminal in the system, it will be configured automatically.

Fig. 34: "Adapter" tab

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, and has nothing to do with the PROFINET IP address that will be used later.

Promiscuous Mode:
Is not supported by the EL6631-0010

Virtual Device Names:
Is not supported by the EL6631-0010

Shared Adapter Interface:
Is not supported by the EL6631-0010

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!

If, e.g., a CX9000 is to be parameterized, the target system must first be accessed remotely; the corresponding Ethernet port can subsequently be selected.

The NetID used can be found under the "Protocol" tab. It is, e.g., necessary for an ADS access. Besides that, there is a possibility in the "Sync Task" tab to append a free-running task for PROFINET communication (this is recommended). This allows the RealTime communication to run independently of a PLC task. For performance reasons, however (e.g. in the case of a CX9000), the Sync Tasks can be dispensed with and standard mapping takes place, i.e. a PLC task, for example, must be running in order for PROFINET communication to be possible.

A PROFINET protocol is then added by means of the right mouse button. Precisely one TwinCAT device protocol can be appended!
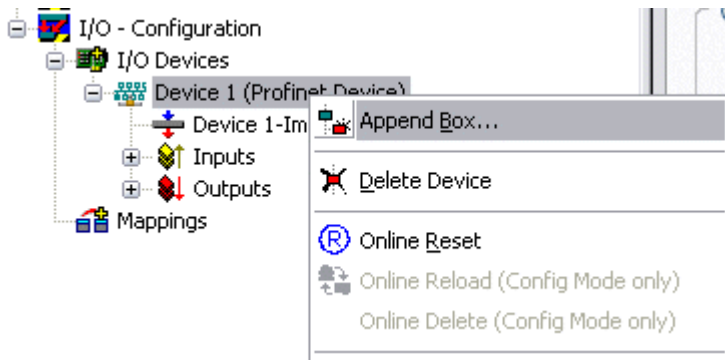
Fig. 35: Inserting a Profinet device protocol

A Box in the form of a GSDML is then integrated (right mouse button on "PROFINET Device").
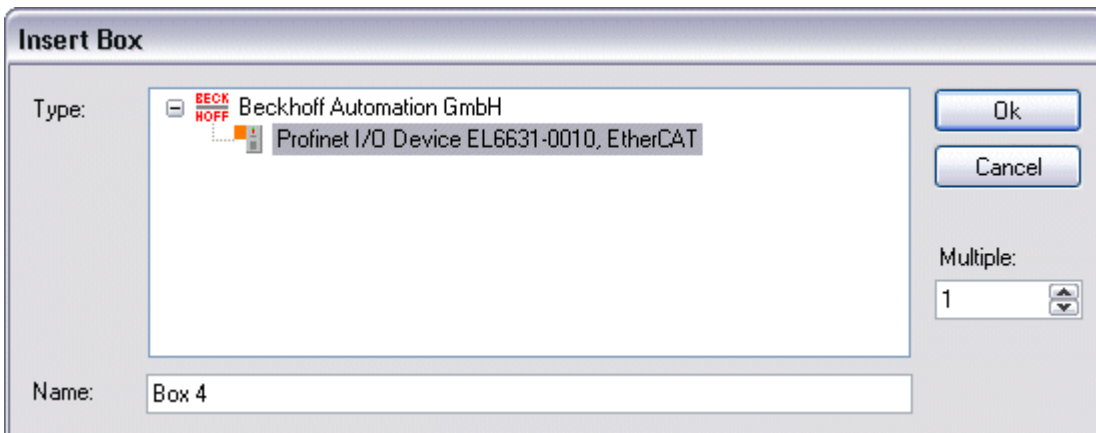

Fig. 36: "Insert Box" dialog

Now select the version to be used to integrate your EL6631-0010. The main difference between V2.0 and V2.2 lies in the support for sub-modules.
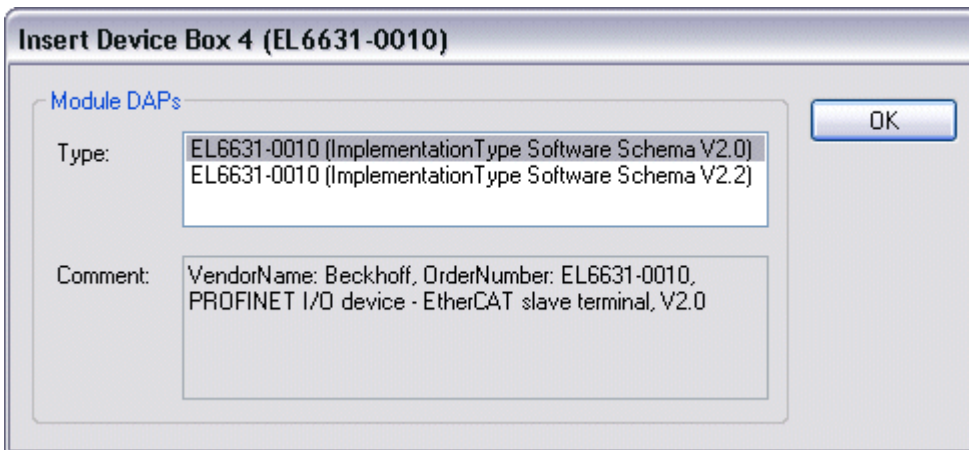

Fig. 37: EL6631-0010 selection with or without port diagnosis

In the first EL6631-0010 (according to V2.2) the ports, and therefore the port diagnosis, can be seen in the DAP. The following EL6631-0010 is integrated according to V2.0. No port diagnosis is possible here. The same version must be used in the master. Older PROFINET controllers cannot yet support Version 2.2, in which case you should use Version 2.0.
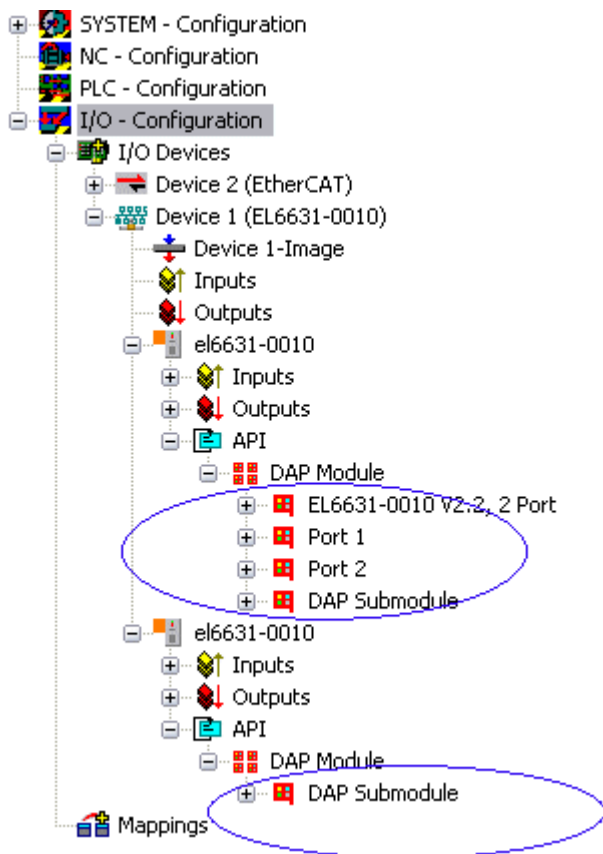
Fig. 38: Several PROFINET devices in the TwinCAT tree

Virtual PROFINET device

It is also possible in TwinCAT to configure a number of virtual PROFINET devices (a maximum of 7 in the PROFINET Device Supplement, and one additional device in the EL6631-0010). Each device is again to be viewed as its own adaptor, i.e. each virtual device is assigned a default MAC address from a reserved range. To avoid possible address conflicts, the MAC address of virtual devices can also be changed. The name which the device was given in the strand (or in the "General" tab) is also taken as a PROFINET station name following a restart. On the basis of different MAC addresses and names, a PROFINET controller can now distinguish several devices in the strand.

It is therefore possible to use 8 devices per Ethernet interface in the PROFINET Device Supplement, and therefore to communicate with 8 different PROFINET controllers. In the case of the EL6631-0010 it is 2 devices.

In PROFINET, different user profiles can be defined within a device (API - Application Process Instance). Precisely one API is permitted / supported for each device in TwinCAT.

In accordance with PROFINET, the DAP (Device Access Point) is always defined on slot 0. Device-specific data is defined in it. In accordance with PROFINET, several DAPs can be defined; however, in TwinCAT there is currently precisely one!

3. Process data

The process data can be inserted from slot 1. This takes place by integrating the desired module in the API. Each module (slot) has at present precisely one sub-module (sub-slot). The configuration of the modules and hence the creation of the process data image on the PROFINET side takes place by means of specifying data types / data widths (e.g. byte, word, dword, float 32).

The designation of the inputs and outputs always takes place in TwinCAT from the TwinCAT point of view, i.e. the inputs are from the PROFINET point of view (from the controller) the outputs! Therefore the data of the PROFINET I/O devices is represented as follows:
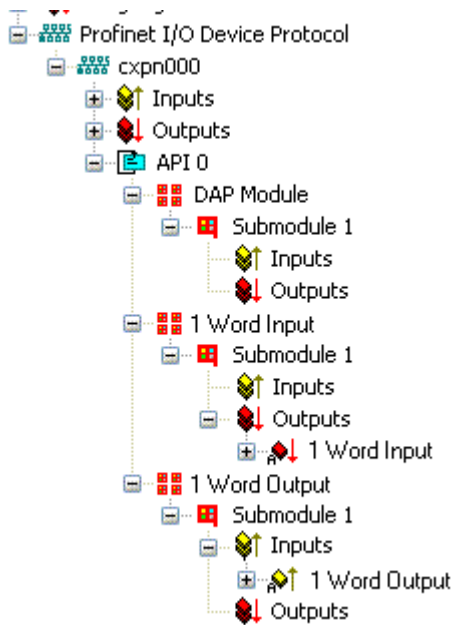
Fig. 39: Representation of the inputs/outputs of the PROFINET I/O device

This representation applies only to the TwinCAT PROFINET device, in the case of a TwinCAT PROFINET controller, the view will correlate again!

4. General settings

The FrameID can be changed on the "PROFINET Devices" tab. However, the default settings suffice for normal operation!
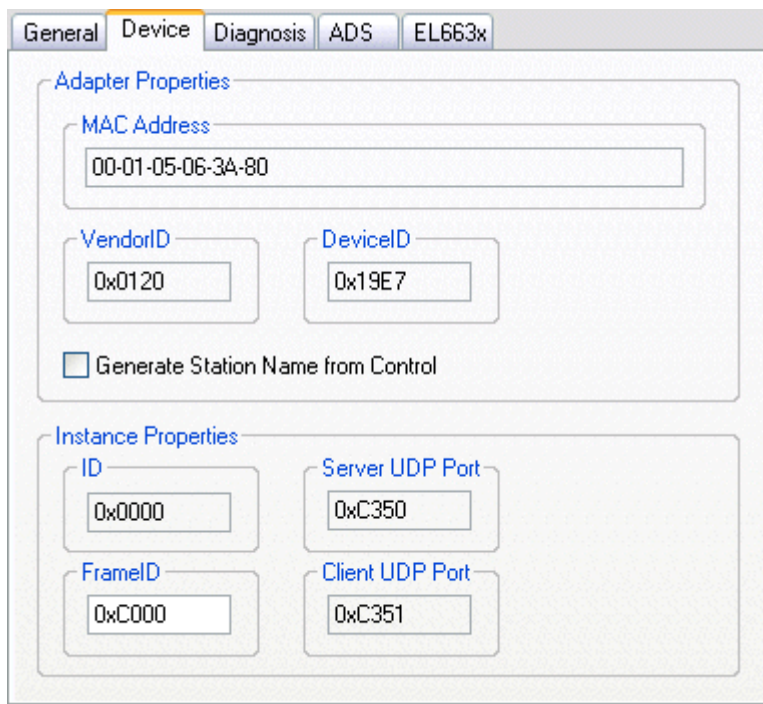

Fig. 40: "Device" tab

4.1 PROFINET name assignment

When the EL6631-0010 is shipped it does not have a PROFINET name. It returns an empty string when the devices are searched.

There are several ways to assign a name to an EL6631-0010:

1. Assignment through the PROFINET controller
2. Assignment through the EtherCAT Master (assignment through the controller is then no longer possible)
3. Assignment through a link to the PLC program (assignment through the controller is then no longer possible)
4. Automatic device startup through topology specification is supported
   You can obtain more information from your PROFINET controller

4.1.1 Assignment through the PROFINET controller

In this case, the PROFINET controller assigns the name of the device. You can learn about this from the manufacturer of your PROFINET controller.

4.1.2 Assignment by the EtherCAT Master

Tick the "get PN station name from ECAT" to activate this.
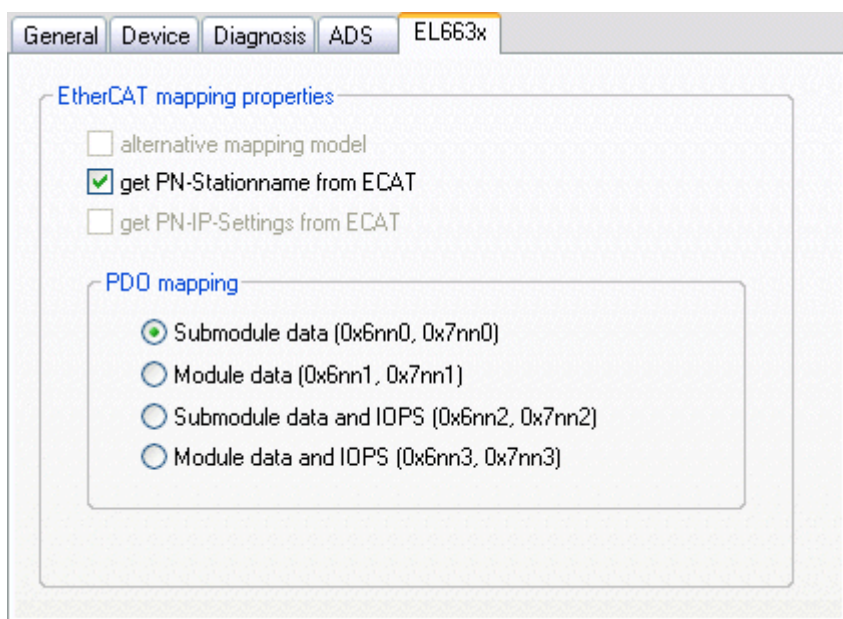

Fig. 41: "EL663x" tab

The name used in the System Manager tree is then used. In this case, for example, "el6631-test-name".
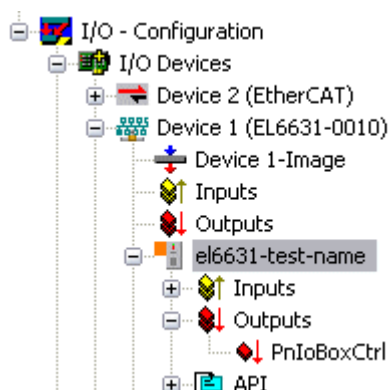

Fig. 42: TwinCAT tree: assigning a PROFINET station name

4.1.3 Assignment through a link to the PLC program

This is comparable with the DIP switches in the BK9103 and can be carried out via a PLC task. For activation, "Generate Station Name from Control" must be activated. For clarification, 000 will be appended to the previous tree name (default: "tcpniodevice").
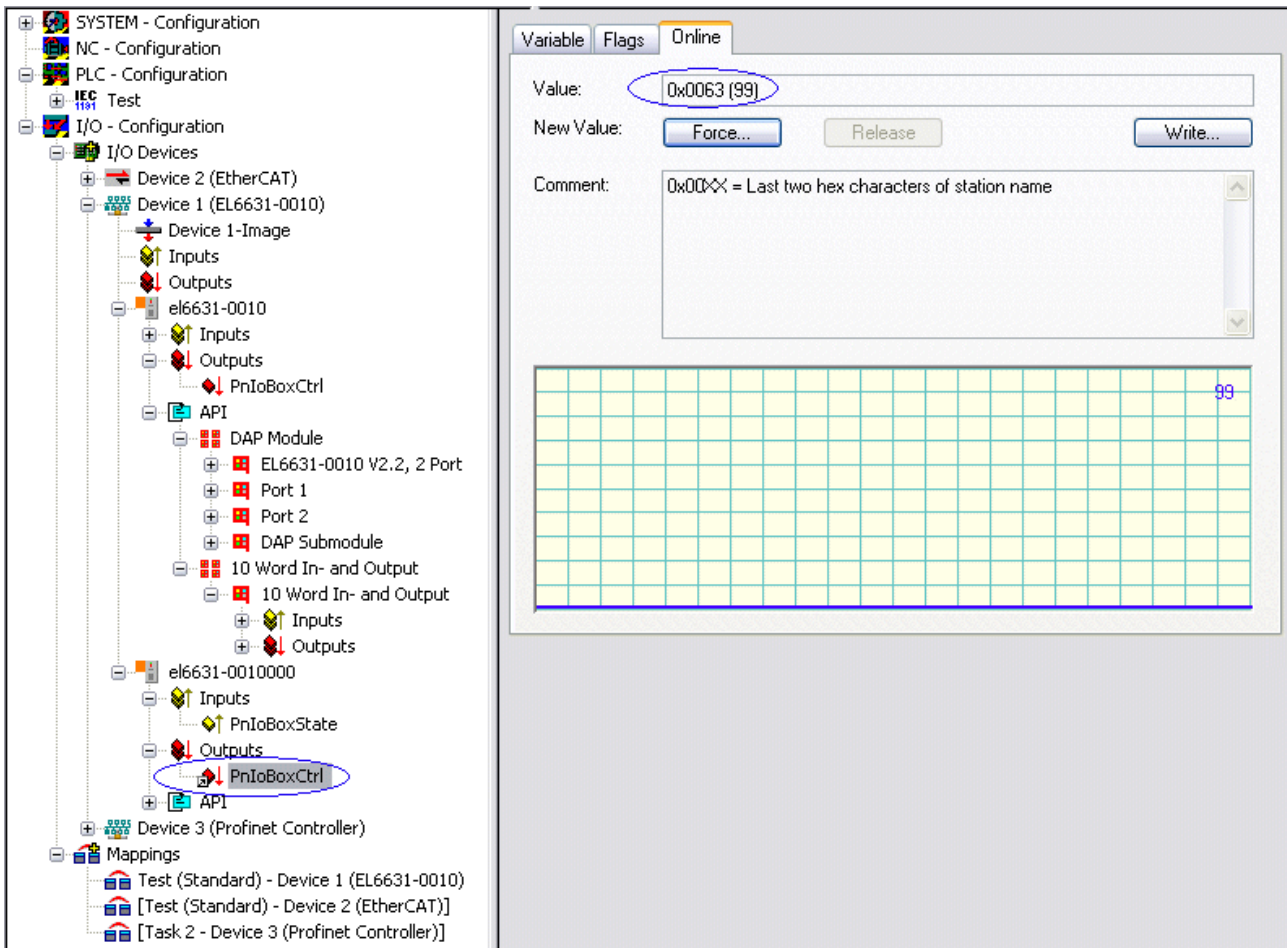
Fig. 43: Tree name in the TwinCAT tree

ⓘ Tree name

This tree name no longer corresponds to the PROFINET station name! The Ctrl WORD of the PROFINET protocol is used to help assign a name. This means that the number that is entered (range of values 0 - 255) is appended to the existing station name. In addition, the Ctrl WORD must be linked to a task. It is subsequently necessary to restart TwinCAT. If, e.g., the task now specifies a value of 11 for the linked Ctrl WORD, its previous station name will change from, for example, "tcpniodevice" to "tcpniodevice011". The current tree name remains "tcpniodevice000".

The current station names and the IP configuration used can be checked via the "PROFINET diagnosis" tab.

4.2 Diagnostics

The actual PROFINET name is displayed on the Diagnostics tab. The IP configuration is specified when the master boots up, and has the values that are assigned by the master. The "Module Difference" field is displayed if the specified configuration is not the same as the actual configuration. The field should be empty if there is no error.
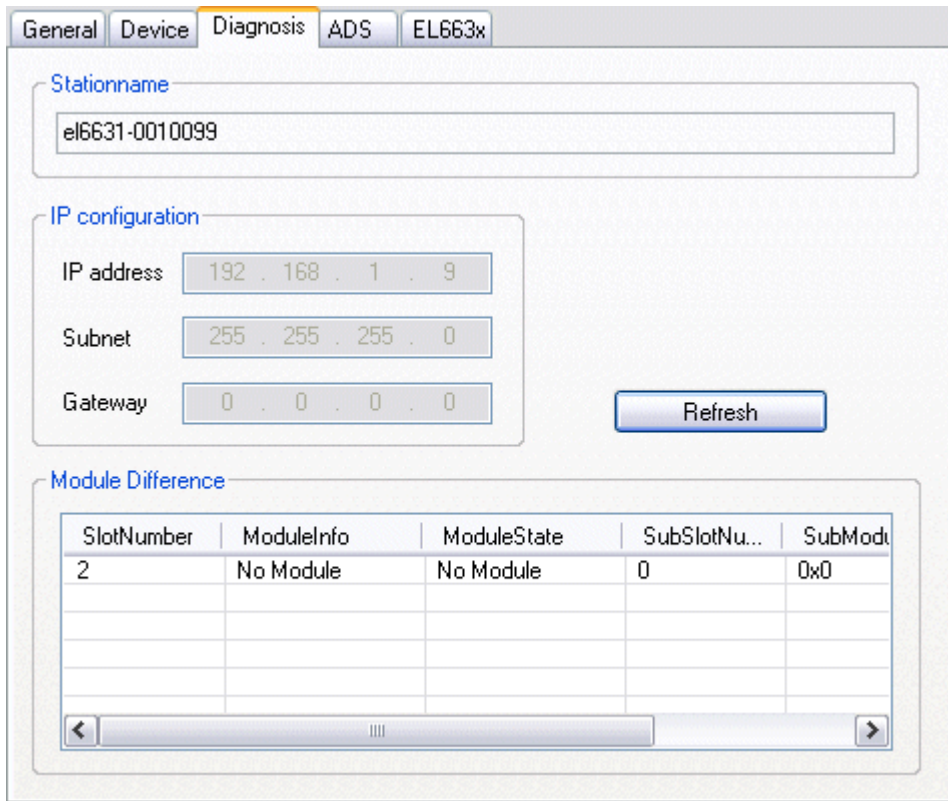
Fig. 44: "Diagnostics" tab

5. Cycle times

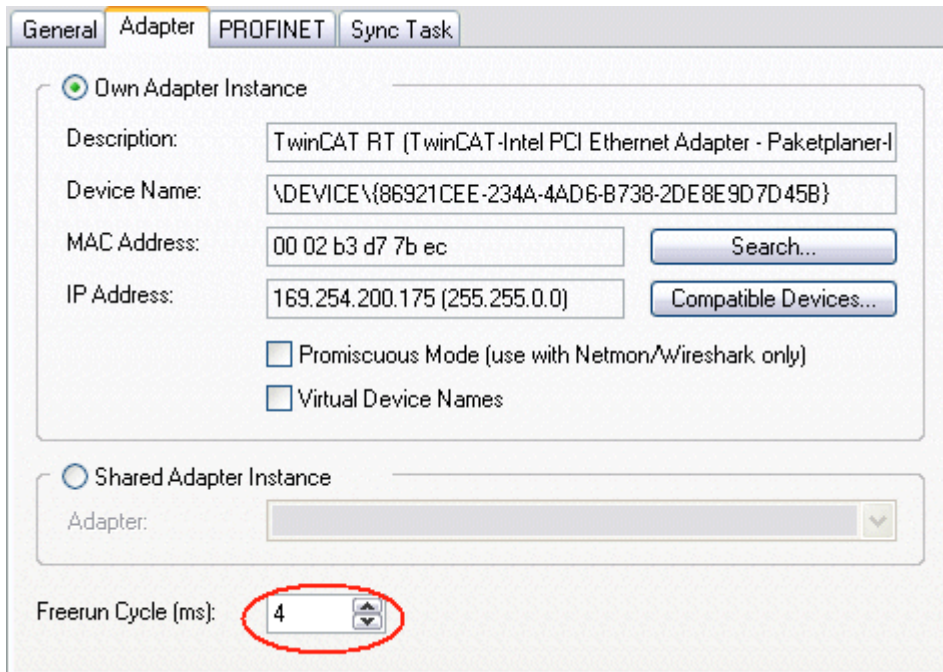The device can be operated in the minimum cycle time of 1 ms defined by PROFINET!

Fig. 45: "Adapter" tab, setting cycle time

If TwinCAT is to be operated in RUN mode, a task must be created. In the simplest case, this is the aforementioned SyncTask. The call time for the task must not be longer than the PROFINET cycle time! If a second task is created, e.g. for the PLC, this can also run more slowly.
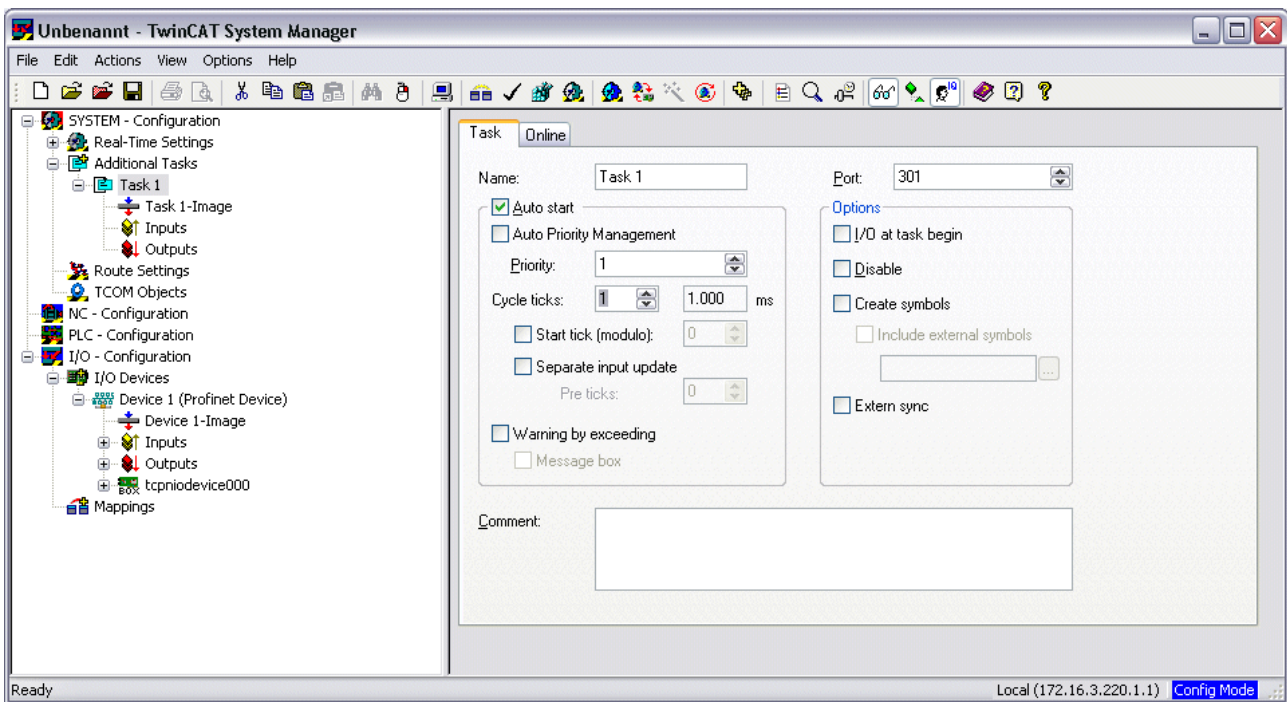
Fig. 46: Inserting a task

In the case of less powerful devices (e.g. CX1000, CX9000 with E-bus), the CPU load may reach its limits. However, the performance can be improved again by means of the following considerations/measures:

- Is it possible to work with just one PLC task?
- Does the cycle time have to be 1 ms?
- If two tasks are used, can the PLC task time be shortened?
- Will several virtual PROFINET devices by required?

6. PDO Mapping

The PDO mapping describes how the PROFINET data is to be represented in the EtherCAT. The "Submodule data" setting should be used if TwinCAT is used.
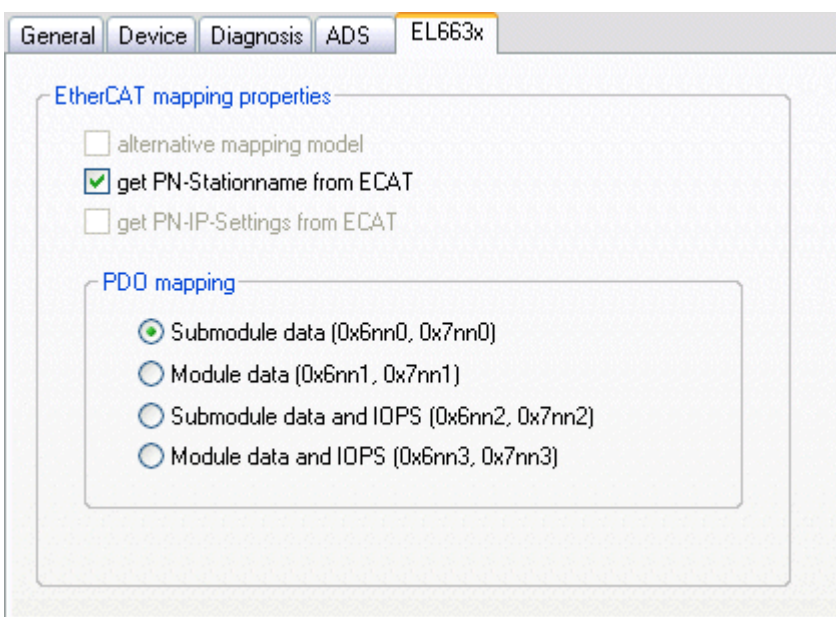

Fig. 47: "EL663x" tab

4 Profinet Features

4.1 Alarms

PROFINET alarm

PROFINET alarms can also be transmitted through the ADS interface.

An ADS WRITE block is to be used for this.

ADS WRITE	Comment
AMSNETID	AMS Net ID of the PROFINET device
PORT	0xFFFF (if a virtual PROFINET device is used, the port number is formed from 0x1000 + the device ID)
Index Group	High word - alarm type (see PROFINET specification), low word - 0xF807
Index Offset	High word - slot number, low word - sub slot number

No further data is transmitted.

PROFINET alarm (examples)

ALARMS	Comment
0x0000	reserved
0x0001	Diagnosis appears
0x0002	Process
0x0003	Pull
0x0004	Plug
0x0005	Status
...	further diagnostic messages (see PROFINET specification)

4.2 Record data

PROFINET record data

PROFINET record data can also be received directly by means of the PLC. To do this an "indication" is set in the PLC ((READINDICATION for reading and WRITEINDICATION for writing). The RECORD data structure has the following description:

```
WORD            RW;
   #define    PN_READ        0
   #define    PN_WRITE        1
WORD            NrOfAR;
DWORD           API;
WORD            Slot;
WORD            SubSlot;
PNIO_RECORD    RecordData;
```

Record Data

```
WORD        Index;
WORD        Length;            //zum Lesen auf 0
WORD        TransferSequence;
WORD        LengthOfAligned;
```

Table of PROFINET RECORD DATA represented on ADS

PROFINET	Length in bytes	Comment	ADS
-	String	AMS NET ID of the PROFINET device	AMSNETID
-	2	0xFFFF When the virtual interface is used, the port number is to be taken from the System Manager.	Port
Read/Write	2	0 - Read 1 - Write	Read - ReadIndication Write - WriteIndication
Number of AR	2	0x0000	-
API Application Process Identifier	4	0x00000000	-
Slot	2	Slot number 0x0000-0x7FFF	IndexOffset High Word
SubSlot	2	SubSlot number 0x0000-0x8FFF	IndexOffset Low Word
Record Data Index	2	0x0000-0x1FFF	IndexGroup Low Word
-	2	0x8000	IndexGroup High Word
Record Data Length	2	to "0" to read	LENGTH
Record Data Transfer Sequence	2	consecutive number	-
Record Data Length of Aligned	2	can be zero	-

Example for reading data; the PROFINET controller wants to read data from a Beckhoff device via the record data. In this example, all the RecIndex values are accepted, and will return the same data - 10 bytes that are located in the "Data" data array.

```
CASE i OF
0:  ADSREADIND(
        CLEAR:=FALSE ,
        VALID=> ,
        NETID=> ,
        PORT=> ,
        INVOKEID=> ,
        IDXGRP=> ,
        IDXOFFS=> ,
        LENGTH=> );
    IF ADSREADIND.VALID THEN
        IdxGroup:=ADSREADIND.IDXGRP;
        IdxOffset:=ADSREADIND.IDXOFFS ;
        i:=10;
        END_IF
10: ADSREADRES(
        NETID:=ADSREADIND.NETID,
        PORT:=ADSREADIND.PORT,
        INVOKEID:=ADSREADIND.INVOKEID ,
        RESULT:=0 ,
        LEN:=10 ,
        DATAADDR:=ADR(Data),
        RESPOND:=TRUE );
        i:=20;
    ADSREADIND(CLEAR:=TRUE);
20: i:=0;
    ADSREADRES(RESPOND:=FALSE);
END_CASE
```

4.3   Shared Device

The ProfiNET feature "Shared Device" is supported by the EL6631-0010, by the TwinCAT Supplement from version TC211 22.xx (= TC2 R3).
Shared Device features a function that allows two controllers to simultaneously access one device and, if applicable, the same process data image.

The difference to the use of the virtual device lies in the fact that, when creating a virtual device, a completely new device is generated (with its own name, MAC, IP, etc.). i.e. as a result the controller sees two separate devices and can establish two completely separate connections.

In the case of Shared Device, conversely, there is only ever one device. On the device side nothing needs to be engineered for this; the complete functionality is handled in the ProfiNET stack. A common example for this would be the use of the safety functionality. A "normal CONTROLLER" is assigned the "normal" inputs and outputs, while the F-CPU is assigned the safe I/Os. If the requested process image should overlap in both controllers, then the first AR (primary) that is established is given full read and write access to the submodule. The second AR (secondary) receives a corresponding ModuleDiff block at the submodules that are already occupied. If the primary AR is released, then the secondary AR receives an alarm now indicating that the requested submodule is free. The second controller now has the possibility, if it supports this, to re-parameterize the submodules and to adopt them into its AR.

4.4 Diagnostics

4.4.1 Status and Ctrl. flags

PROFINET status

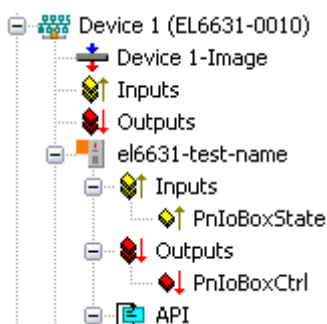The current status of the PROFINET communication can be monitored by means of PnIoBoxState.

Fig. 48: TwinCAT tree: PROFINET status via "PnIoBoxState"

PnIoBoxState	Comment	Meaning
0x0001 (Bit 0)	Device is in I/O exchange	PROFINET device is exchanging data
0x0002 (Bit 1)	Device is blinking	PROFINET device is being searched for by identification
0x0004 (Bit 2)	Provider State 0=STOP, 1=RUN	The PROFINET controller is stopped
0x0008 (Bit 3)	Problem indicator 0=OK, 1=Error	The PROFINET device has encountered problems

In the absence of an error, the value of PnIoBoxState is "5" - in other words, bits 0 and 2 are set.

PROFINET control

PnIoBoxCtrl can be used for assigning names; only the low byte is to be used for this. The high byte must be 0x00.

4.4.2 Port diagnostics

Port diagnostics can be used to identify the neighboring PROFINET devices. The device's own port can also be diagnosed.
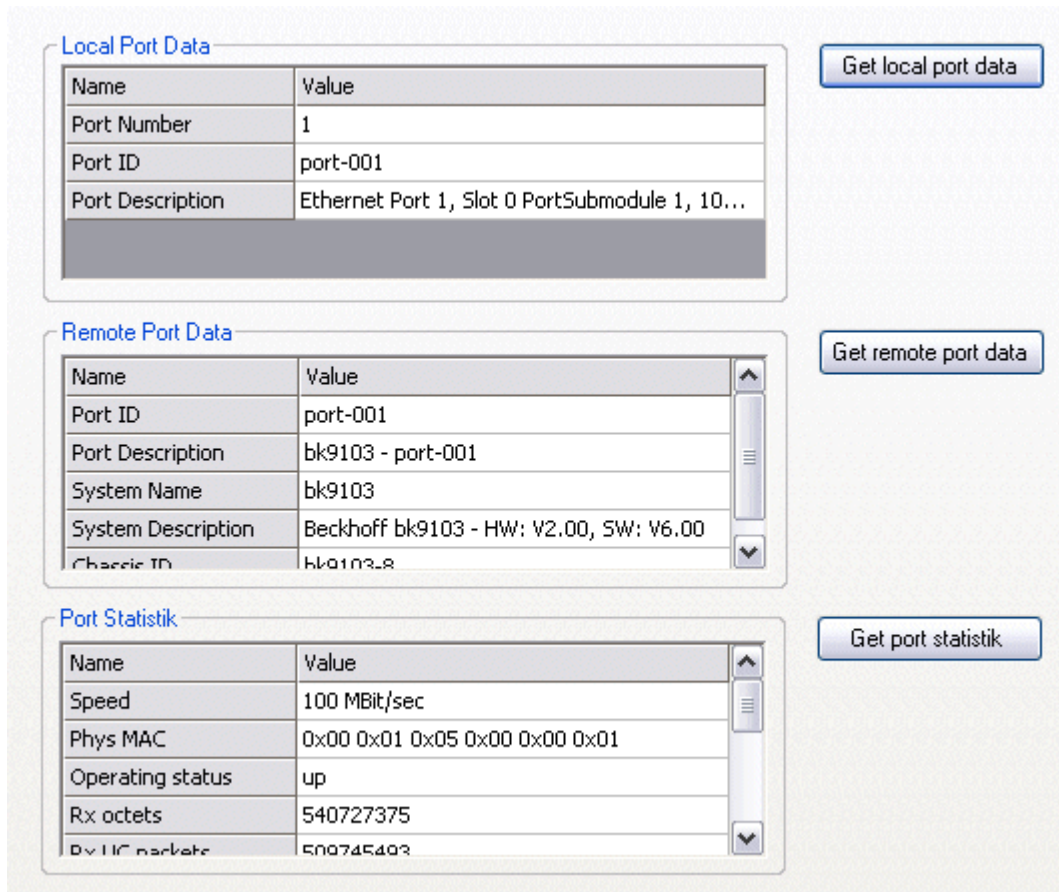
Fig. 49: Port diagnosis

4.4.3 Further diagnosis through the ADS interface

Additional diagnostic facilities are available through the ADS interface.

Read out the PROFINET name and the IP settings

An ADS READ block is to be used for this.

ADS WRITE	Comment
AMSNETID	AMS Net ID of the PROFINET device
PORT	0xFFFF (if a virtual PROFINET device is used, the port number is formed from 0x1000 + the device ID)
Index Group	High Word - 0x0000, Low Word - 0xF804
Index Offset	0x0000
Length	257

Data byte offset	Value	Comment
0..3	reserved	reserved
4..7	ARRAY4 of Byte	IP Address
8..11	ARRAY4 of Byte	Sub Net Mask
12..15	ARRAY4 of Byte	Default Gateway
16...x	STRING (max. length 240)	PROFINET Name

Reading out the module difference

An ADS READ block is to be used for this.

ADS WRITE	Comment
AMSNETID	AMS Net ID des PROFINET Devices
PORT	0xFFFF (if a virtual PROFINET device is used, the port number is formed from 0x1000 + the device ID)
Index Group	High Word - 0x0000, Low Word - 0xF826
Index Offset	0x0000
Length	20882

The modules are identical if the length null is returned. The length can differ, depending on the type (see the ModuleDiffBlock PROFINET specification)
Example of how the data can be constructed:

Data byte offset	Value	Comment
0..3	UINT32	API
4..5	UINT16	Number of Modules
6..7	UINT16	Slot
8..11	UINT32	Module Ident (see GSDML File)
12..13	UINT16	Module State (0-noModul, 1- WrongModule, 2- ProperModule, 3- Substitute)
14..15	UINT16	NumberOfSubModules
16..17	UINT16	SubSlot

5 TwinCAT library and programming

5.1 Functions

5.1.1 FUNCTION_BLOCK FB_Write_IuM_EL6631_0010

```
                  FB_WRITE_IUM_EL6631_0010
   ─bWrite : BOOL                       bBusy : BOOL─
   ─byState : BYTE                      bError : BOOL─
   ─iNumber : INT                      iErrorID : DINT─
   ─NETID : STRING(80)
   ─PORT : WORD
   ─st_IM_TagFunction : STRING(80)
   ─st_IM_TagLocation : STRING(80)
   ─st_IM_Date : STRING(80)
   ─st_IM_Descriptor : STRING(80)
   ─st_IM_Signature : STRING(80)
```
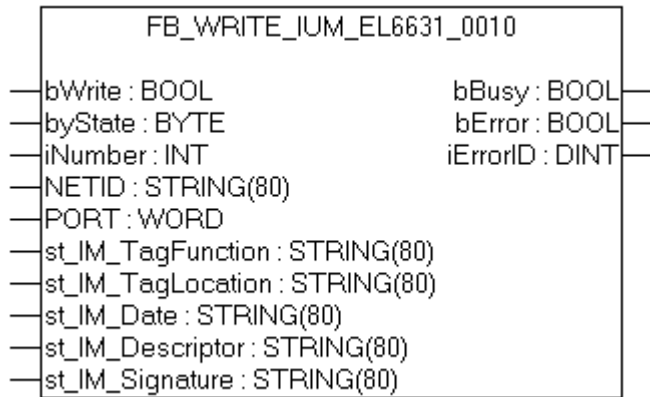
Fig. 50: FUNCTION_BLOCK FB_Write_IuM_EL6631_0010

This function block writes I&M1, I&M2, I&M3 and I&M4 (Identification & Maintenance) data according to ProfiNET specification over EtherCAT as a string to the ProfiNET device.

VAR_INPUT

```
VAR_INPUT
    bWrite              : BOOL;
    NETID               : STRING;  (* AMS NET ID EtherCAT Master *)
    PORT                : WORD;    (* EtherCAT Slave address *)
    byState             : BYTE;    (* Bit 0 -> I&M1 || Bit 1 -> I&M2 || Bit 2 -> I&M3 || Bit 3 -
> I&M4*)
    iNumber             : INT:=0;  (* "0"=EL6631-0010 or "1"=vitual EL6631-0010*)
    st_IM_TagFunction : STRING;  (* I&M1 byState.0=TRUE*)
    st_IM_TagLocation : STRING;  (* I&M1 byState.0=TRUE*)
    st_IM_Date          : STRING;  (* I&M2 byState.1=TRUE*)
    st_IM_Descriptor  : STRING;  (* I&M3 byState.2=TRUE*)
    st_IM_Signature   : STRING;  (* I&M4 byState.3=TRUE*)END_VAR
```

bWrite: A rising edge on this input activates the function block and the I&M data is written to the ProfiNET device.

NETID: AMS Net ID of the controller

PORT: Port via which the controller communicates with the device (port = Device ID + 1000_{hex})

byState: This byte can be used to select which I&M data is to be written.

iNumber: Two ProfiNET devices can be mapped with one terminal. *iNumber ("0" or "1")* is used to select the device for which the I&M data is to be written.

st_IM_TagFunction: Label for the function of the device is written to the device.

st_IM_Taglocation: Label for the installation site of the device is written to the device.

st_IM_Date: Date of the installation of the device is written to the device.

st_IM_Descriptor: Description of the manufacturer is written to the device.

st_IM_Signature: Description of the manufacturer is written to the device.

BECKHOFF

VAR_OUTPUT

```
VAR_OUTPUT
    bBusy    : BOOL;
    bError   : BOOL;
    iErrorID : DINT;
END_VAR
```

bBusy: When the function block is activated this output is set. It remains set until a feedback is received. While Busy = TRUE, no new command will be accepted at the inputs.

bError: In the event of an error during the command transfer, this output is set once the *bBusy* output has been reset.

iErrorID: Supplies an ADS error number when the output *bError* is set.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

5.1.2   FUNCTION_BLOCK FB_Read_IuM_EL6631_0010
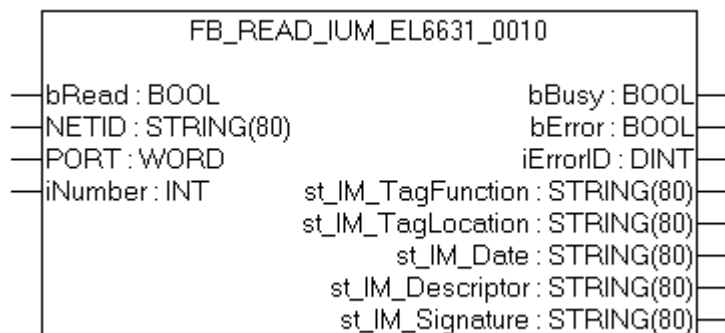
Fig. 51: FUNCTION_BLOCK FB_Read_IuM_EL6631_0010

This function block reads I&M1, I&M2, I&M3 and I&M4 (Identification & Maintenance) data over EtherCAT as a string from a ProfiNET device. The reading of the I&M0 data takes place in the case of a ProfiNET device with EtherCAT via CoE (CAN over EtherCAT).

VAR_INPUT

```
VAR_INPUT
    bRead   : BOOL;
    NETID   : STRING; (* AMS NET ID EtherCAT Master *)
    PORT    : WORD;   (* EtherCAT Slave address *)
    iNumber : INT;    (* "0"=EL6631-0010 or "1"=vitual EL6631-0010*)END_VAR
```

bRead: A rising edge on this input activates the function block and the I&M data is read from the ProfiNET device.

NETID: AMS Net ID of the controller

PORT: Port via which the controller communicates with the device (port = Device ID + 1000$_{hex}$)

iNumber: Two ProfiNET devices can be mapped with one terminal. *iNumber ("0" or "1")* is used to select the device for which the I&M data is to be read out.

VAR_OUTPUT

```
VAR_OUTPUT
    bBusy            : BOOL;
    bError           : BOOL;
    iErrorID         : DINT;
    st_IM_TagFunction : STRING; (* I&M1 *)
    st_IM_TagLocation : STRING; (* I&M1 *)
    st_IM_Date       : STRING; (* I&M2 *)
```

```
    st_IM_Descriptor  : STRING; (* I&M3 *)
    st_IM_Signature   : STRING; (* I&M4 *)
END_VAR
```

bBusy: When the function block is activated this output is set. It remains set until a feedback is received. While Busy = TRUE, no new command will be accepted at the inputs.

bError: In the event of an error during the command transfer, this output is set once the *bBusy* output has been reset.

iErrorID: Supplies an ADS error number when the output *bError* is set.

st_IM_TagFunction: label read out for the function of the device.

st_IM_Taglocation: label read out for the installation site of the device.

st_IM_Date: Supplies the date of installation of the device in the format.

st_IM_Descriptor: Returns the manufacturer description stored for the device.

st_IM_Signature: Returns the manufacturer signature stored for the device.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

6   Appendix

6.1   EtherCAT AL Status Codes

For detailed information please refer to the <u>EtherCAT system description</u>.

6.2   Firmware compatibility

Beckhoff EtherCAT devices are delivered with the latest available firmware version. Compatibility of firmware and hardware is mandatory; not every combination ensures compatibility. The overview below shows the hardware versions on which a firmware can be operated.

Note

- It is recommended to use the newest possible firmware for the respective hardware
- Beckhoff is not under any obligation to provide customers with free firmware updates for delivered products.

NOTE
Risk of damage to the device! Pay attention to the instructions for firmware updates on the <u>separate page [▶ 56]</u>. If a device is placed in BOOTSTRAP mode for a firmware update, it does not check when downloading whether the new firmware is suitable. This can result in damage to the device! Therefore, always make sure that the firmware is suitable for the hardware version!

EL6631-0010			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
03 - 07	01	EL6631-0010-0016	2011/05
	02	EL6631-0010-0017	2012/12
08-14*	02	EL6631-0010-0018	2013/05
	03		2014/05
	04		2014/10
	05		2014/12
	06		2015/05
	07		2015/10
02 – 14*	08		2016/11
	09		2017/01
	10		2017/06
	11		2017/11
	12		2018/04
	13		2018/07
	14*		2018/12

*) This is the current compatible firmware/hardware version at the time of the preparing this documentation. Check on the Beckhoff web page whether more up-to-date <u>documentation</u> is available.

6.3   Firmware Update EL/ES/EM/ELM/EPxxxx

This section describes the device update for Beckhoff EtherCAT slaves from the EL/ES, ELM, EM, EK and EP series. A firmware update should only be carried out after consultation with Beckhoff support.

Storage locations

An EtherCAT slave stores operating data in up to 3 locations:

- Depending on functionality and performance EtherCAT slaves have one or several local controllers for processing I/O data. The corresponding program is the so-called **firmware** in *.efw format.

- In some EtherCAT slaves the EtherCAT communication may also be integrated in these controllers. In this case the controller is usually a so-called **FPGA** chip with *.rbf firmware.

- In addition, each EtherCAT slave has a memory chip, a so-called **ESI-EEPROM**, for storing its own device description (ESI: EtherCAT Slave Information). On power-up this description is loaded and the EtherCAT communication is set up accordingly. The device description is available from the download area of the Beckhoff website at (https://www.beckhoff.de). All ESI files are accessible there as zip files.

Customers can access the data via the EtherCAT fieldbus and its communication mechanisms. Acyclic mailbox communication or register access to the ESC is used for updating or reading of these data.

The TwinCAT System Manager offers mechanisms for programming all 3 parts with new data, if the slave is set up for this purpose. Generally the slave does not check whether the new data are suitable, i.e. it may no longer be able to operate if the data are unsuitable.

Simplified update by bundle firmware

The update using so-called **bundle firmware** is more convenient: in this case the controller firmware and the ESI description are combined in a *.efw file; during the update both the firmware and the ESI are changed in the terminal. For this to happen it is necessary

- for the firmware to be in a packed format: recognizable by the file name, which also contains the revision number, e.g. ELxxxx-xxxx_REV0016_SW01.efw

- for password=1 to be entered in the download dialog. If password=0 (default setting) only the firmware update is carried out, without an ESI update.

- for the device to support this function. The function usually cannot be retrofitted; it is a component of many new developments from year of manufacture 2016.

Following the update, its success should be verified

- ESI/Revision: e.g. by means of an online scan in TwinCAT ConfigMode/FreeRun – this is a convenient way to determine the revision

- Firmware: e.g. by looking in the online CoE of the device

NOTE
Risk of damage to the device!
Note the following when downloading new device files
• Firmware downloads to an EtherCAT device must not be interrupted • Flawless EtherCAT communication must be ensured. CRC errors or LostFrames must be avoided. • The power supply must adequately dimensioned. The signal level must meet the specification.
In the event of malfunctions during the update process the EtherCAT device may become unusable and require re-commissioning by the manufacturer.

6.3.1   Device description ESI file/XML

NOTE
Attention regarding update of the ESI description/EEPROM
Some slaves have stored calibration and configuration data from the production in the EEPROM. These are irretrievably overwritten during an update.

The ESI device description is stored locally on the slave and loaded on start-up. Each device description has a unique identifier consisting of slave name (9 characters/digits) and a revision number (4 digits). Each slave configured in the System Manager shows its identifier in the EtherCAT tab:
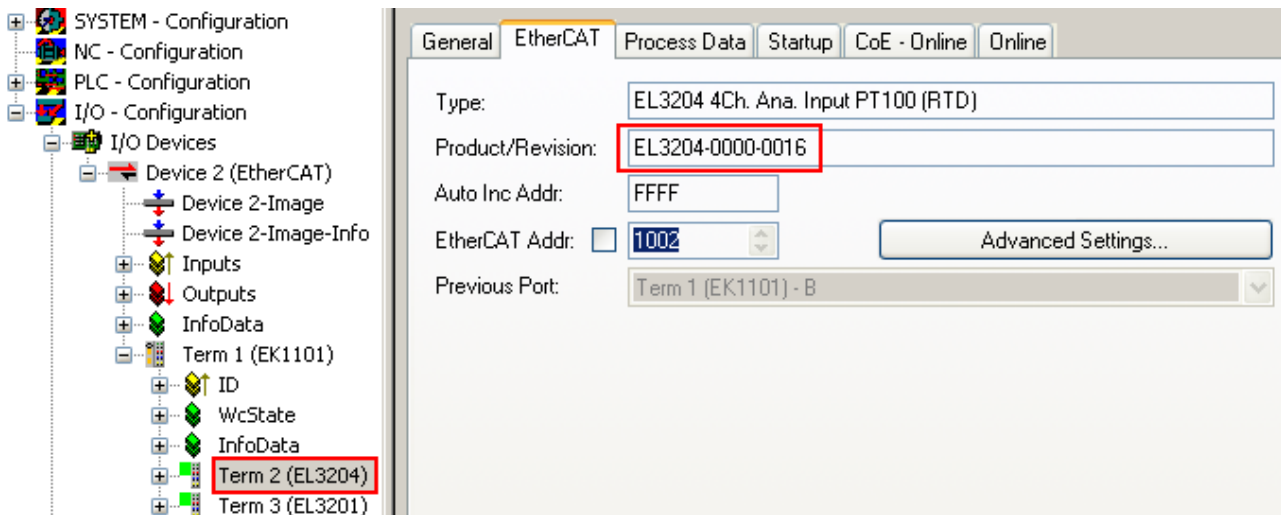
BECKHOFF

Fig. 52: Device identifier consisting of name EL3204-0000 and revision -0016

The configured identifier must be compatible with the actual device description used as hardware, i.e. the description which the slave has loaded on start-up (in this case EL3204). Normally the configured revision must be the same or lower than that actually present in the terminal network.

For further information on this, please refer to the EtherCAT system documentation.

Update of XML/ESI description

The device revision is closely linked to the firmware and hardware used. Incompatible combinations lead to malfunctions or even final shutdown of the device. Corresponding updates should only be carried out in consultation with Beckhoff support.

Display of ESI slave identifier

The simplest way to ascertain compliance of configured and actual device description is to scan the EtherCAT boxes in TwinCAT mode Config/FreeRun:
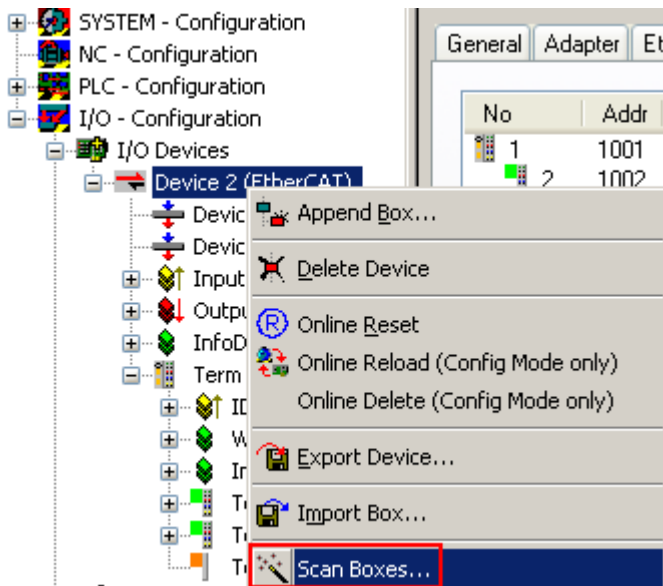
Fig. 53: Scan the subordinate field by right-clicking on the EtherCAT device

If the found field matches the configured field, the display shows

Fig. 54: Configuration is identical

otherwise a change dialog appears for entering the actual data in the configuration.
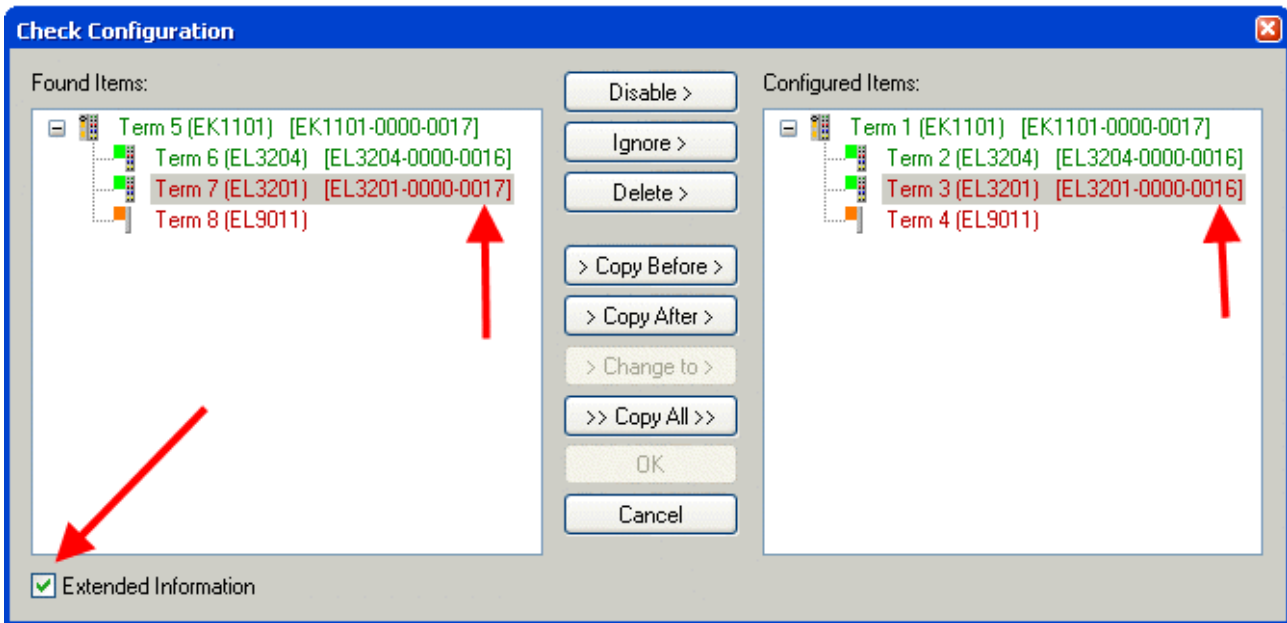
Fig. 55: Change dialog

In this example in Fig. *Change dialog*, an EL3201-0000-**0017** was found, while an EL3201-0000-**0016** was configured. In this case the configuration can be adapted with the *Copy Before* button. The *Extended Information* checkbox must be set in order to display the revision.

Changing the ESI slave identifier

The ESI/EEPROM identifier can be updated as follows under TwinCAT:

- Trouble-free EtherCAT communication must be established with the slave.
- The state of the slave is irrelevant.
- Right-clicking on the slave in the online display opens the *EEPROM Update* dialog, Fig. *EEPROM Update*

BECKHOFF
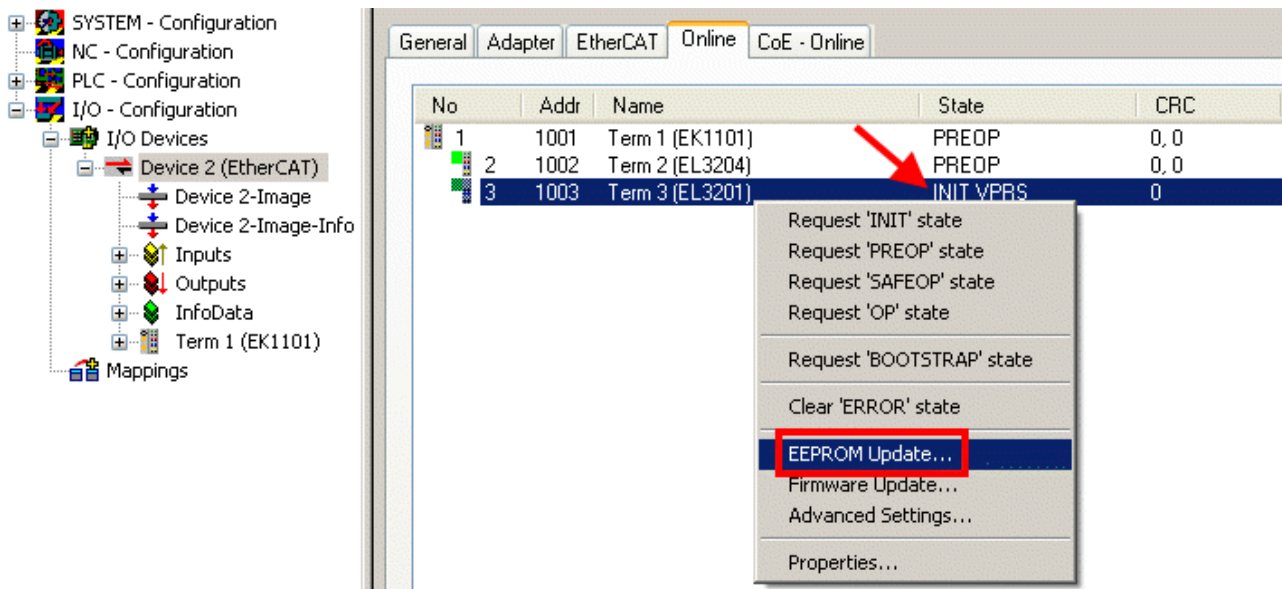
Fig. 56: EEPROM Update

The new ESI description is selected in the following dialog, see Fig. *Selecting the new ESI.* The checkbox *Show Hidden Devices* also displays older, normally hidden versions of a slave.
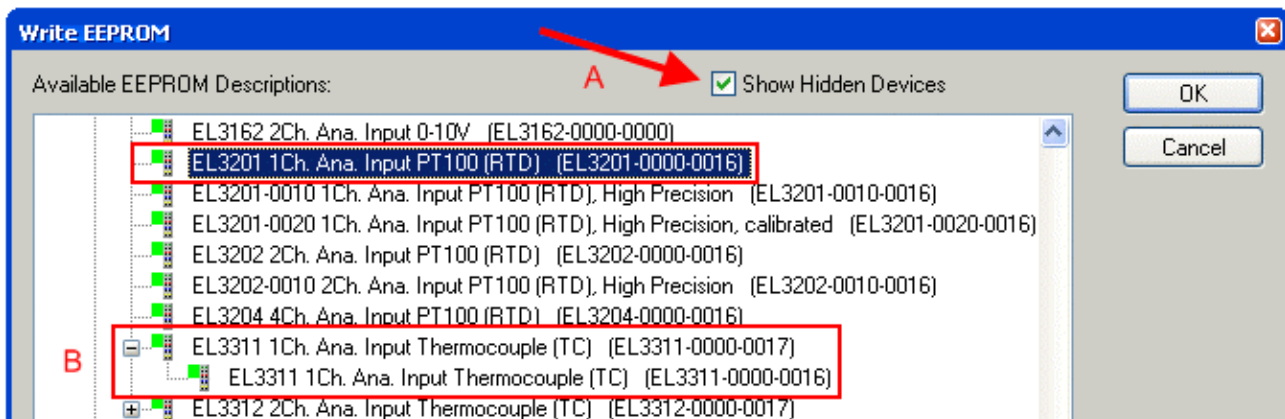
Fig. 57: Selecting the new ESI

A progress bar in the System Manager shows the progress. Data are first written, then verified.

> ● **The change only takes effect after a restart.**
>
> ℹ Most EtherCAT devices read a modified ESI description immediately or after startup from the INIT. Some communication settings such as distributed clocks are only read during power-on. The Ether-CAT slave therefore has to be switched off briefly in order for the change to take effect.

6.3.2 Firmware explanation

Determining the firmware version

Determining the version on laser inscription

Beckhoff EtherCAT slaves feature serial numbers applied by laser. The serial number has the following structure: **KK YY FF HH**

KK - week of production (CW, calendar week)
YY - year of production
FF - firmware version
HH - hardware version

Example with ser. no.: 12 10 03 02:

12 - week of production 12
10 - year of production 2010
03 - firmware version 03
02 - hardware version 02

Determining the version via the System Manager

The TwinCAT System Manager shows the version of the controller firmware if the master can access the slave online. Click on the E-Bus Terminal whose controller firmware you want to check (in the example terminal 2 (EL3204)) and select the tab *CoE Online* (CAN over EtherCAT).

> ℹ **CoE Online and Offline CoE**
>
> Two CoE directories are available:
> • **online**: This is offered in the EtherCAT slave by the controller, if the EtherCAT slave supports this. This CoE directory can only be displayed if a slave is connected and operational.
> • **offline**: The EtherCAT Slave Information ESI/XML may contain the default content of the CoE. This CoE directory can only be displayed if it is included in the ESI (e.g. "Beckhoff EL5xxx.xml").
>
> The Advanced button must be used for switching between the two views.

In Fig. *Display of EL3204 firmware version* the firmware version of the selected EL3204 is shown as 03 in CoE entry 0x100A.
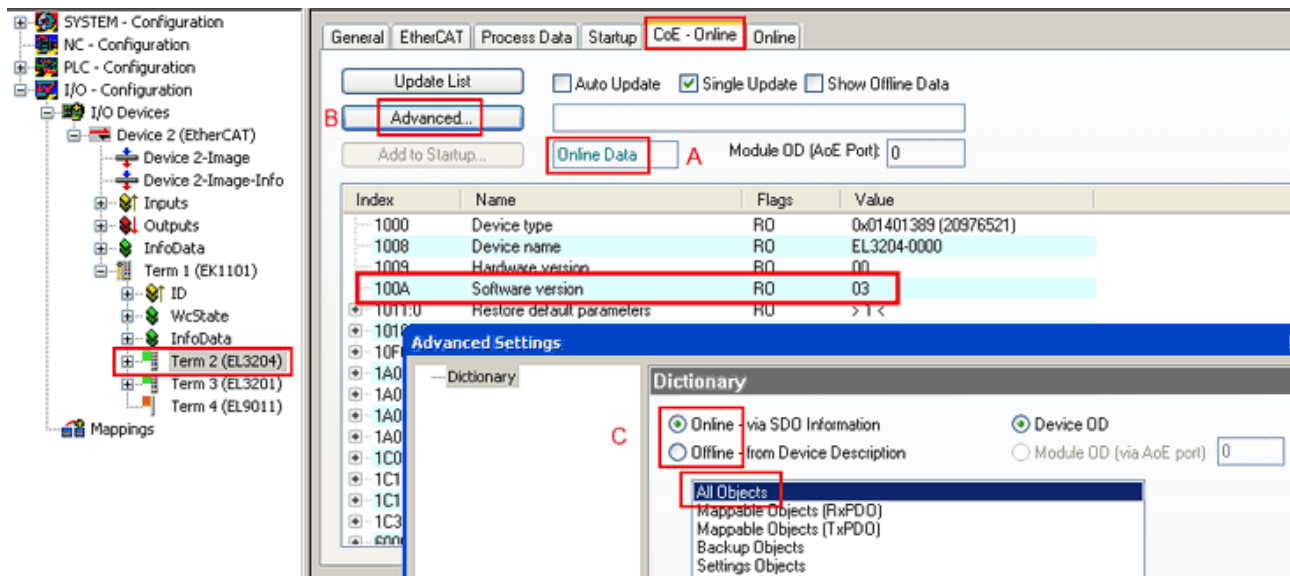
Fig. 58: Display of EL3204 firmware version

In (A) TwinCAT 2.11 shows that the Online CoE directory is currently displayed. If this is not the case, the Online directory can be loaded via the *Online* option in Advanced Settings (B) and double-clicking on *AllObjects*.

6.3.3   Updating controller firmware *.efw

> ℹ **CoE directory**
>
> The Online CoE directory is managed by the controller and stored in a dedicated EEPROM, which is generally not changed during a firmware update.

Switch to the *Online* tab to update the controller firmware of a slave, see Fig. *Firmware Update.*
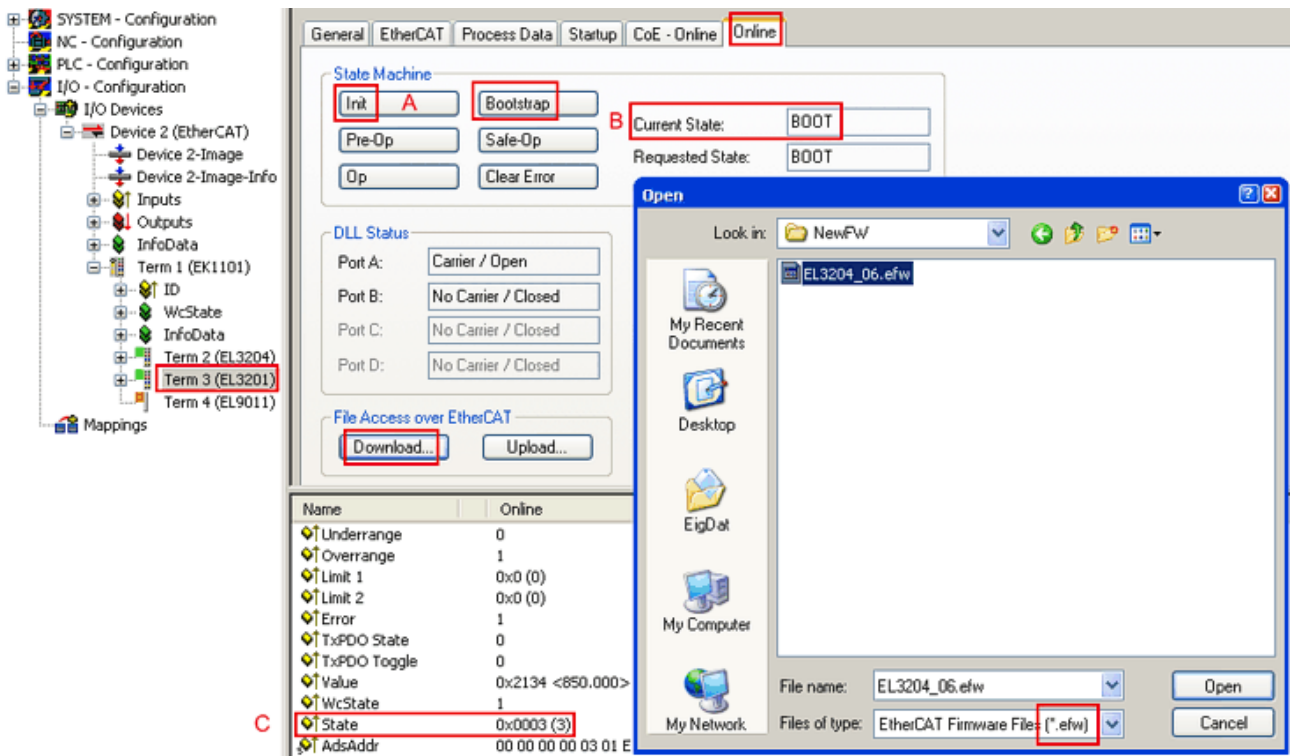
Fig. 59: Firmware Update

Proceed as follows, unless instructed otherwise by Beckhoff support. Valid for TwinCAT 2 and 3 as EtherCAT master.

- Switch TwinCAT system to ConfigMode/FreeRun with cycle time >= 1 ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.
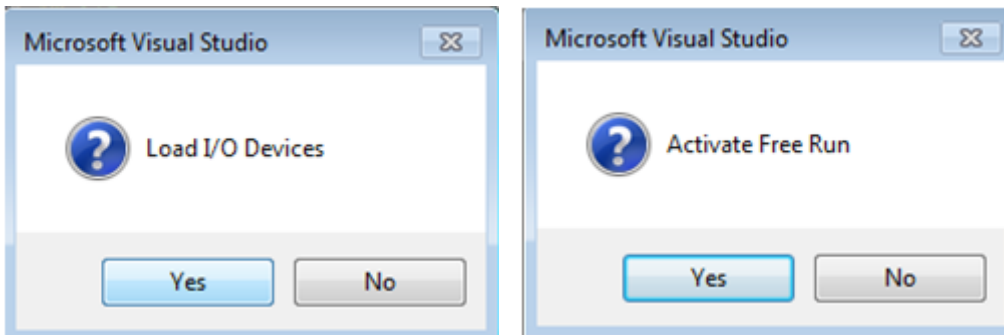

- Switch EtherCAT Master to PreOP
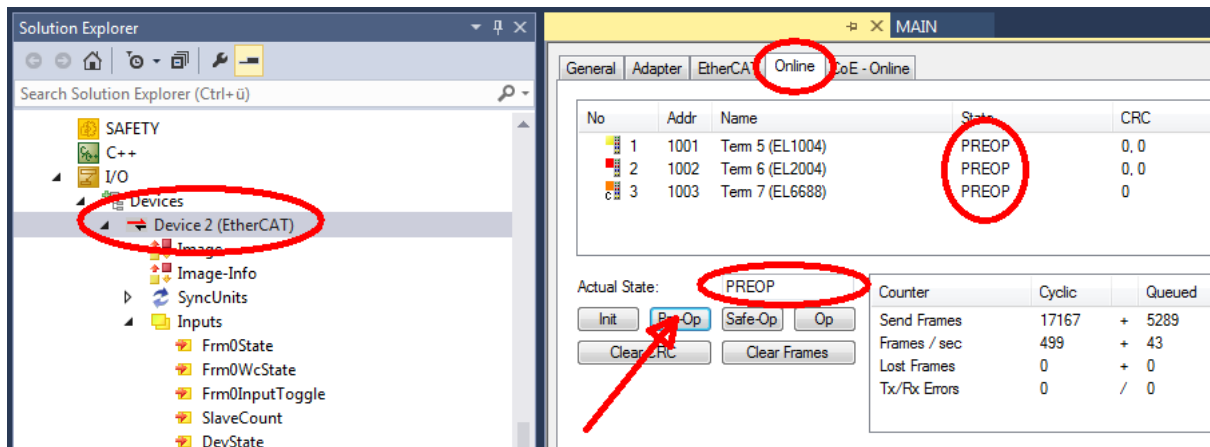
- Switch slave to INIT (A)
- Switch slave to BOOTSTRAP

- Check the current status (B, C)
- Download the new *efw file (wait until it ends). A pass word will not be neccessary usually.
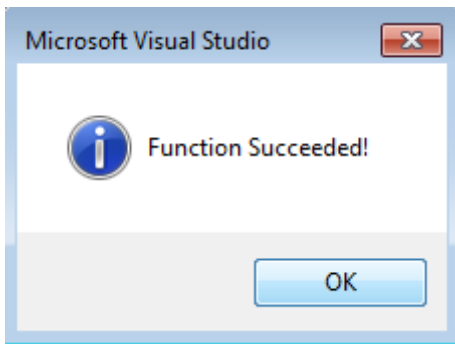
- After the download switch to INIT, then PreOP
- Switch off the slave briefly (don't pull under voltage!)
- Check within CoE 0x100A, if the FW status was correctly overtaken.

6.3.4   FPGA firmware *.rbf

If an FPGA chip deals with the EtherCAT communication an update may be accomplished via an *.rbf file.

- Controller firmware for processing I/O signals
- FPGA firmware for EtherCAT communication (only for terminals with FPGA)

The firmware version number included in the terminal serial number contains both firmware components. If one of these firmware components is modified this version number is updated.

Determining the version via the System Manager

The TwinCAT System Manager indicates the FPGA firmware version. Click on the Ethernet card of your EtherCAT strand (Device 2 in the example) and select the *Online* tab.

The *Reg:0002* column indicates the firmware version of the individual EtherCAT devices in hexadecimal and decimal representation.
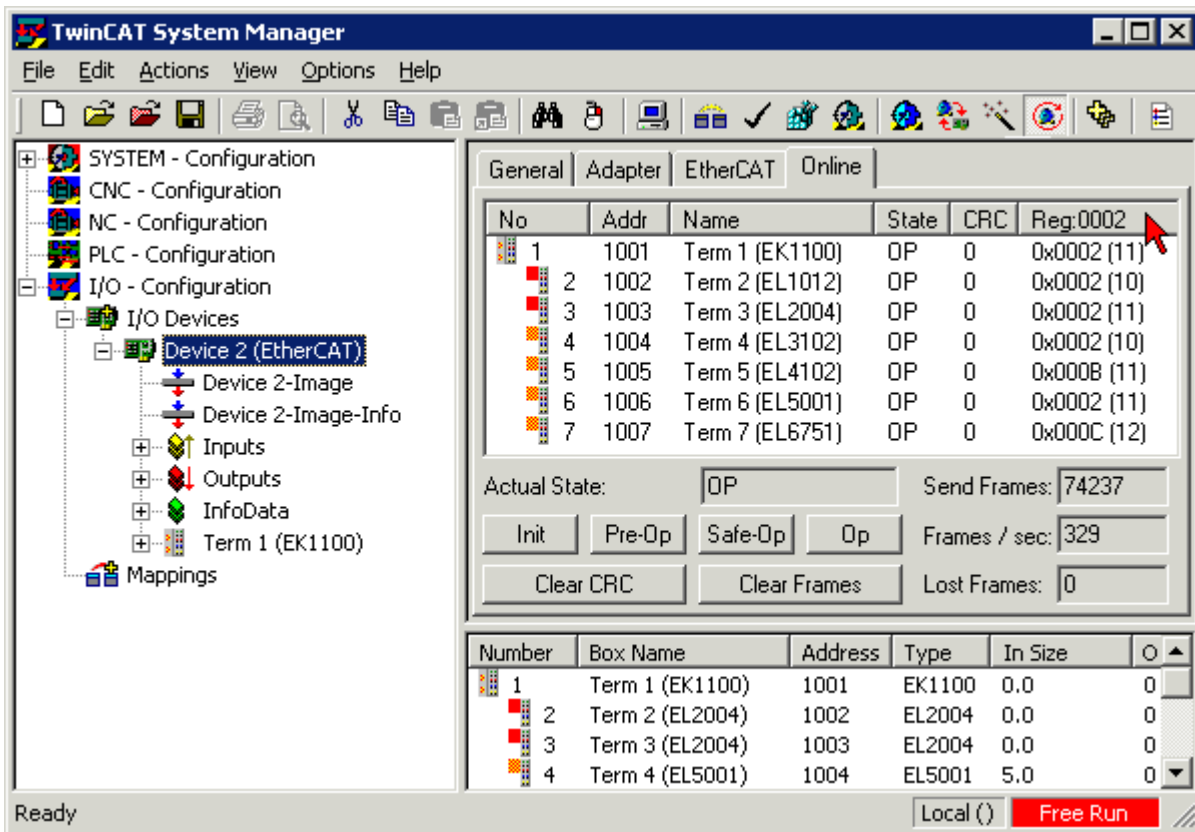
Fig. 60: FPGA firmware version definition

If the column *Reg:0002* is not displayed, right-click the table header and select *Properties* in the context menu.
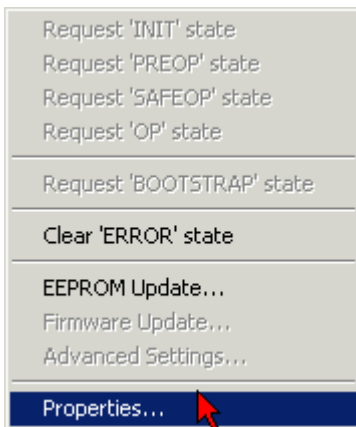
Fig. 61: Context menu *Properties*

The *Advanced Settings* dialog appears where the columns to be displayed can be selected. Under *Diagnosis/***Online View** select the *'0002 ETxxxx Build'* check box in order to activate the FPGA firmware version display.
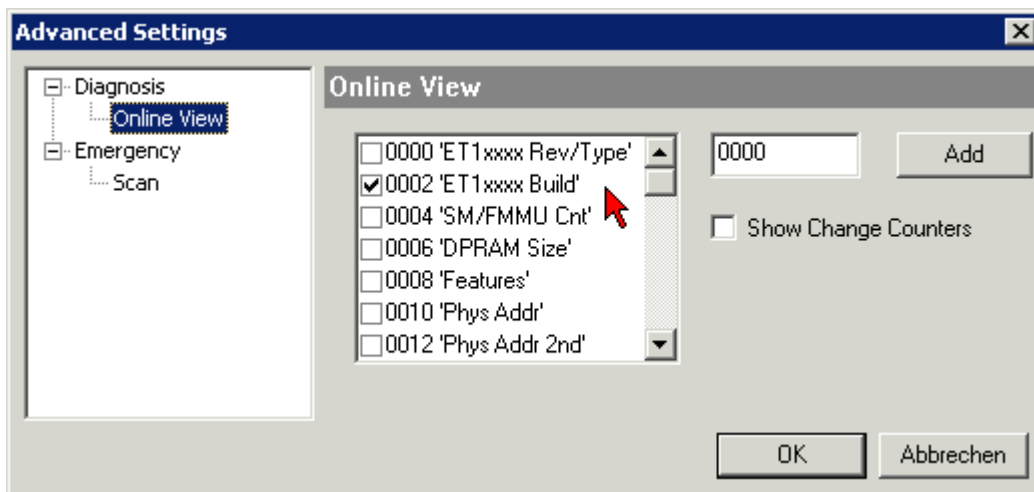
Fig. 62: Dialog *Advanced Settings*

Update

For updating the FPGA firmware

- of an EtherCAT coupler the coupler must have FPGA firmware version 11 or higher;
- of an E-Bus Terminal the terminal must have FPGA firmware version 10 or higher.

Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

The following sequence order have to be met if no other specifications are given (e.g. by the Beckhoff support):

- Switch TwinCAT system to ConfigMode/FreeRun with cycle time >= 1 ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.

BECKHOFF

- In the TwinCAT System Manager select the terminal for which the FPGA firmware is to be updated (in the example: Terminal 5: EL5001) and
  click the *Advanced Settings* button in the *EtherCAT* tab:
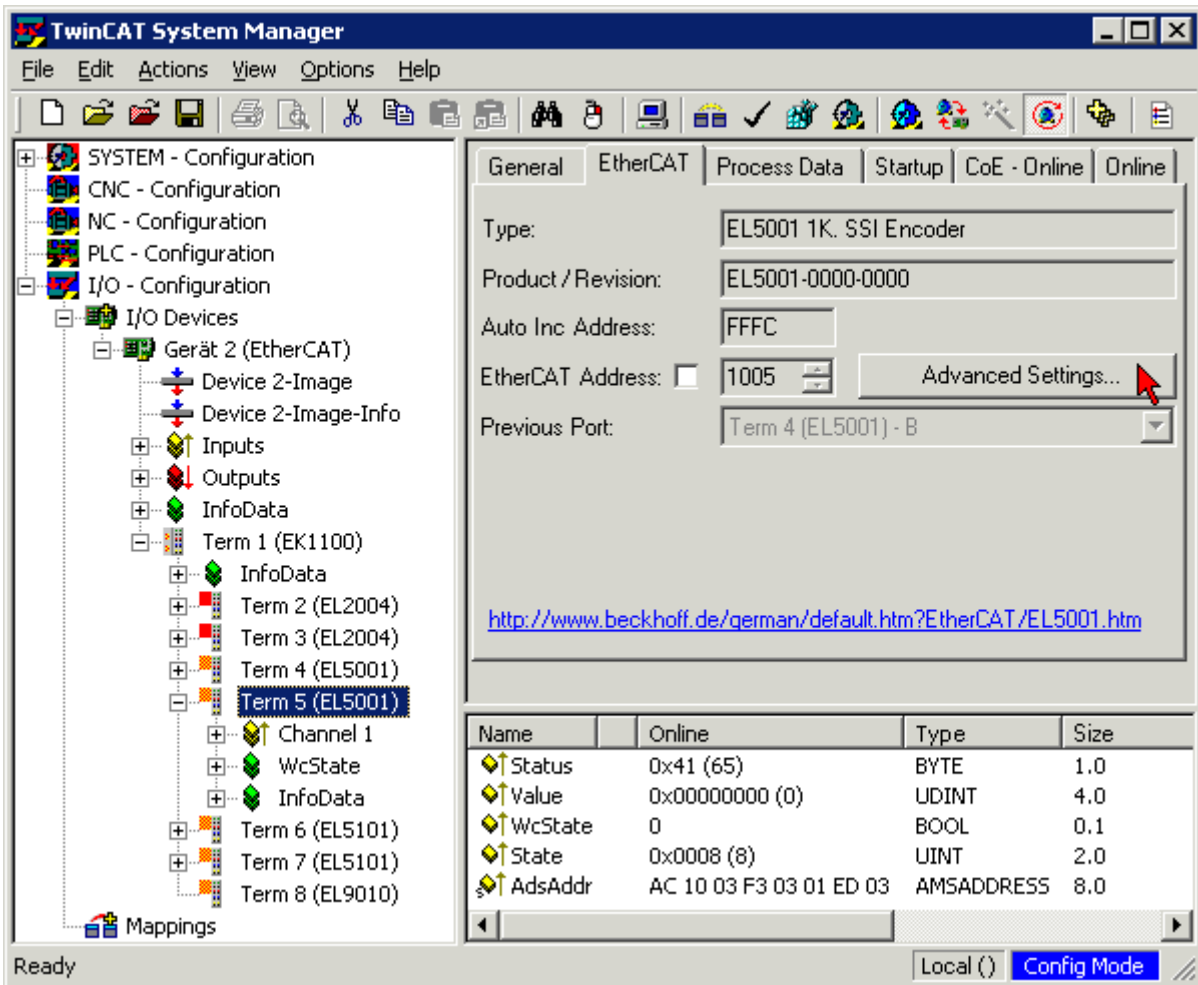
- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM*/FPGA click on *Write FPGA* button:
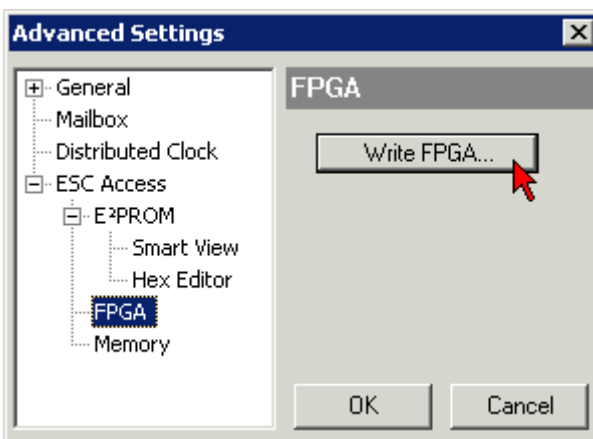
- Select the file (*.rbf) with the new FPGA firmware, and transfer it to the EtherCAT device:
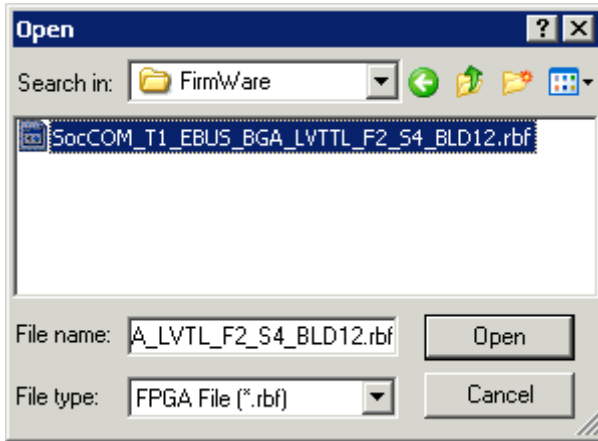
- Wait until download ends
- Switch slave current less for a short time (don't pull under voltage!). In order to activate the new FPGA firmware a restart (switching the power supply off and on again) of the EtherCAT device is required.
- Check the new FPGA status

NOTE
Risk of damage to the device!
A download of firmware to an EtherCAT device must not be interrupted in any case! If you interrupt this process by switching off power supply or disconnecting the Ethernet link, the EtherCAT device can only be recommissioned by the manufacturer!

6.3.5 Simultaneous updating of several EtherCAT devices

The firmware and ESI descriptions of several devices can be updated simultaneously, provided the devices have the same firmware file/ESI.
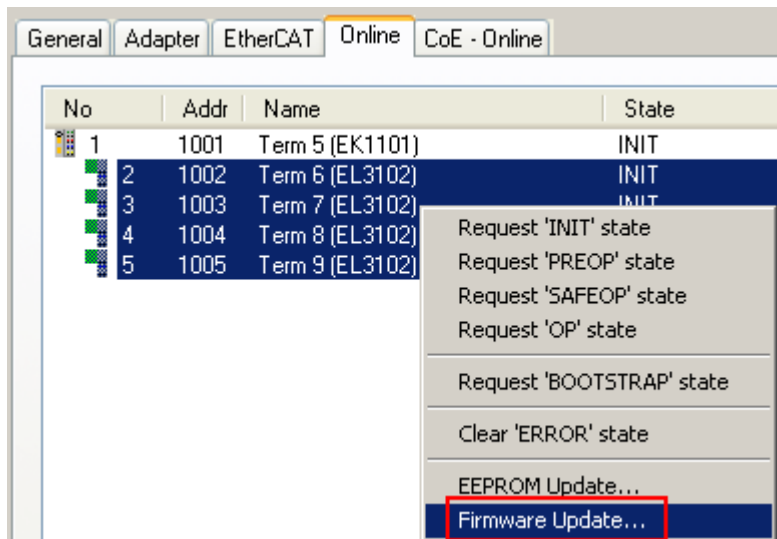
Fig. 63: Multiple selection and firmware update

Select the required slaves and carry out the firmware update in BOOTSTRAP mode as described above.

6.4 Master device file GSDML

The GSDML file describes a PROFINET device. This file is necessary for the PROFINET Controller, and must be integrated in the corresponding configuration software. You can find the required GSDML files in the TwinCAT\IO\PROFINET folder, or on the website at www.beckhoff.de under Download -> Configuration files -> TwinCAT.

Comment	GSDML File Name*	PROFINET Device
Supplement Profinet Device	GSDML-V2.1-beckhoff-TCPNDevice-20091009	PROFINET device for FC9xxx cards with Intel chipset or Ethernet interfaces with Intel chipset, for OS with XP, XPemb
Supplement Profinet Device CE	GSDML-V2.1-beckhoff-TCPNDevice-20091009	PROFINET device for FC9xxx cards with Intel chipset or Ethernet interfaces with Intel chipset, for OS CE
EL6631-0010	GSDML-V2.2-beckhoff-EL6631-20100309.xml	EL6631-0010

* The file name may change as new features or versions are introduced.

6.5 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 5246 963 0
Fax:	+49 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 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

List of illustrations