Documentation

CU1128

EtherCAT hub

Version: 2.0
Date: 2016-02-10

BECKHOFF

Table of contents

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 following notes and explanations are followed when installing and commissioning these components.

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. For that reason the documentation is not in every case checked for consistency with performance data, standards or other characteristics. In the event that it contains technical or editorial errors, we retain the right to make alterations at any time and without warning. 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®, Safety over EtherCAT®, TwinSAFE®, XFC® and XTS® are registered trademarks of and licensed by Beckhoff Automation GmbH & Co. KG.
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, DE102004044764, DE102007017835 with corresponding applications or registrations in various other countries.

The TwinCAT Technology is covered, including but not limited to the following patent applications and patents: EP0851348, US6167425 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 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.
 **Attention**	**Damage to the environment or devices** Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.
 **Note**	**Tip or pointer** This symbol indicates information that contributes to better understanding.

1.3 Documentation issue status

Version	Comment
2.0	- Migration
1.0	- Addenda - First public issue
0.1	- Preliminary version

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
CU2008-0000-0000	CU device	2008 (8-port fast ethernet switch)	0000 (basic type)	0000
ES3602-0010-0017	ES terminal (12 mm, pluggable connection level)	3602 (2-channel voltage measurement)	0010 (high-precision version)	0017

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 website.
  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 batch number and revision ID (since 2014/01)

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

Fig. 3: CU2016 switch with batch number

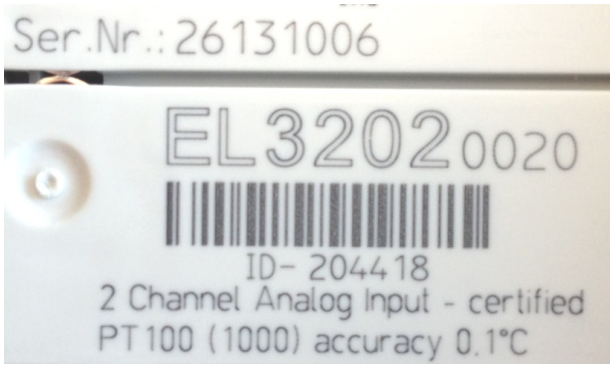


Fig. 4: EL3202-0020 with batch numbers 26131006 and unique ID-number 204418

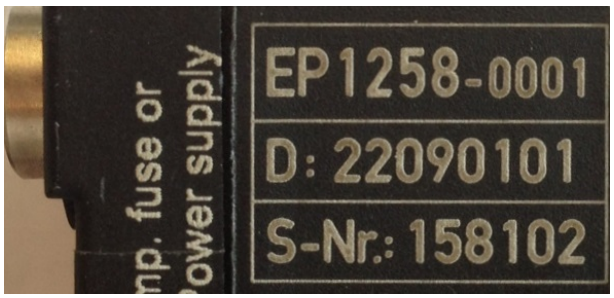


Fig. 5: EP1258-00001 IP67 EtherCAT Box with batch number 22090101 and unique serial number 158102

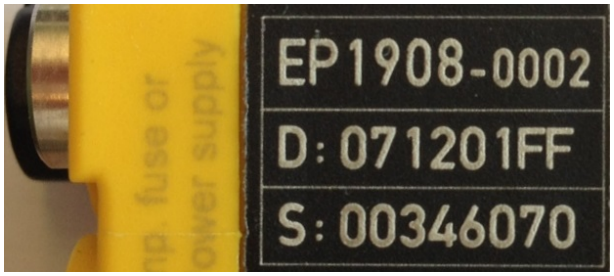


Fig. 6: EP1908-0002 IP76 EtherCAT Safety Box with batch number 071201FF and unique serial number 00346070

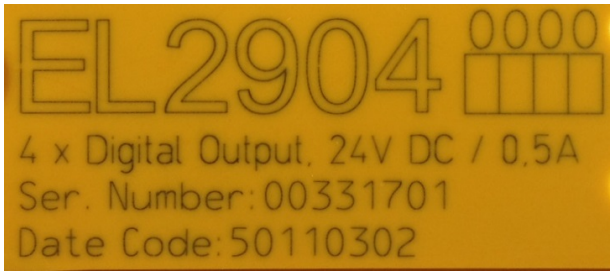


Fig. 7: EL2904 IP20 safety terminal with batch number/date code 50110302 and unique serial number 00331701

2 Product overview

2.1 Introduction

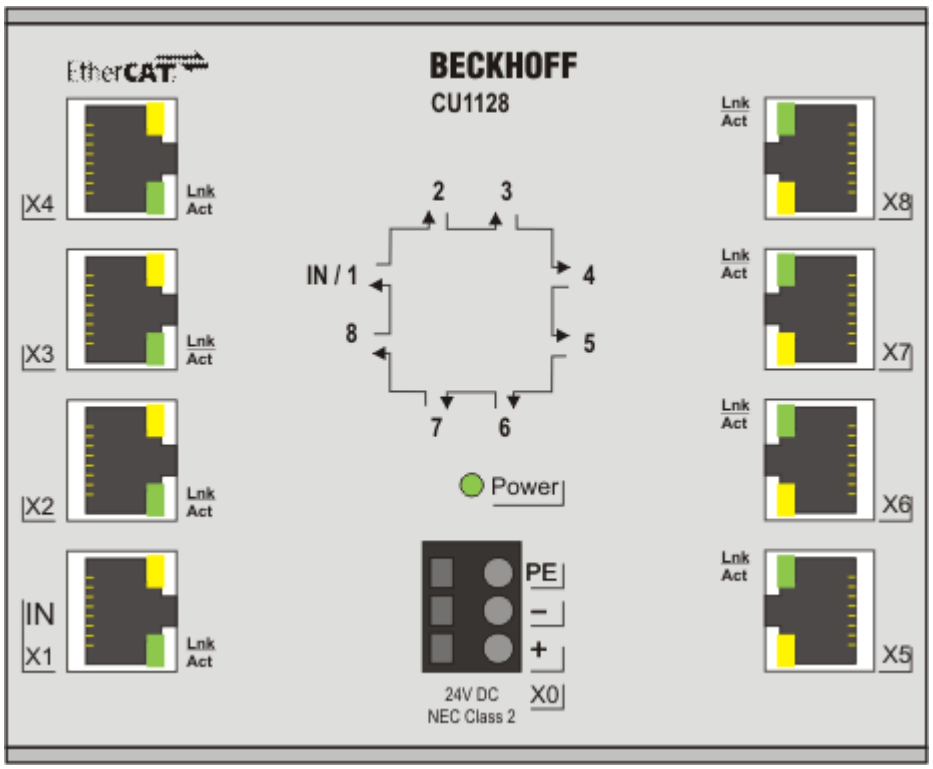


Fig. 8: CU1128

EtherCAT hub

Line, tree or star: EtherCAT supports almost any topology. If a star topology requires several junctions at a particular point, the 8-way CU1128 EtherCAT hub can be used instead of several EK1122 devices. Port 1 is the input port for the network. Further EK1100 or EtherCAT Box modules can be connected at ports 2 to 8. The EtherCAT junctions are connected via RJ-45 sockets with direct display of link and activity status.

In conjunction with TwinCAT or other suitable EtherCAT masters the CU1128 also supports coupling and uncoupling of EtherCAT segments during operation (hot-connect). The device cannot be used as a standard Ethernet switch.

2.2 Technical data

Technical data	CU1128
Task in the EtherCAT system	Coupling of EtherCAT junctions
Transmission medium	Ethernet/EtherCAT cable (min. CAT 5), shielded
Bus interface	8 x RJ 45
Distance between stations	max. 100 m (100BASE-TX)
Protocol	EtherCAT
Distributed Clocks	Yes
Delay	approx. 1 µs per port
Data transfer rates	100 Mbaud
Power supply	via three-pole spring loaded terminal (+, -, PE)
Supply voltage	24 V_{DC} (18 V_{DC} to 30 V_{DC})
Current consumption	typ. 185 mA
Weight	approx. 430 g
Dimensions without connector (W x H x D)	approx. 122 mm x 100 mm x 38 mm
Mounting [▶ 22]	on 35 mm mounting rail (EN 50022)
Permissible ambient temperature range during operation	0°C ... + 55°C
Permissible ambient temperature range during storage	-25°C ... + 85°C
Permissible relative humidity	95 %, no condensation
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	variable
Approval	CE UL [▶ 72]

3 Basic principles

3.1 Basic function principles

The CU1128 is an infrastructure device without controllable input/output data (I/O). It can be used

- as a junction point for conducted Fast Ethernet, in order to connect EtherCAT terminal stations, drives or any other EtherCAT slaves to drop lines.
- as distributed clock reference clock (see notes [▶ 64]).

It has no I/O and no CoE directory and is not parameterizable. The core functions of the link control and distributed clocks synchronization are mapped by the ESCs.

Structure

In order to provide 8 EtherCAT connections, the CU1128 has 3 internal communication ICs (ESCs), which are connected in series internally. For this reason the CU1128 appears as 3 individual slaves in the EtherCAT configurator, although they are located in one housing. The interrelationship between the internal ESC ports (A, B, C, D) and the descriptions of the connection sockets (1..8) is as follows:

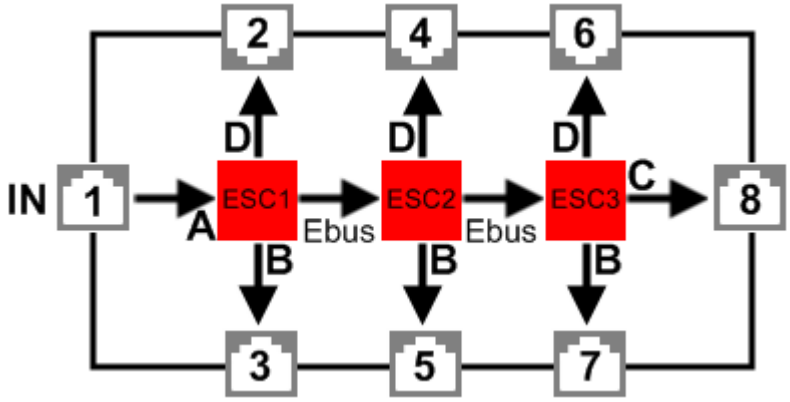


Fig. 9: diagram CU1128

Accordingly, 3 devices are displayed in the TwinCAT System Manager:

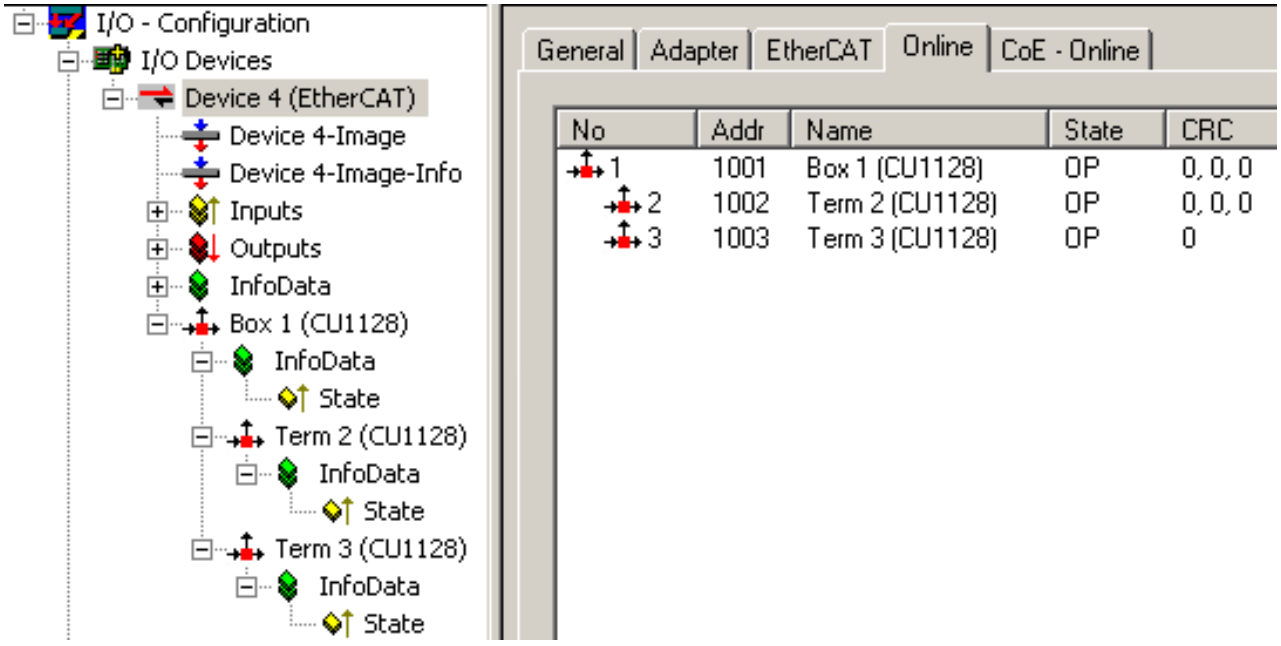


Fig. 10: 3 slaves in the TwinCAT System Manager

Please note:

- port 1 is always the input for the EtherCAT traffic in the CU1128.
- the other ports 2..7 should be used as outputs.
- it is not permitted to delete subdevices once the CU1128 has been configured.

For differentiation see also EEPROM Update [▶ 72].

Topological configuration

With the CU1128, special attention should be paid to the sequence of the EtherCAT slaves. Since the CU1128 has 7 junction ports, drop lines connected to ports must and can be clearly identified in practice. If incorrect information is provided in the configuration (TwinCAT System Manager file *.tsm), the system cannot start.

For each EtherCAT device the System Manager indicates at which *PreviousPort* it is connected, i.e. the name of the connected port (B to D) of the previous slave. This also applies for the internal connections between the ESCs in the CU1128:

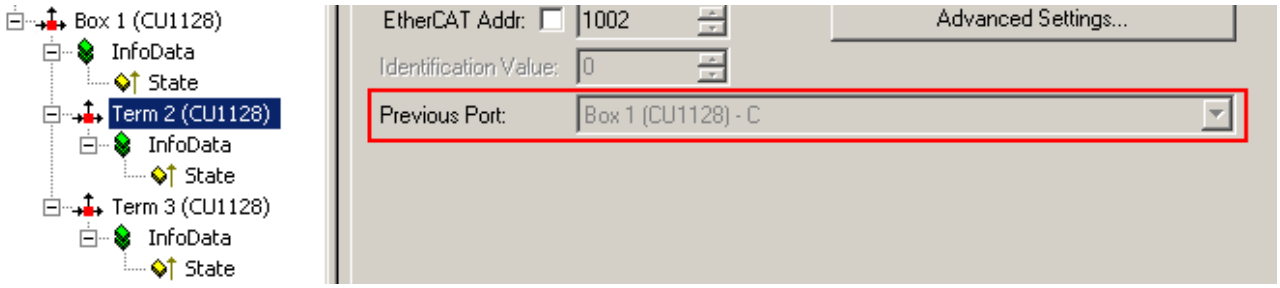


Fig. 11: previous port of the second ESC in the CU1128

In Fig. *previous port of the second ESC in the CU1128*, for example, it is stated that the second ESC in the CU1128 "Box 1", called "Term 2", is connected to ESC 1 Port C; cf. Fig. *diagram CU1128*. This setting cannot be changed, since it is defined by the CU1128 device.
In general, however, it is changeable if a pluggable coupler, EtherCAT Box or similar is present in the configuration; see Fig. *setting the PreviousPort for a EK1100*. Then,

- drag & drop with the mouse can then be used in the System Manager to set up the general position in the I/O system.
- the previous port in the "fine adjustment" can be selected, if several options are available.

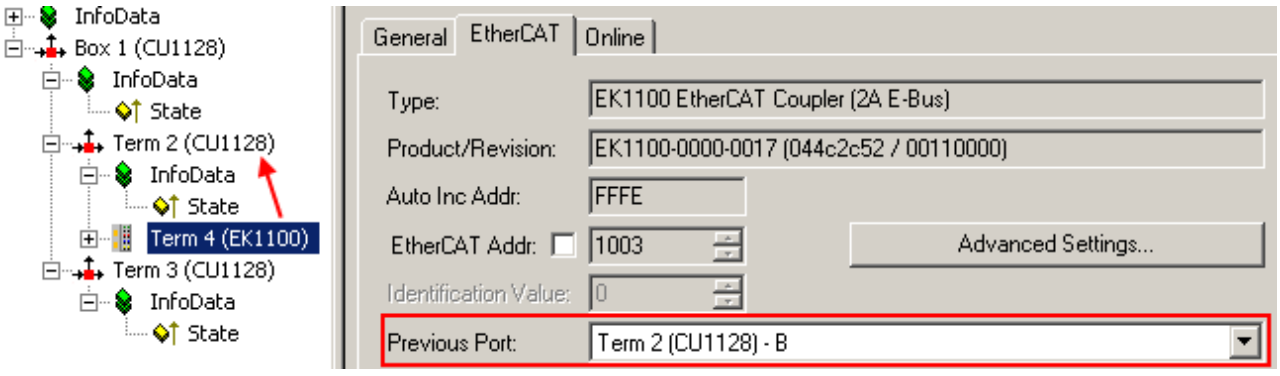


Fig. 12: setting the PreviousPort for a EK1100

In Fig. *setting the previous port for a EK1100*, the coupler "Term 4" is set up as successor to the ESC2 "Term 2". Both free ports of the ESC2 (B and D, see Fig. *diagram CU1128*) are thus available in the previous port selection of the EK1100; *B* is selected here. The System Manager detects whether ports can technically and actually be connected by means of the port property, i.e. *Ethernet* or *E-bus* in the ESI/XML device descriptions. The internal connections in the CU1128 are E-bus connections, the 8 ports on the other hand are Ethernet, see Fig. *diagram CU1128*.

See also notes for configuration setup [▶ 66].

3.2 Basic function principles of EtherCAT junctions

Some Beckhoff EtherCAT devices can be used for junctions in the EtherCAT segment. These include EK1122, EK1521, EP1122 or also CU1128. In the following examples only the EK1122 is used. The technical and system characteristics of the other devices are similar. test

EtherCAT handling in the slaves

With EtherCAT as fieldbus protocol a wide range of bus topologies can be used: line, star and tree topology, with redundancy support even ring topology. The simplest topology is the line topology, in which each EtherCAT slave passes the data on to the *only* next slave; see following Fig. "*EtherCAT line topology*".

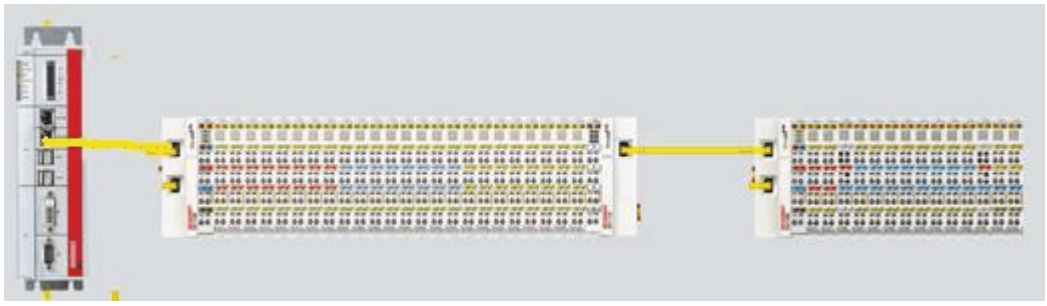


Fig. 13: EtherCAT line topology

When using, for example, EK1100 EtherCAT Couplers, a junction and thus a type of tree topology is possible; see following Fig. *"Line topology with extensions"*.

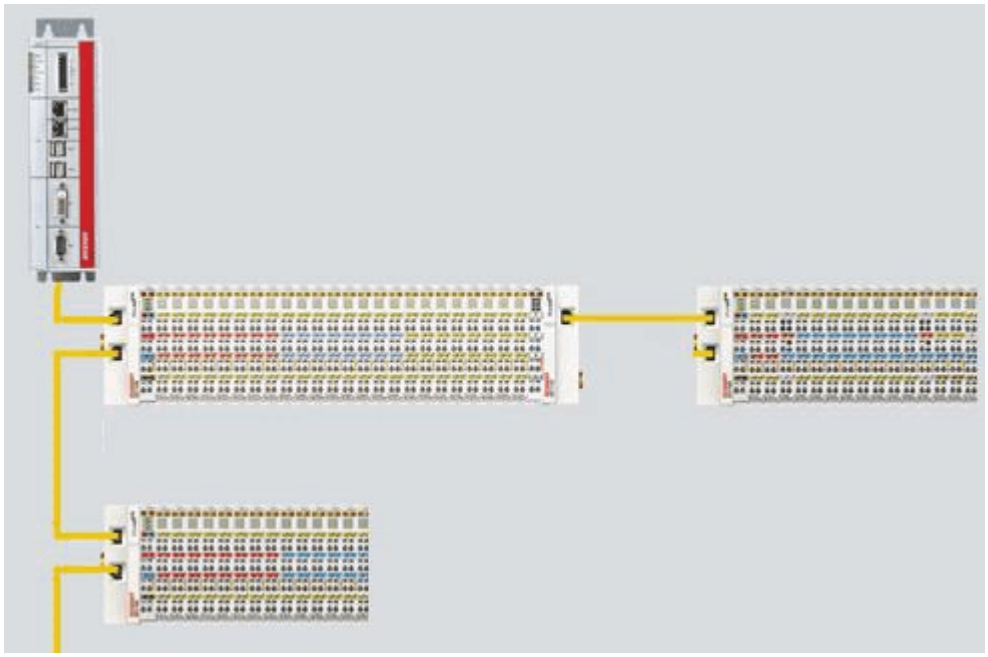


Fig. 14: Line topology with extensions

The basic principle is that internally the Ethernet frame(s) with the EtherCAT protocol data continue to be transported in a logical ring:

- the EtherCAT master sends the frame on the two outgoing lines of the Ethernet cable
- this frame passes each slave once,
- is reversed by the last slave in the logical sequence
- and is returned to the master through each EtherCAT slave via two return lines of the Ethernet cable without further processing.

At short cycle times in the order of 50 µs at 20,000 Ethernet frames are in transit in the EtherCAT system every second, plus acyclic organizational frames. The master awaits the return of the sent frames, which return the device input data to the master, for example. Telegram transfer between slaves is link-based: An

EtherCAT slave will only forward a frame if a 'link' signal to the next device is present. Normally it can be assumed that the downstream device correctly processes each EtherCAT telegram and returns or process it at the end.

The crucial factor for forwarding EtherCAT telegrams is that a link signal is reported only from one slave to the next if both slaves are actually ready for real-time participation in data processing. Specifically, this means that an EtherCAT slave should not open the respective Ethernet port until it is ready to receive and forward an Ethernet frame immediately.

A switch or router is usually used for standard Ethernet traffic forwarding. Any collisions or frame losses are compensated through frame repetition in the higher level protocol layers (e.g. TCP). This mode is generally not used for EtherCAT due to the short cycle times and the real-time requirement. Some Ethernet devices such as special switches, for example, report a link to the remote terminal even if they will only be ready for data processing in a few milliseconds. This behavior is particularly noticeable in media converters from 100Base-TX (copper) to 100Base-Fx (optical fiber), which may report a link to the preceding EtherCAT slave even if the optical fiber connection is interrupted, depending on the setting on the copper side.

Fast link detection is therefore a central component of each ESC (EtherCAT slave controller, hardware processing unit for the EtherCAT protocol). According to the EtherCAT specification an ESC can have and control 1 to 4 ports. Via an open port it can handle outgoing and incoming Ethernet traffic. The direction of data flow in a fully configured ESC is shown in Fig. "*Direction of data flow in the ESC*" – the data in the EtherCAT datagrams are thereby processed only between Ports 0 (A) and 3 (D) in the EtherCAT processing unit.

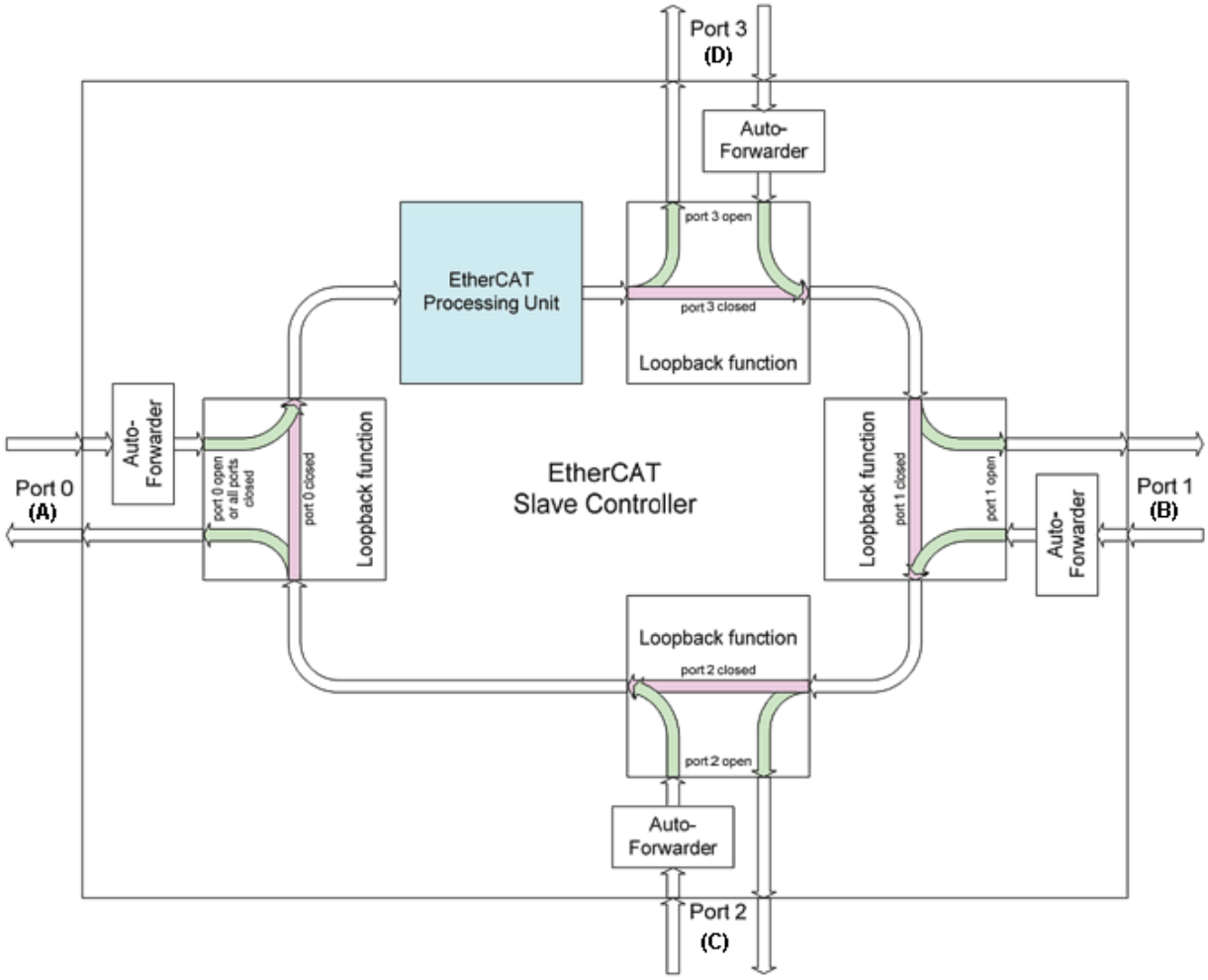


Fig. 15: Direction of data flow in the ESC

Ideally link detection and therefore port handling in the ESC should be fast enough that lost frame events are avoided even at 100 µs cycle time. Nevertheless, at least one lost frame can never be ruled out if a connection is disconnected while an Ethernet frame is in transit on this line and in the bus segment downstream of the separation point.

Implementation: EL terminal

A standard EtherCAT slave such as a Beckhoff EL terminal has 2 ports:

- one for incoming frames (port 0 [A])
- and one for outgoing frames (e.g. port [D]).

The other two ports are internally closed in the ESC. An EtherCAT telegram enters the processing unit via port 0 (A)/top and is forwarded to the next slave via port 3 (D)/left, if a link to this port exists - see green arrows. This is the case if a further EL terminal is connected to the right.

If no link exists, the frame is forwarded to port 1(B) via the purple route. This and port 2 (C) have no link and therefore return the frame to port 0 (A), where the frame leaves via the same Ethernet port through which it arrived at the slave. This is the case if the terminal acts as end terminal.

An EtherCAT device with a single port is therefore only of limited use, since it can only be used as end device.

Implementation: EK1100 EtherCAT Coupler

Three of the four available ports in the EK1100 EtherCAT Coupler are used, thus enabling a connection to the right to terminals and via an RJ45 socket to further couplers; cf. Fig. "Line topology with extensions [▶ 14]". In the EK1100 the processing unit is not used for process data exchange.

Implementation: EK1122 EtherCAT junction

In the EK1122 all 4 ESC ports can be connected - two via the internal E-bus and two via the RJ45 sockets with Ethernet configuration. In the TwinCAT System Manager the link statuses of ports 0, 1, 2 and 3 are shown by the online display – they are designated there as ports A, B, C and D; see Fig. "Topology display for interrupted line [▶ 19]".

Implementation: EK1521 / EK1521-0010 / EK1561 EtherCAT junction

As in the EK1100, three ESC ports can be connected in these junctions: Two via E-bus within the terminal and one via the SC socket/versatile link and optical fiber cable/POF line.

Implementation: CU1128 EtherCAT junction

The CU1128 integrates three ESCs, which means eight ports in total are available to users. The three ESCs are interconnected via E-bus.

Example configuration with EK1122

The following section describes the link characteristics under TwinCAT and its representation in the System Manager.

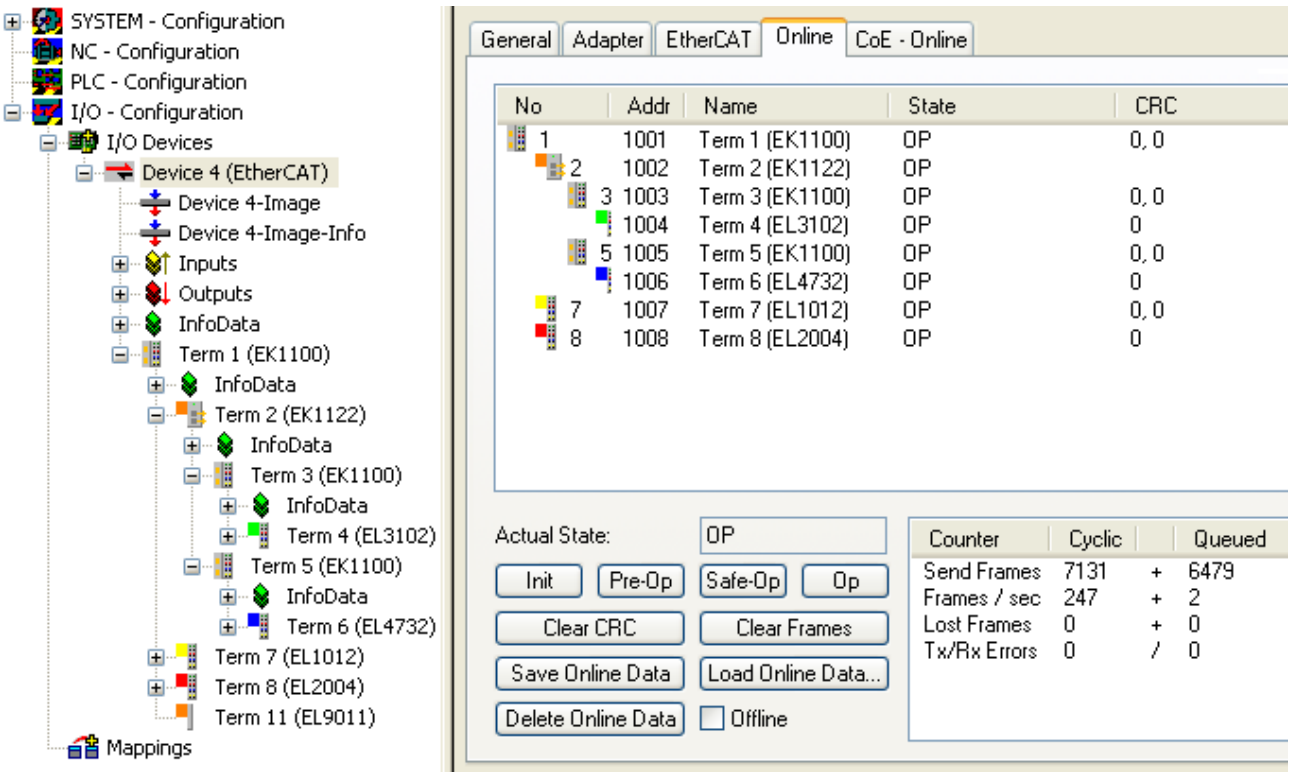


Fig. 16: Example configuration

The TwinCAT online topology shows the wiring scheme, see Fig. "*Online Topology*". The EK1122 is selected, so that further information is shown. The green bars above the slaves indicate the correct RUN state in all slaves.

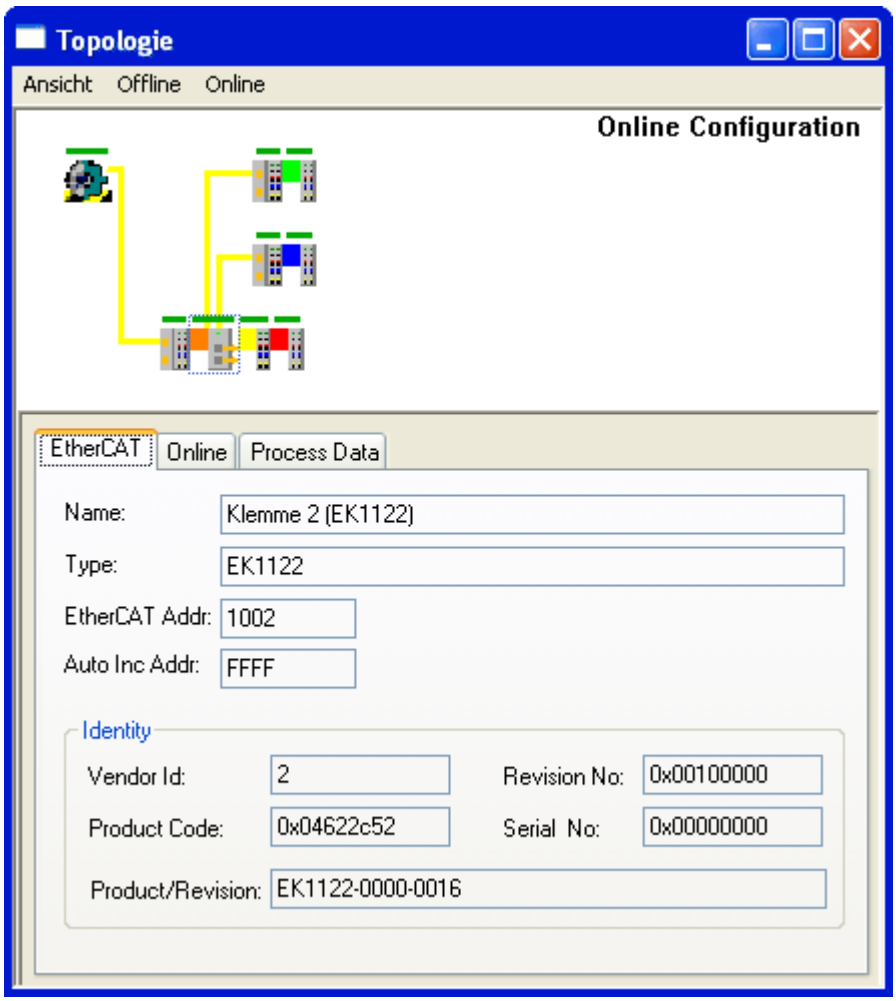


Fig. 17: Online topology

An error is now generated by disconnecting the connection between the upper RJ45 socket (X1) and the EL3102 device. Within a few µs the ESC in the EK1122 detects the lost link and automatically closes the affected port. This has the effect that the next incoming EtherCAT telegram is immediately forwarded to port D (port 3) and the EL4732. The link is thus missing here and the System Manager marks this in the online display; see following Fig. "*Example configuration with interrupted cable*".

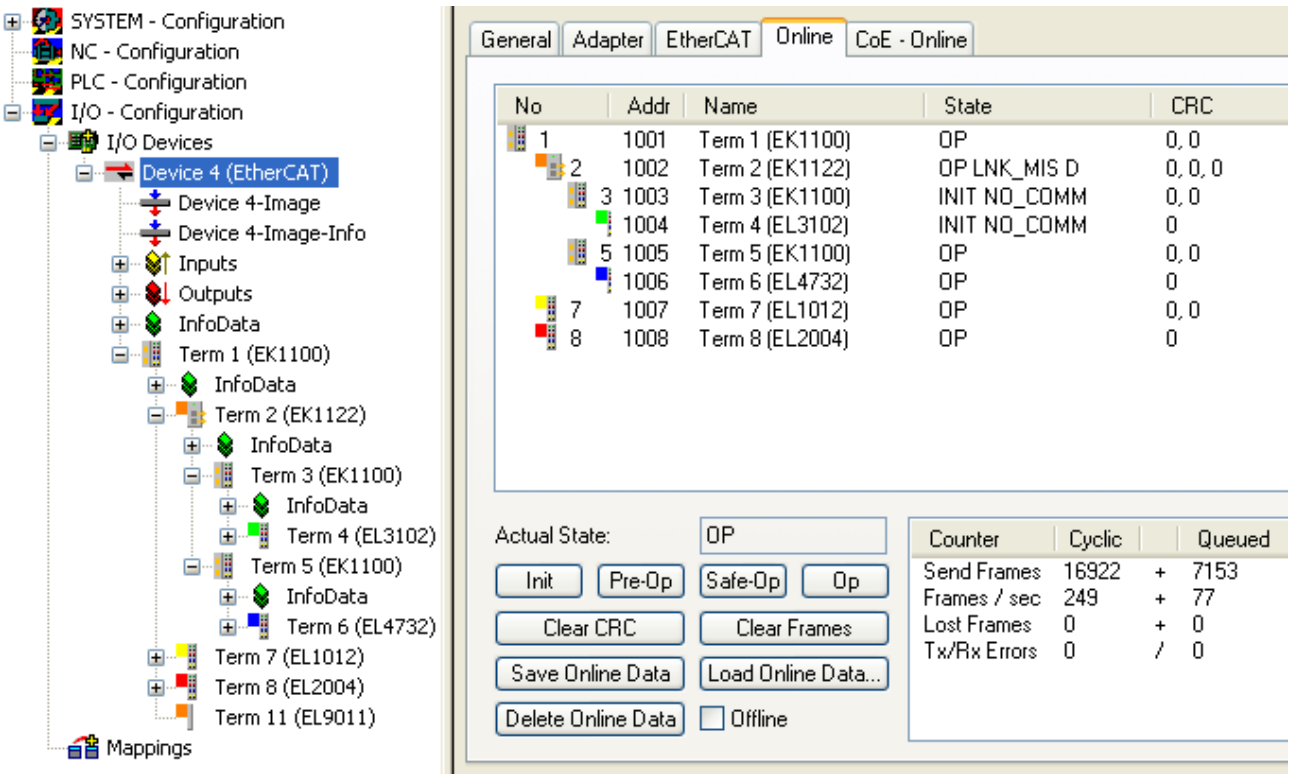


Fig. 18: Example configuration with interrupted cable

The System Manager messages can be interpreted as follows:

- Address 1002 - EK1122: "OP LNK:MIS D": The slave is in OP state, although a link is missing at port D (3) that should be present according to the configuration
- Address 1003 - EK1100: "INIT NO_COMM": Since communication with this slave is interrupted its state is shown as INIT
- Address 1004 - EL3104: ditto

Note	**Logger output** The logger output can be displayed in the lower part of the System Manager (Display → Show Logger Output). This may be helpful for diagnostic purposes (for link interruptions and other situations).

In the topology display any slaves affected by interruption are shown with a red border, see the following Fig. "*Topology display for interrupted line*".

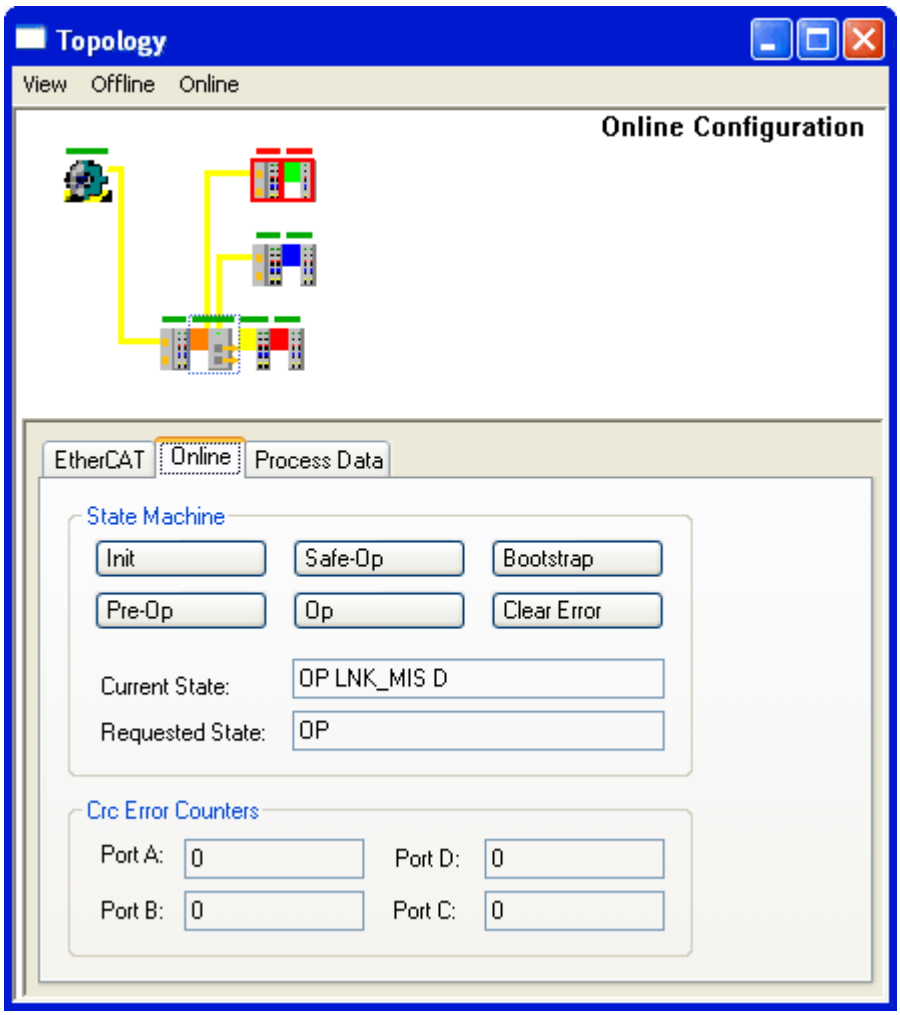


Fig. 19: Topology display for interrupted line

In Fig. "Example configuration [▶ 17]" and Fig. "Example configuration with interrupted cable" [▶ 19] note the display of acyclic frames, see the following Fig. "*Comparison of the frame displays in the System Manager*".

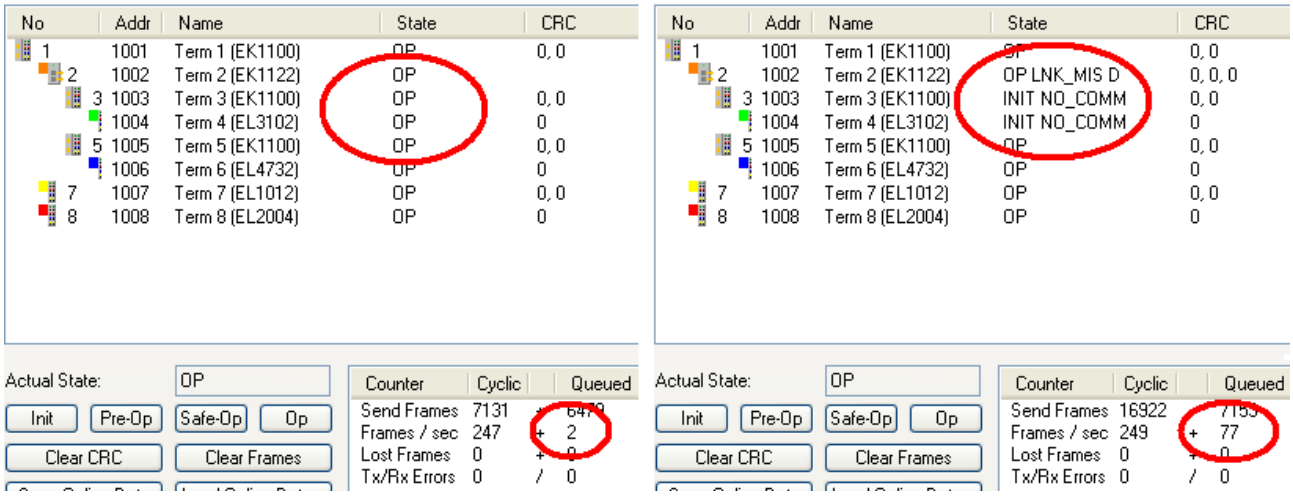


Fig. 20: Comparison of the frame displays in the System Manager

The image on the left shows a small number (2) of acyclic frames sent by the master during the respective second - all slaves are operating properly. The image on the right shows a significant increase (currently 77 acyclic frames/sec): The EtherCAT master has quickly detected that not all slaves are properly taking part in the data exchange. Once the master has located the fault, it continuously tries to restore the connection.

Reconnection

Once the connection has been restored, the EK1122 reports to the master that a link is present again at port D (3). The EtherCAT master will then make its process data available again for this section. One the preparations are complete, it will instruct the EK1122 to re-open port D (3) for regular data exchange. Cyclic and acyclic data traffic with the other EtherCAT slaves continues normally.

Note	**External access to EtherCAT diagnostics** The system offers a wide range of options for accessing status and diagnostic information and EtherCAT master functions from the PLC. Almost all information displayed by the System Manager online can also be retrieved via ADS (see figures on this page). System Manager functions can also be triggered via PLC or ADS. Please refer to the relevant sections in the Beckhoff Information System and the notes on EtherCAT diagnostics.

4 Mounting

4.1 Mounting and demounting

The CU20xx und CU22xx switches are fastened to the mounting surface at 35 mm with the aid of a mounting rail according to EN 50022.

Mounting

- Fit the mounting rail to the planned assembly location.
- Suspend the switch on the mounting rail with the spring on the lower side of its latching flange.
- Press the switch upwards (1).
- Press the upper side of the switch (2) against the assembly surface until it latches in the mounting rail.
- Attach the cable.

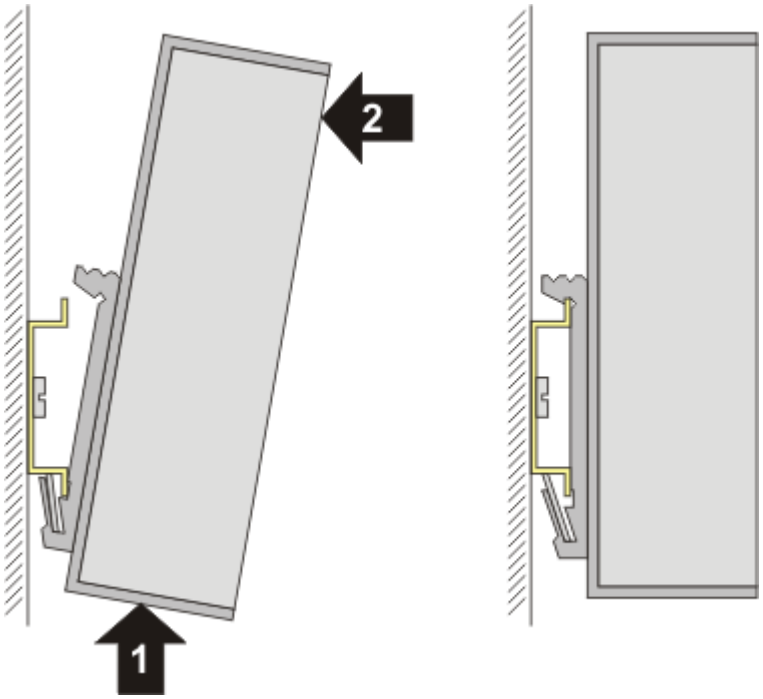


Fig. 21: Mounting

Removal

- Remove all the cables.
- Press the switch upwards (3).
- Pull the other side of the switch (4) away from the assembly surface.

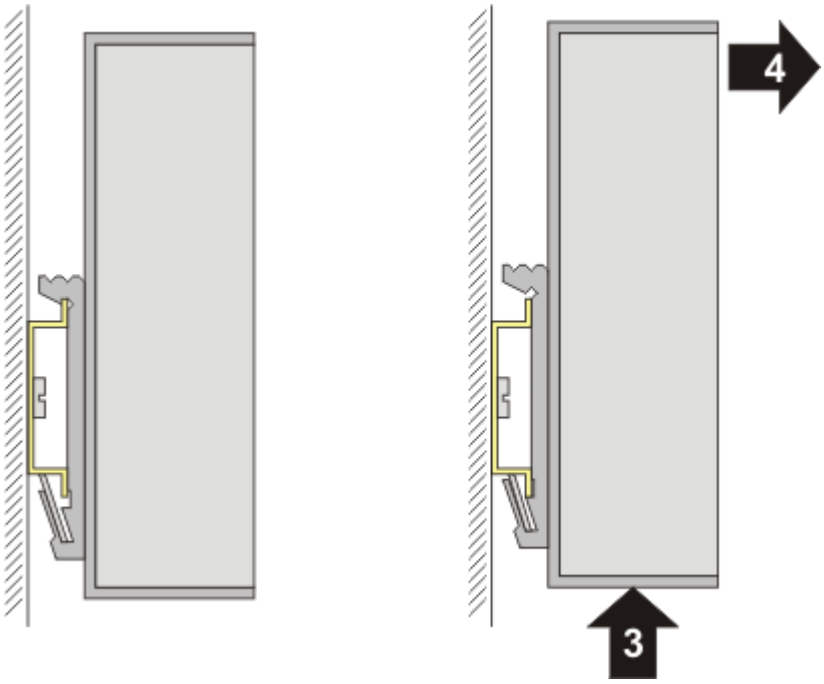


Fig. 22: Removal

4.2 Cabling

Power supply

Pin configuration of the spring loaded terminal

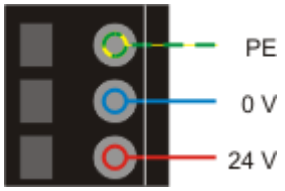


Fig. 23: Pin configuration of the spring loaded terminal

EtherCAT

Pin configuration of the RJ45 sockets

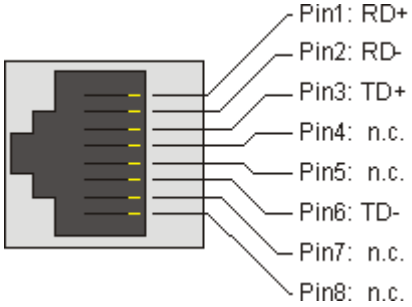


Fig. 24: Pin configuration of the RJ45 sockets

4.3 Dimensions

Note	**Space requirement in the control cabinet** • The RJ45 connector increase the depth depending on their design and the Ethernet cable used. • Above the mounting rail an additional height of approx. 10 mm is required to enable latching [▶ 22] of the switch onto the rail.

CU1128

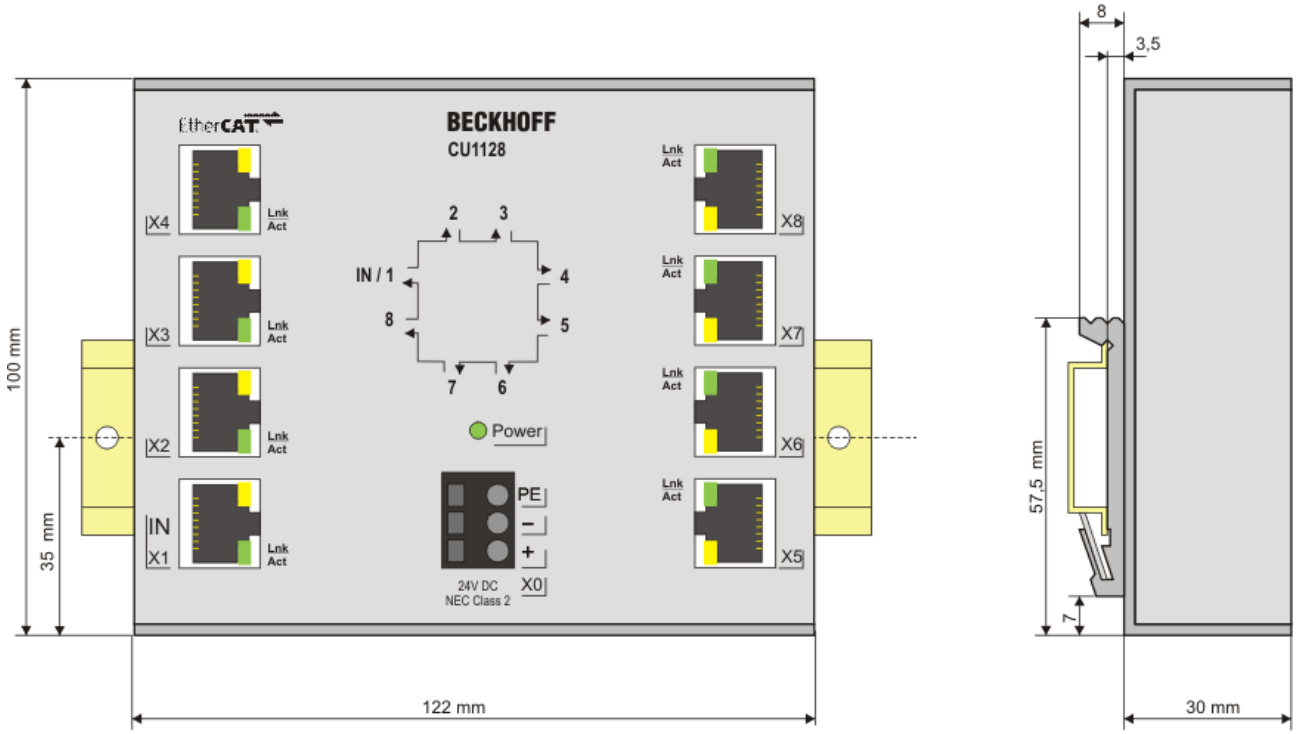


Fig. 25: CU1128 dimensions

4.4 LED Displays

Ethernet

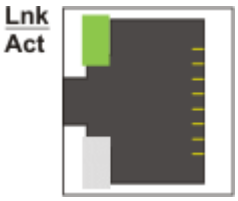


Fig. 26: LEDs

1 LED indicates the current state of each channel.

Table 1: LED display per channel

LED	Display	
Link Act	off	No connection
	on	Connection available (link)
	flashing	Data transfer (act)

Supply voltage

The presence of the supply voltage (24 V_{DC}) is indicated by the green Power LED.

Fig. 27: Green Power LED

5 Commissioning

5.1 Quick start

No special measures are required for commissioning the CU1128.

Install the CU1128 as described in chapter Mounting and wiring [▶ 22].
Notes on the configuration setting can be found in chapter Configuration in the TwinCAT System Manager [▶ 66]

5.2 TwinCAT 2.1x

5.2.1 Installation of the TwinCAT real-time driver

In order to assign real-time capability to a standard Ethernet port of an IPC controller, the Beckhoff real-time driver has to be installed on this port under Windows.

This can be done in several ways. One option is described here.

In the System Manager call up the TwinCAT overview of the local network interfaces via Options -> Show Real Time Ethernet Compatible Devices.

Fig. 28: System Manager option

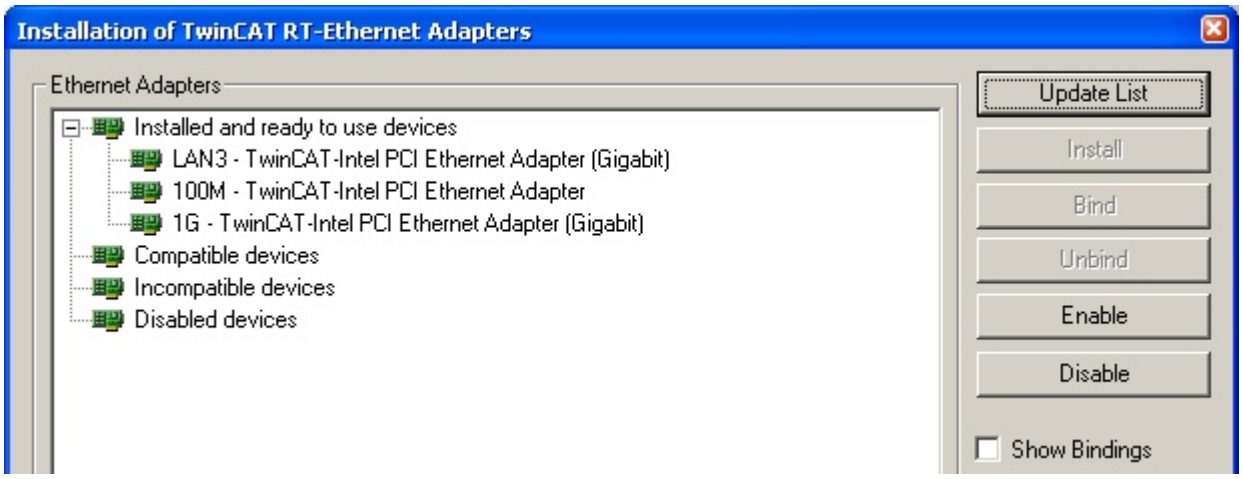


Fig. 29: Overview of network interfaces

Interfaces listed under “Compatible devices” can be assigned a driver via the “Install” button. A driver should only be installed on compatible devices.

A Windows warning regarding the unsigned driver can be ignored.

Alternatively, the compatible Ethernet ports can be viewed in the System Manager via EtherCAT properties.

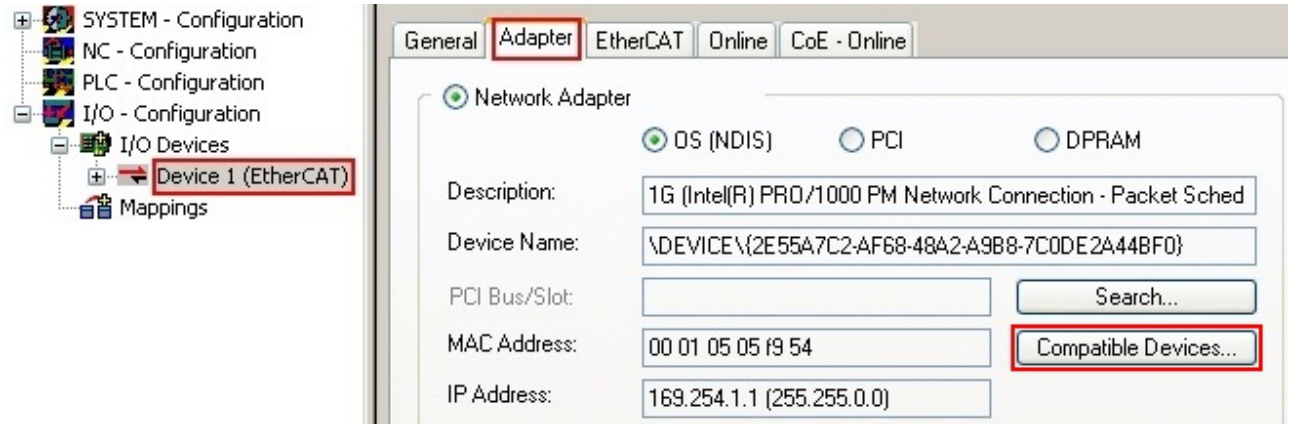


Fig. 30: EtherCAT device properties

After the installation the driver appears activated in the Windows overview for the network interface (Windows Start -->System Properties -> Network)

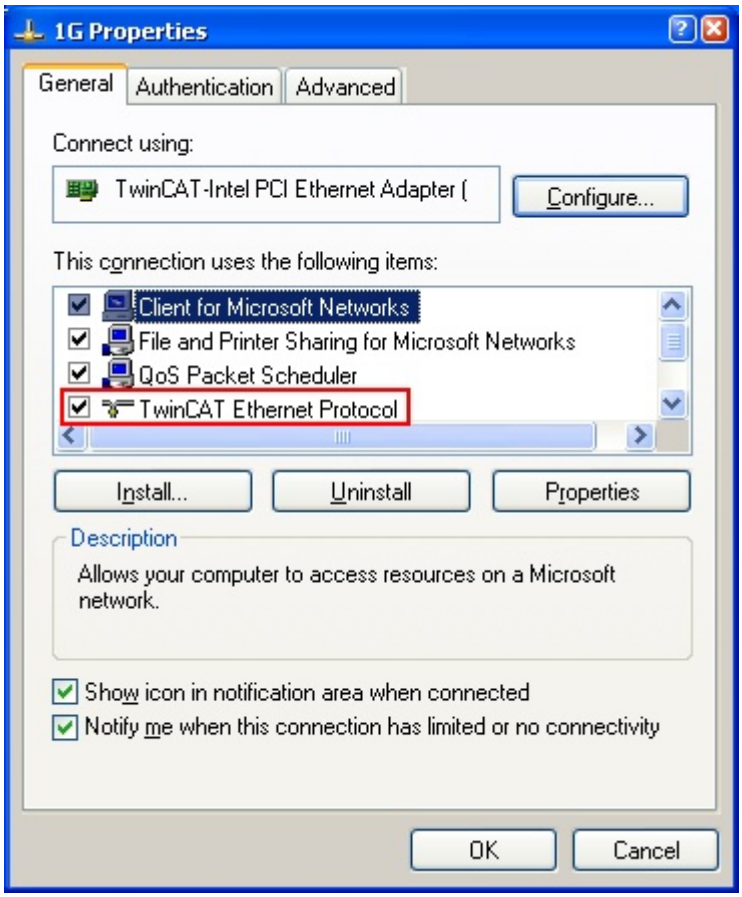


Fig. 31: Windows properties of the network interface

Other possible settings are to be avoided:

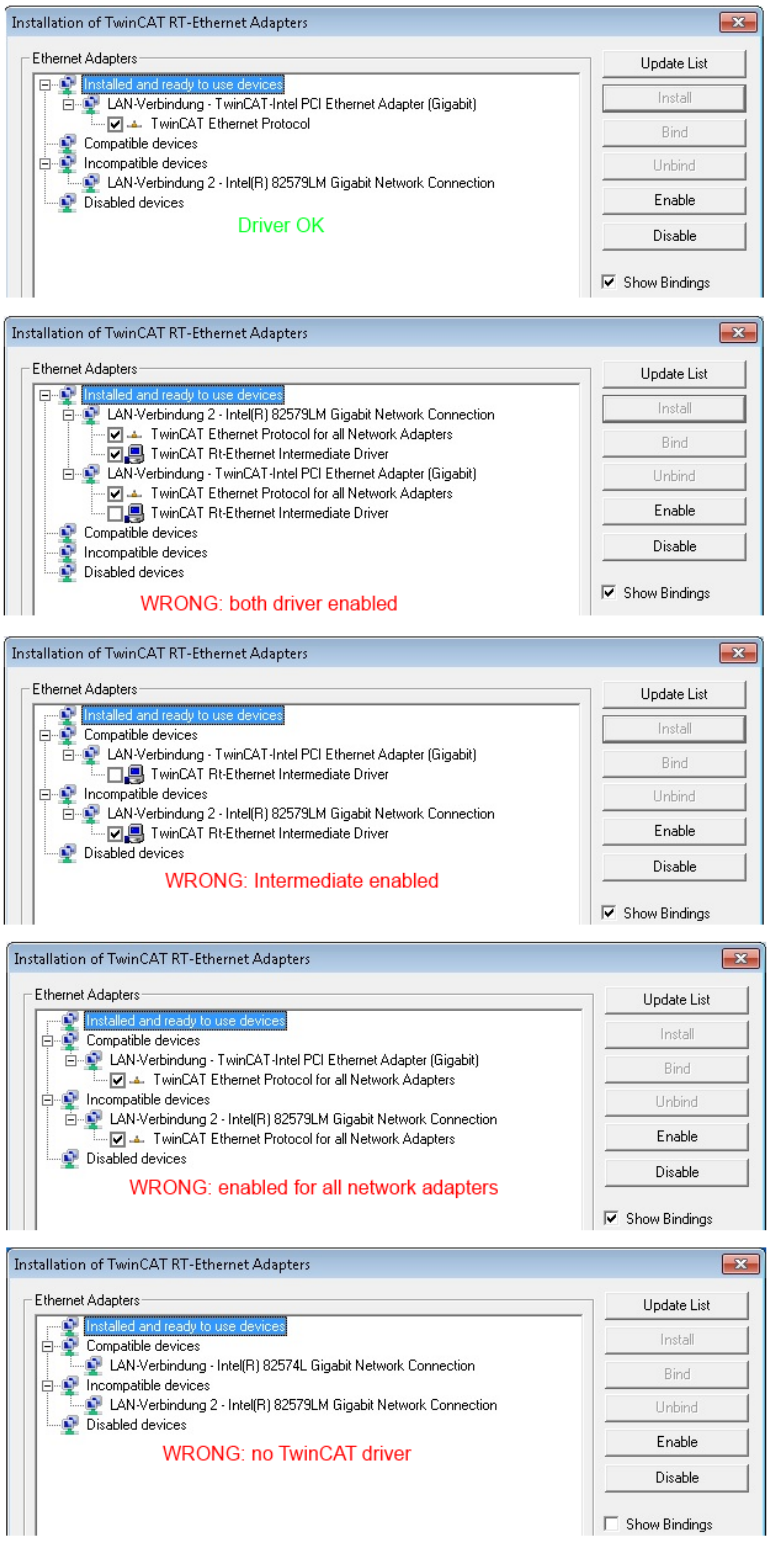


Fig. 32: Incorrect driver settings for the Ethernet port

IP address of the port used

Note	**IP address/DHCP** In most cases an Ethernet port that is configured as an EtherCAT device will not transport general IP packets. For this reason and in cases where an EL6601 or similar devices are used it is useful to specify a fixed IP address for this port via the “Internet Protocol TCP/IP” driver setting and to disable DHCP. In this way the delay associated with the DHCP client for the Ethernet port assigning itself a default IP address in the absence of a DHCP server is avoided. A suitable address space is 192.168.x.x, for example.

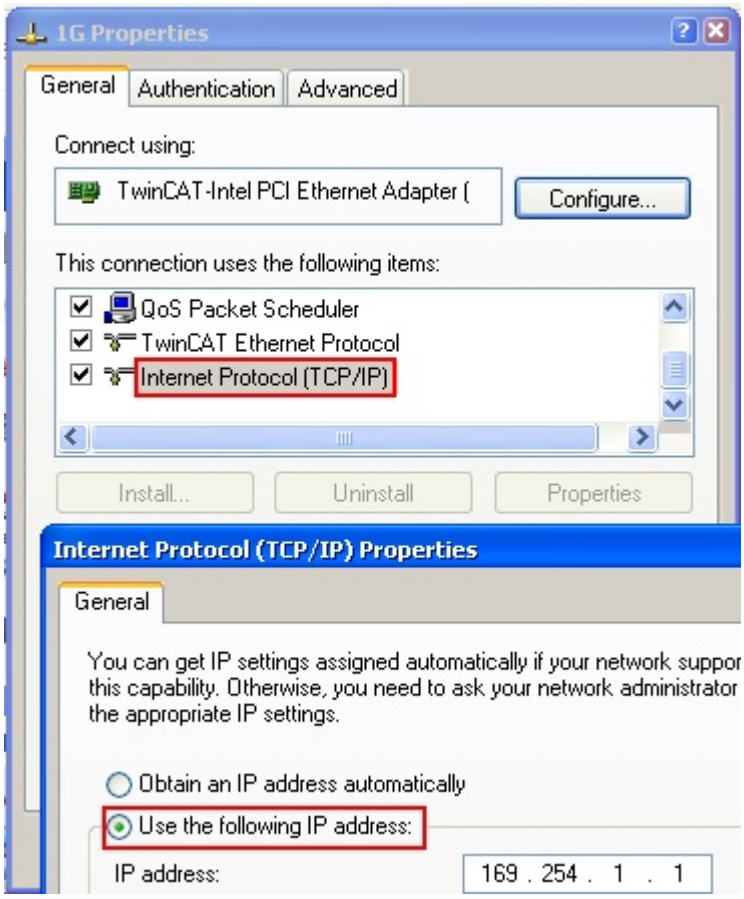


Fig. 33: TCP/IP setting for the Ethernet port

5.2.2 Notes regarding ESI device description

Installation of the latest ESI device description

The TwinCAT EtherCAT master/System Manager needs the device description files for the devices to be used in order to generate the configuration in online or offline mode. The device descriptions are contained in the so-called ESI files (EtherCAT Slave Information) in XML format. These files can be requested from the respective manufacturer and are made available for download. An *.xml file may contain several device descriptions.

The ESI files for Beckhoff EtherCAT devices are available on the Beckhoff website.

The ESI files should be stored in the TwinCAT installation directory (default TwinCAT2: C:\TwinCAT\IO \EtherCAT). The files are read (once) when a new System Manager window is opened, if they have changed since the last time the System Manager window was opened.

A TwinCAT installation includes the set of Beckhoff ESI files that was current at the time when the TwinCAT build was created.

For TwinCAT 2.11/TwinCAT 3 and higher, the ESI directory can be updated from the System Manager, if the programming PC is connected to the Internet (Option -> “Update EtherCAT Device Descriptions”)

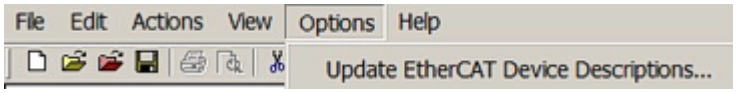


Fig. 34: For TwinCAT 2.11 and higher, the System Manager can search for current Beckhoff ESI files automatically, if an online connection is available

Note	**ESI** The *.xml files are associated with *.xsd files, which describe the structure of the ESI XML files. To update the ESI device descriptions, both file types should therefore be updated.

Device differentiation

EtherCAT devices/slaves are distinguished by 4 properties, which determine the full device identifier. The EL2521-0025-1018 ID consists of

- family key “EL”
- name “2521”
- type “0025”
- and revision “1018”

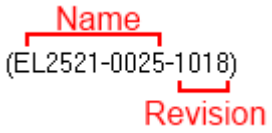


Fig. 35: Identifier structure

The order identifier consisting of name + type (here: EL2521-0010) describes the device function. The revision indicates the technical progress 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. Each revision has its own ESI description. See further notes [▶ 6].

Online description

If the EtherCAT configuration is created online through scanning of real devices (see section Online setup) and no ESI descriptions are available for a slave (specified by name and revision) that was found, the System Manager asks whether the description stored in the device should be used. In any case, the System Manager needs this information for setting up the cyclic and acyclic communication with the slave correctly.

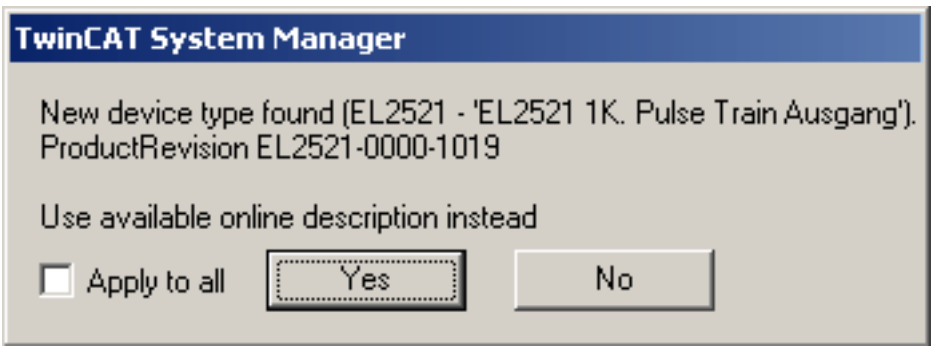


Fig. 36: OnlineDescription information window

In TwinCAT 3.x a similar window appears, which also offers the Web update:

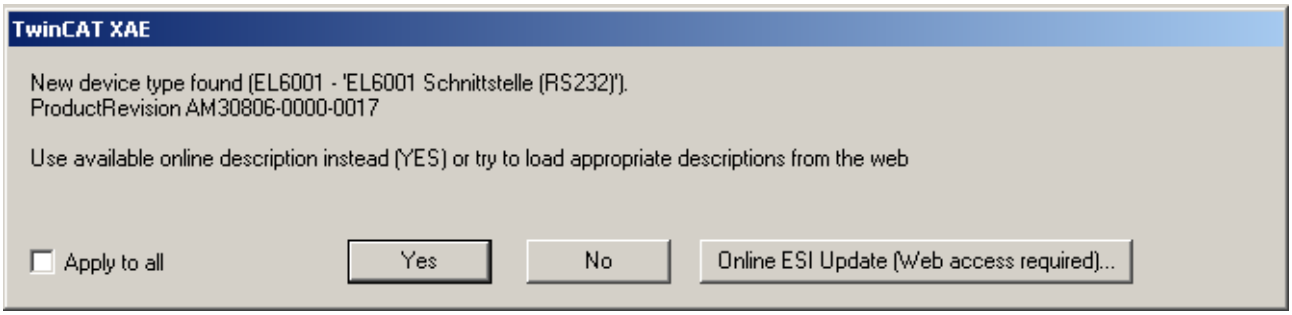


Fig. 37: Information window OnlineDescription, TwinCAT 3.x

If possible, the *Yes* is to be rejected and the required ESI is to be requested from the device manufacturer. After installation of the XML/XSD file the configuration process should be repeated.

Attention	**Changing the 'usual' configuration through a scan** ✓ If a scan discovers a device that is not yet known to TwinCAT, distinction has to be made between two cases. Taking the example here of the EL2521-0000 in the revision 1019 a) no ESI is present for the EL2521-0000 device at all, either for the revision 1019 or for an older revision. The ESI must then be requested from the manufacturer (in this case Beckhoff). b) an ESI is present for the EL2521-0000 device, but only in an older revision, e.g. 1018 or 1017. In this case an in-house check should first be performed to determine whether the spare parts stock allows the integration of the increased revision into the configuration at all. A new/higher revision usually also brings along new features. If these are not to be used, work can continue without reservations with the previous revision 1018 in the configuration. This is also stated by the Beckhoff compatibility rule.

Refer in particular to the chapter 'General notes on the use of Beckhoff EtherCAT IO components' and for manual configuration to the chapter 'Configuration creation – manual'

If the OnlineDescription is used regardless, the System Manager reads a copy of the device description from the EEPROM in the EtherCAT slave. In complex slaves the size of the EEPROM may not be sufficient for the complete ESI, in which case the ESI would be incomplete in the configurator. The route via the ESI files is therefore recommended.

The System Manager creates a new file "OnlineDescription0000...xml" its ESI directory, which contains all ESI descriptions that were read online.

OnlineDescriptionCache00000002.xml

Fig. 38: File OnlineDescription.xml created by the System Manager

If slaves are added manually to the configuration at a later stage, slaves created in the manner described above are indicated by an arrow, see Fig. *"Arrow indicates ESI recorded from OnlineDescription"*, EL2521.

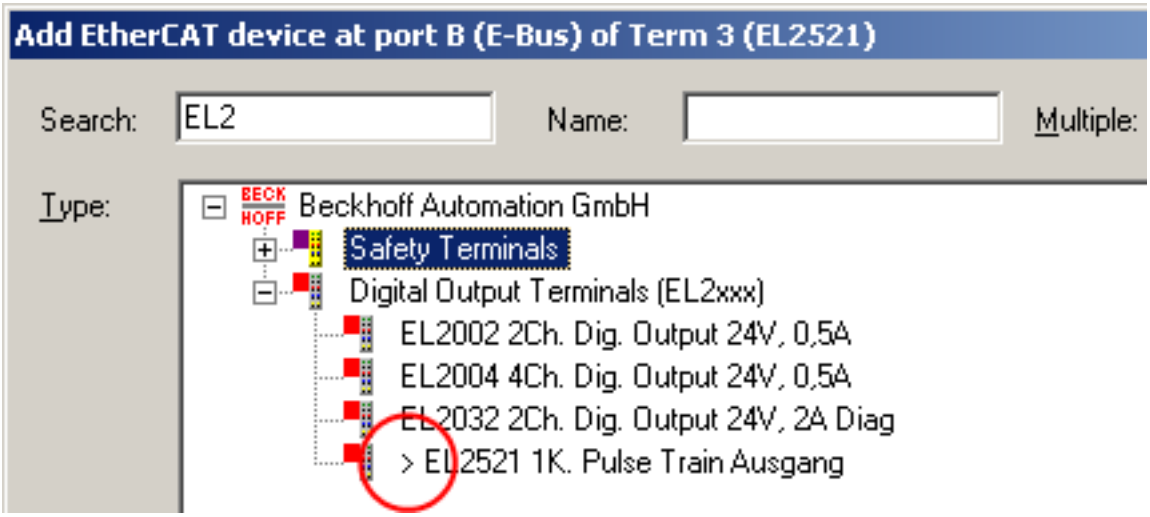


Fig. 39: Arrow indicates ESI recorded from OnlineDescription

If such ESI files are used and the manufacturer's files become available later, the file OnlineDescription.xml should be deleted as follows:

- close all System Manager windows
- restart TwinCAT in Config mode
- delete "OnlineDescription0000...xml"
- restart TwinCAt System Manager

This file should not be visible after this procedure, if necessary press <F5> to update

Note	**OnlineDescription for TwinCAT 3.x** In addition to the file described above "OnlineDescription0000...xml" , a so called EtherCAT cache with new discovered devices is created by TwinCAT 3.x (e.g. under Windows 7)C:\User\[USERNAME]\AppData\Roaming\Beckhoff\TwinCAT3\Components\Base\EtherCAT-Cache.xml (Please note the language settings of the OS!)You have to delete this file, too.

Faulty ESI file

If an ESI file is faulty and the System Manager is unable to read it, the System Manager brings up an information window.

Fig. 40: Information window for faulty ESI file

Reasons may include:

- Structure of the *.xml does not correspond to the associated *.xsd file --> check your schematics
- Contents cannot be translated into a device description --> contact the file manufacturer

5.2.3 Offline configuration creation (master: TwinCAT 2.x)

Distinction between Online and Offline

The distinction between online and offline refers to the presence of the actual I/O environment (drives, terminals). If the configuration is to be prepared in advance of the system configuration as a programming system, e.g. on a laptop, this is only possible in “Offline configuration” mode. In this case all components have to be entered manually in the configuration, e.g. based on the electrical design.

If the designed control system is already connected to the EtherCAT system and all components are energised and the infrastructure is ready for operation, the TwinCAT configuration can simply be generated through “scanning” from the runtime system. This is referred to as online configuration.

In any case, during each startup the EtherCAT master checks whether the slaves it finds match the configuration. This test can be parameterised in the extended slave settings.

Note	**Installation of the latest ESI-XML device description** The TwinCAT EtherCAT master/System Manager needs the device description files for the devices to be used in order to generate the configuration in online or offline mode. The device descriptions are contained in the so-called ESI files (EtherCAT Slave Information) in XML format. These files can be requested from the respective manufacturer and are made available for download. The ESIs for Beckhoff EtherCAT devices are provided on the Beckhoff website. The ESI files should be saved in the TwinCAT installation directory (default: C:\TwinCAT\IO\EtherCAT). The files are read (once) when a new System Manager window is opened.A TwinCAT installation includes the set of Beckhoff ESI files that was current at the time when the TwinCAT build was created.

For TwinCAT 2.11 and higher, the ESI directory can be updated from the System Manager, if the programming PC is connected to the Internet (Option -> “Update EtherCAT Device Descriptions”)

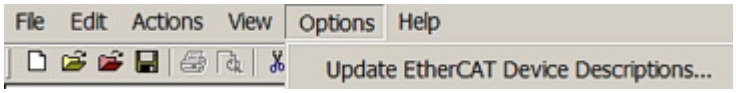


Fig. 41: Updating of the ESI directory

The following conditions must be met before a configuration can be set up:

- the EtherCAT device must be created/defined in the System Manager [▶ 33]
- the EtherCAT slaves must be defined [▶ 35]

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.

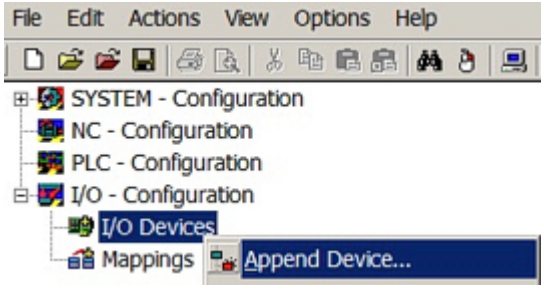


Fig. 42: Append EtherCAT device

Select type ‘EtherCAT’ for an EtherCAT I/O application with EtherCAT slaves. For the present publisher/subscriber service in combination with an EL6601/EL6614 terminal select “EtherCAT Automation Protocol via EL6601”.

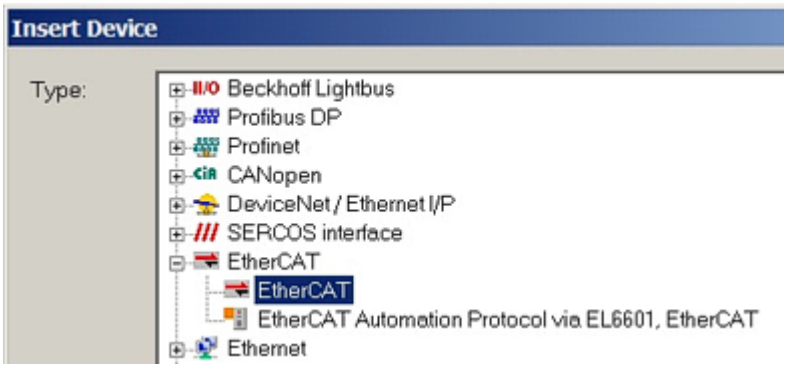


Fig. 43: Selecting the EtherCAT connection (TwinCAT 2.11)

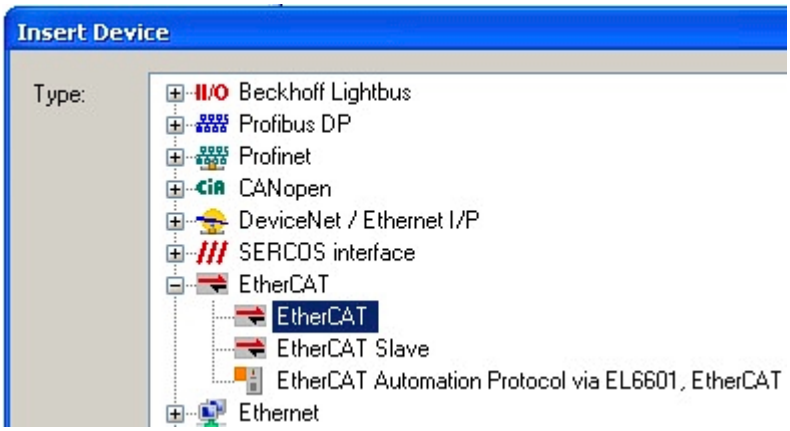


Fig. 44: Selecting the EtherCAT connection (TwinCAT 2.11 R2)

Then assign a real Ethernet port to this virtual device in the runtime system.

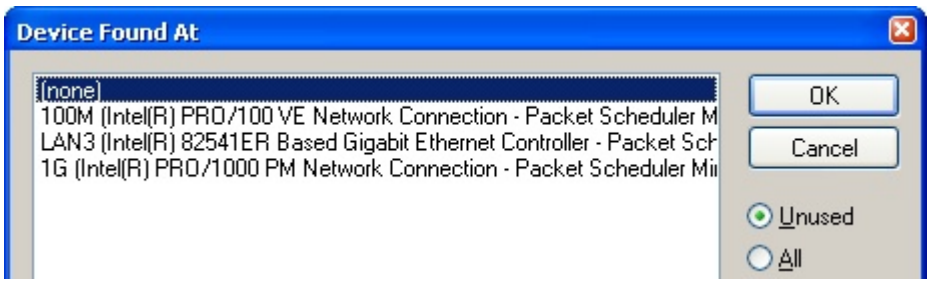


Fig. 45: Selecting the Ethernet port

This query may appear automatically when the EtherCAT device is created, or the assignment can be set/ modified later in the properties dialog (see Fig. *"EtherCAT properties dialog"*).

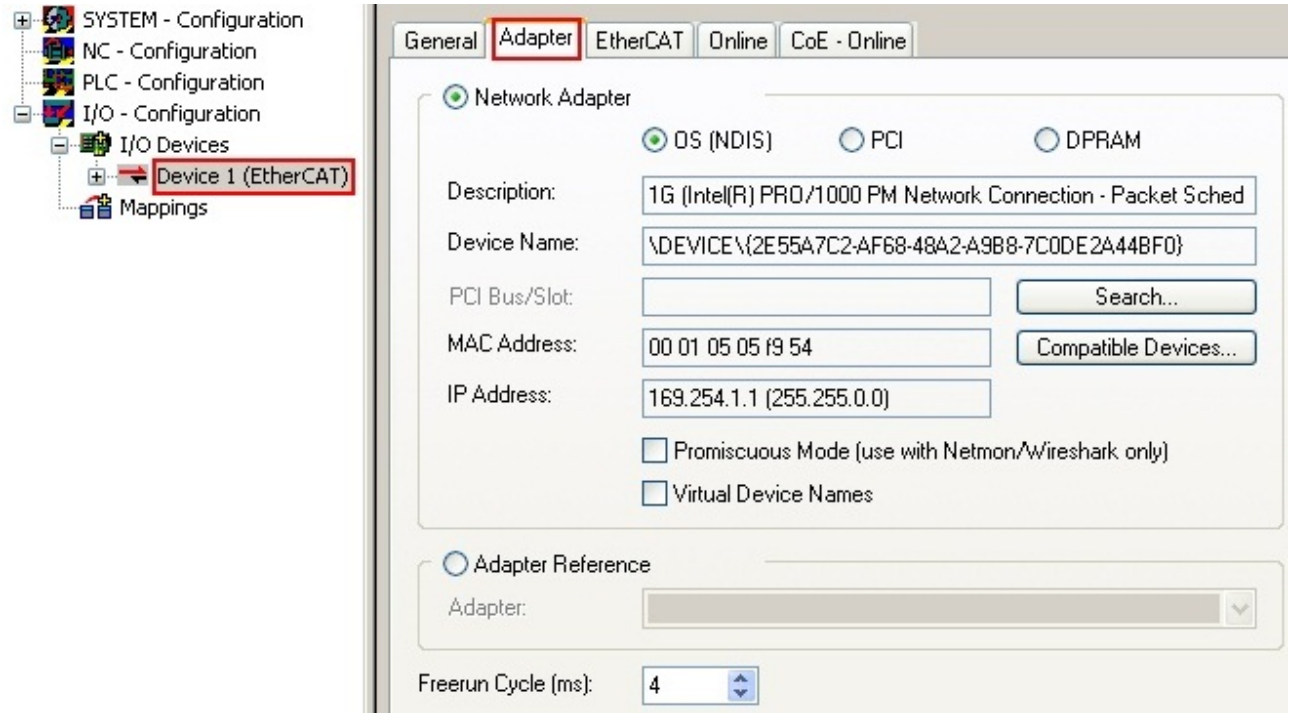


Fig. 46: EtherCAT properties dialog

Note	**Selecting the Ethernet port** Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective installation page.

Defining EtherCAT slaves

Further devices can be appended by right-clicking on a device in the configuration tree.

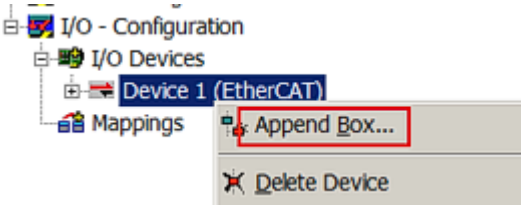


Fig. 47: Appending EtherCAT devices

The dialog for selecting a new device opens. Only devices for which ESI files are available are displayed.

Only devices are offered for selection that can be appended to the previously selected device. Therefore the physical layer available for this port is also displayed (Fig. *“Selection dialog for new EtherCAT device”*, A). In the case of cable-based Fast-Ethernet physical layer with PHY transfer, then also only cable-based devices are available, as shown in Fig. *“Selection dialog for new EtherCAT device”*. If the preceding device has several free ports (e.g. EK1122 or EK1100), the required port can be selected on the right-hand side (A).

Overview of physical layer

- “Ethernet”: cable-based 100BASE-TX: EK couplers, EP boxes, devices with RJ45/M8/M12 connector
- “E-Bus”: LVDS “terminal bus”: EL/ES terminals, various modular modules

The search field facilitates finding specific devices (TwinCAT 2.11 or higher).

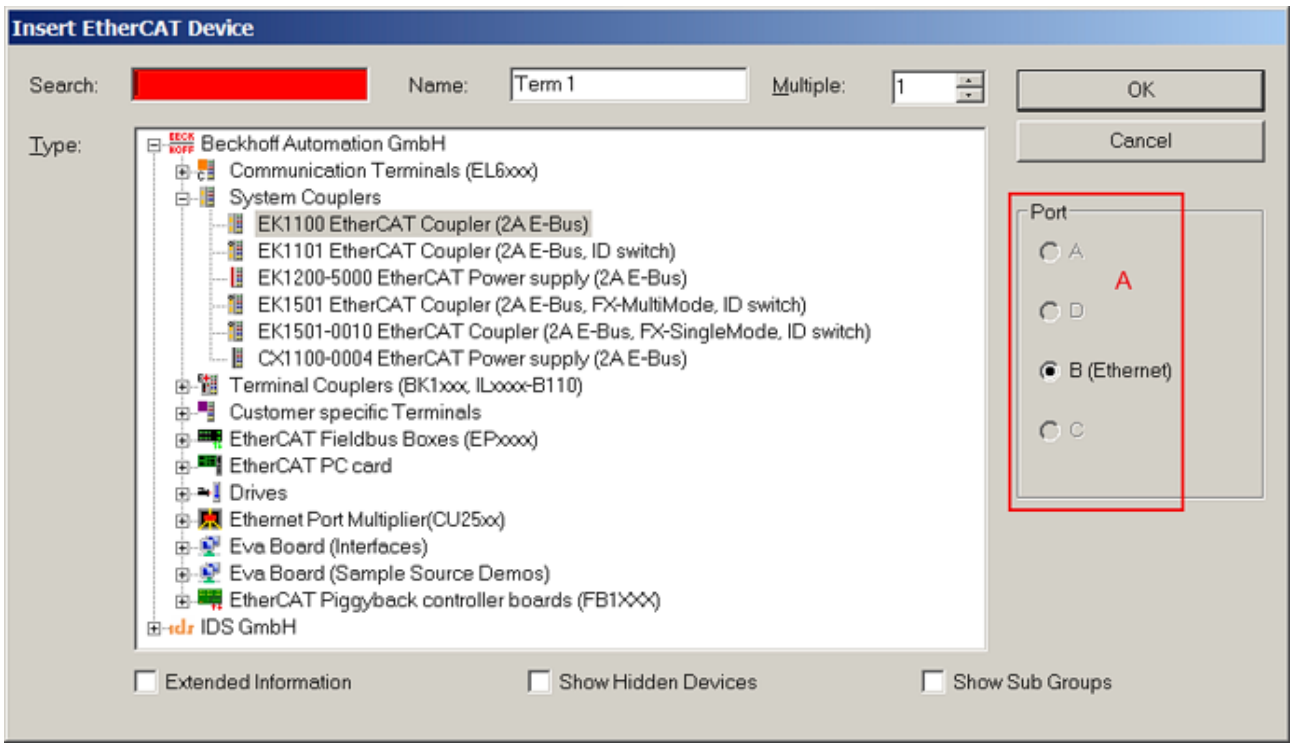


Fig. 48: Selection dialog for new EtherCAT device

By default only the name/device type is used as selection criterion. For selecting a specific revision of the device the revision can be displayed as “Extended Information”.

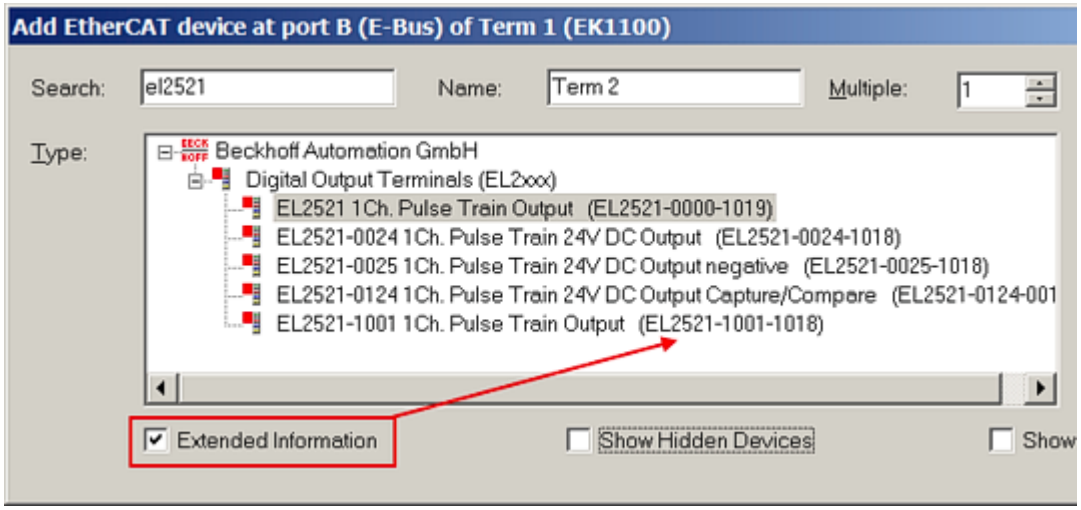


Fig. 49: Display of device revision

In many cases several device revisions were created for historic or functional reasons, e.g. through technological advancement. For simplification purposes (see Fig. *“Selection dialog for new EtherCAT device”*) only the last (i.e. highest) revision and therefore the latest state of production is displayed in the selection dialog for Beckhoff devices. To show all device revisions available in the system as ESI descriptions tick the “Show Hidden Devices” check box, see Fig. *“Display of previous revisions”*.

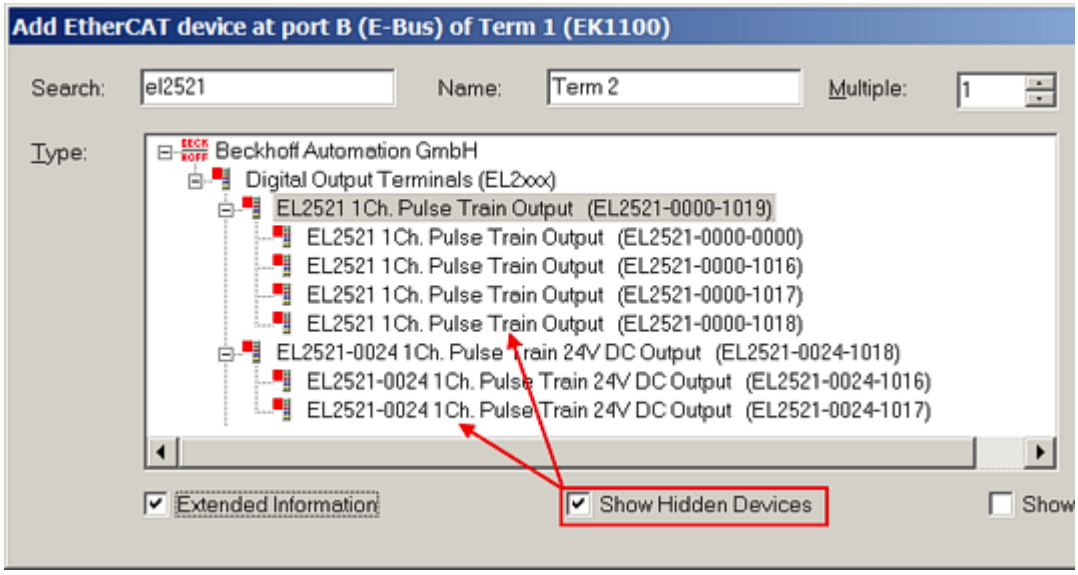


Fig. 50: Display of previous revisions

Note	**Device selection based on revision, compatibility** The ESI description also defines the process image, the communication type between master and slave/device and the device functions, if applicable. The physical device (firmware, if available) has to support the communication queries/settings of the master. This is backward compatible, i.e. newer devices (higher revision) should be supported if the EtherCAT master addresses them as an older revision. The following compatibility rule of thumb is to be assumed for Beckhoff EtherCAT Terminals/Boxes: **device revision in the system >= device revision in the configuration** This also enables subsequent replacement of devices without changing the configuration (different specifications are possible for drives).

Example:

If an EL2521-0025-**1018** is specified in the configuration, an EL2521-0025-**1018** or higher (-**1019**, -**1020**) can be used in practice.

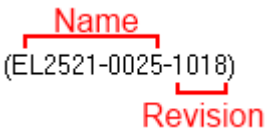


Fig. 51: Name/revision of the terminal

If current ESI descriptions are available in the TwinCAT system, the last revision offered in the selection dialog matches the Beckhoff state of production. It is recommended to use the last device revision when creating a new configuration, if current Beckhoff devices are used in the real application. Older revisions should only be used if older devices from stock are to be used in the application.

In this case the process image of the device is shown in the configuration tree and can be parameterised as follows: linking with the task, CoE/DC settings, plug-in definition, startup settings, ...

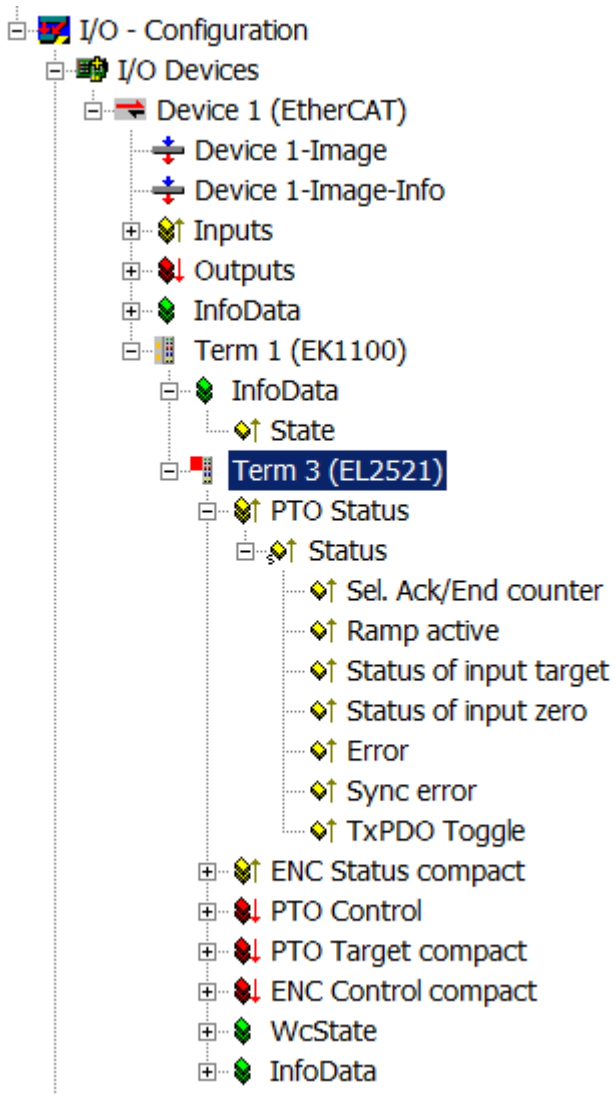


Fig. 52: EtherCAT terminal in the TwinCAT tree

5.2.4 Online configuration creation ‘scanning’ (master: TwinCAT 2.x)

Distinction between Online and Offline

Distinction between Online and Offline

The distinction between online and offline refers to the presence of the actual I/O environment (drives, terminals). If the configuration is to be prepared in advance of the system configuration as a programming system, e.g. on a laptop, this is only possible in “Offline configuration” mode. In this case all components have to be entered manually in the configuration, e.g. based on the electrical design.

If the designed control system is already connected to the EtherCAT system and all components are energised and the infrastructure is ready for operation, the TwinCAT configuration can simply be generated through “scanning” from the runtime system. This is referred to as online configuration.

In any case, during each startup the EtherCAT master checks whether the slaves it finds match the configuration. This test can be parameterised in the extended slave settings.

Note	**Installation of the latest ESI-XML device description** The TwinCAT EtherCAT master/System Manager needs the device description files for the devices to be used in order to generate the configuration in online or offline mode. The device descriptions are contained in the so-called ESI files (EtherCAT Slave Information) in XML format. These files can be requested from the respective manufacturer and are made available for download. The ESIs for Beckhoff EtherCAT devices are provided on the Beckhoff website. The ESI files should be saved in the TwinCAT installation directory (default: C:\TwinCAT\IO\EtherCAT). The files are read (once) when a new System Manager window is opened.A TwinCAT installation includes the set of Beckhoff ESI files that was current at the time when the TwinCAT build was created.

For TwinCAT 2.11 and higher, the ESI directory can be updated from the System Manager, if the programming PC is connected to the Internet (Option -> “Update EtherCAT Device Descriptions”)

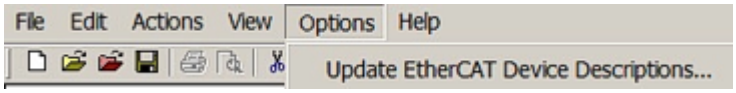


Fig. 53: Updating ESI directory

The following conditions must be met before a configuration can be set up:

- the real EtherCAT hardware (devices, couplers, drives) must be present and installed
- the devices/modules must be connected via EtherCAT cables or in the terminal strand in the same way as they are intended to be used later
- the devices/modules be connected to the power supply and ready for communication
- TwinCAT must be in CONFIG mode on the target system.

The online scan process consists of:

- detecting the EtherCAT device [▶ 39] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 41]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 44]

The scan with existing configuration [▶ 45] can also be carried out for comparison.

Detecting/scanning of the EtherCAT device

The online device search can be used if the TwinCAT system is in CONFIG mode (blue TwinCAT icon or blue indication in the System Manager).

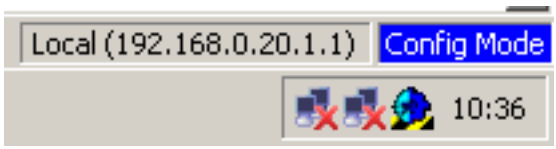


Fig. 54: TwinCAT CONFIG mode display

Note	**Online scanning in Config mode** The online search is not available in RUN mode (production operation). Note the differentiation between TwinCAT programming system and TwinCAT target system.

The TwinCAT icon next to the Windows clock always shows the TwinCAT mode of the local IPC. The System Manager window shows the TwinCAT state of the target system.

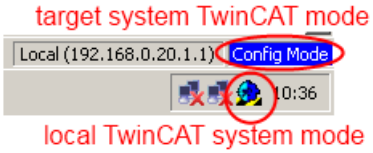


Fig. 55: Differentiation local/target system

Right-clicking on “I/O Devices” in the configuration tree opens the search dialog.

Fig. 56: Scan Devices

This scan mode attempts to find not only EtherCAT devices (or Ethernet ports that are usable as such), but also NOVRAM, fieldbus cards, SMB etc. However, not all devices can be found automatically.

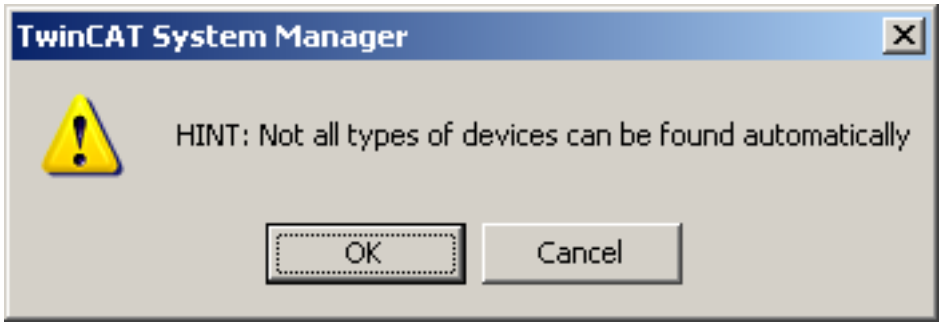


Fig. 57: Note for automatic device scan

Ethernet ports with installed TwinCAT real-time driver are shown as “RT Ethernet” devices. An EtherCAT frame is sent to these ports for testing purposes. If the scan agent detects from the response that an EtherCAT slave is connected, the port is immediately shown as an “EtherCAT Device” .

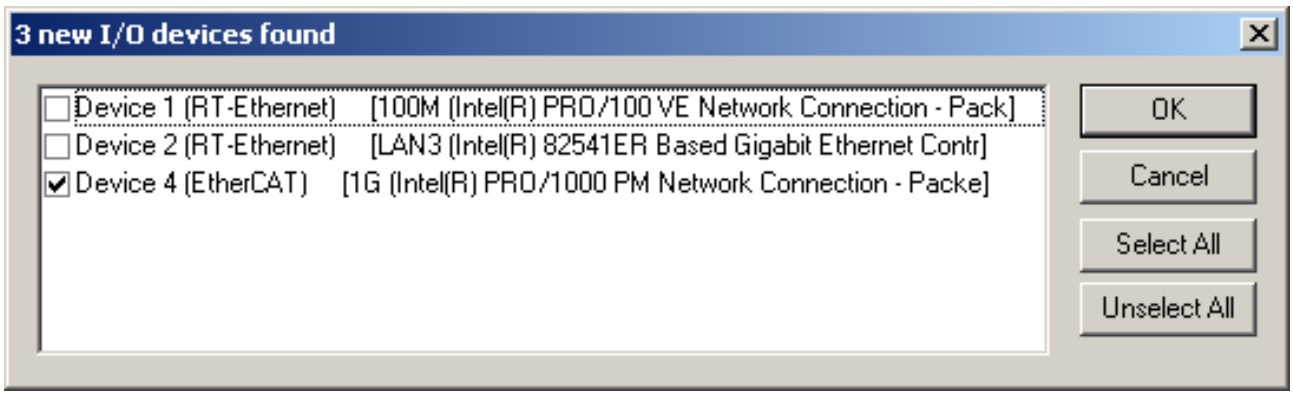


Fig. 58: Detected Ethernet devices

After confirmation with “OK” a device scan is suggested for all selected devices, see Fig. 5.

Note	**Selecting the Ethernet port** Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective installation page [▶ 26].

Detecting/Scanning the EtherCAT devices

Note	**Online scan functionality** During a scan the master queries the identity information of the EtherCAT slaves from the slave EEPROM. The name and revision are used for determining the type. The respective devices are located in the stored ESI data and integrated in the configuration tree in the default state defined there.

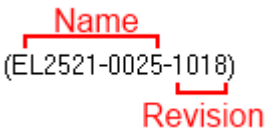


Fig. 59: Example default state

Attention	**Slave scanning in practice in series machine production** The scanning function should be used with care. It is a practical and fast tool for creating an initial configuration as a basis for commissioning. In series machine production or reproduction of the plant, however, the function should no longer be used for the creation of the configuration, but if necessary for comparison [▶ 45] with the defined initial configuration.Background: since Beckhoff occasionally increases the revision version of the delivered products for product maintenance reasons, a configuration can be created by such a scan which (with an identical machine construction) is identical according to the device list; however, the respective device revision may differ from the initial configuration.

Example:

Company A builds the prototype of a machine B, which is to be produced in series later on. To do this the prototype is built, a scan of the IO devices is performed in TwinCAT and the initial configuration ‘B.tsm’ is created. The EL2521-0025 EtherCAT terminal with the revision 1018 is located somewhere. It is thus built into the TwinCAT configuration in this way:

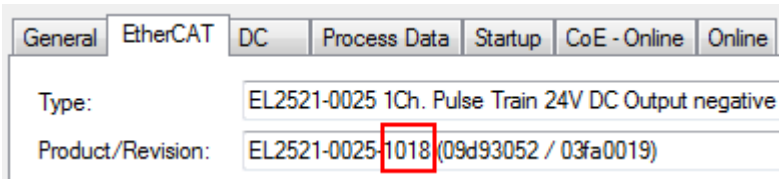


Fig. 60: Installing EthetCAT terminal with revision -1018

Likewise, during the prototype test phase, the functions and properties of this terminal are tested by the programmers/commissioning engineers and used if necessary, i.e. addressed from the PLC ‘B.pro’ or the NC. (the same applies correspondingly to the TwinCAT3 solution files).

The prototype development is now completed and series production of machine B starts, for which Beckhoff continues to supply the EL2521-0025-0018. If the commissioning engineers of the series machine production department always carry out a scan, a B configuration with the identical contents results again for each machine. Likewise, A might create spare parts stores worldwide for the coming series-produced machines with EL2521-0025-1018 terminals.

After some time Beckhoff extends the EL2521-0025 by a new feature C. Therefore the FW is changed, outwardly recognizable by a higher FW version and **a new revision -1019**. Nevertheless the new device naturally supports functions and interfaces of the predecessor version(s); an adaptation of ‘B.tsm’ or even ‘B.pro’ is therefore unnecessary. The series-produced machines can continue to be built with ‘B.tsm’ and ‘B.pro’; it makes sense to perform a comparative scan [▶ 45] against the initial configuration ‘B.tsm’ in order to check the built machine.

However, if the series machine production department now doesn’t use ‘B.tsm’, but instead carries out a scan to create the productive configuration, the revision **-1019** is automatically detected and built into the configuration:

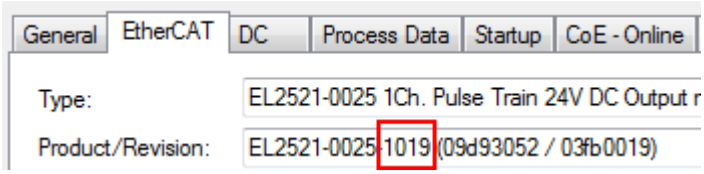


Fig. 61: Detection of EtherCAT terminal with revision -1019

This is usually not noticed by the commissioning engineers. TwinCAT cannot signal anything either, since virtually a new configuration is created. According to the compatibility rule, however, this means that no EL2521-0025-**1018** should be built into this machine as a spare part (even if this nevertheless works in the vast majority of cases).

In addition, it could be the case that, due to the development accompanying production in company A, the new feature C of the EL2521-0025-1019 (for example, an improved analog filter or an additional process data for the diagnosis) is discovered and used without in-house consultation. The previous stock of spare part devices are then no longer to be used for the new configuration ‘B2.tsm’ created in this way.Þ if series machine production is established, the scan should only be performed for informative purposes for comparison with a defined initial configuration. Changes are to be made with care!

If an EtherCAT device was created in the configuration (manually or through a scan), the I/O field can be scanned for devices/slaves.

Fig. 62: Scan query after automatic creation of an EtherCAT device

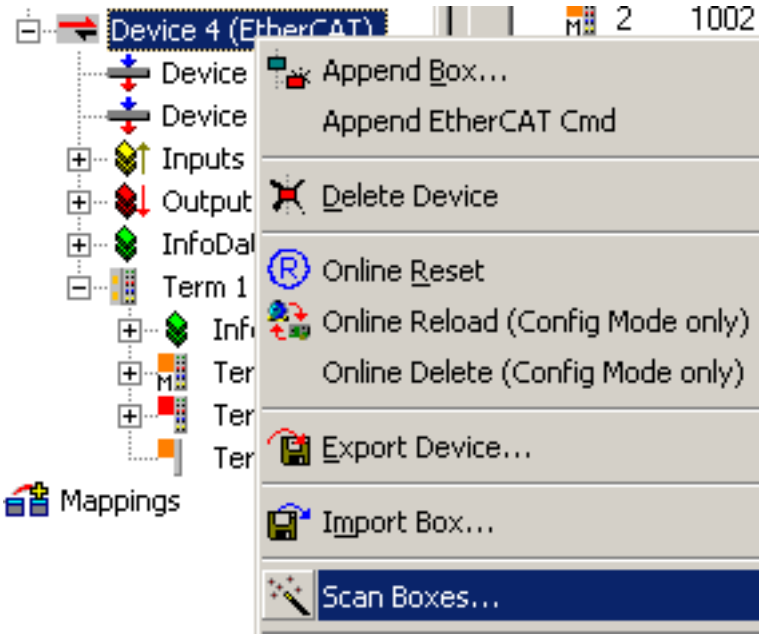


Fig. 63: Manual triggering of a device scan on a specified EtherCAT device

In the System Manager the scan process can be monitored via the progress bar at the bottom of the screen.

Fig. 64: Scan progress

The configuration is established and can then be switched to online state (OPERATIONAL).

Fig. 65: Config/FreeRun query

In Config/FreeRun mode the System Manager display alternates between blue and red, and the EtherCAT device continues to operate with the idling cycle time of 4 ms (default setting), even without active task (NC, PLC).

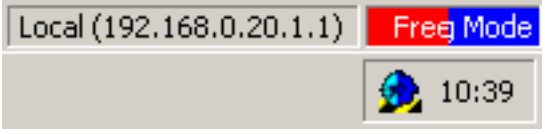


Fig. 66: Config/FreeRun indicator

Fig. 67: TwinCAT can also be switched to this state by using a button

The EtherCAT system should then be in a functional cyclic state, as shown in Fig. *"Online display example"*.

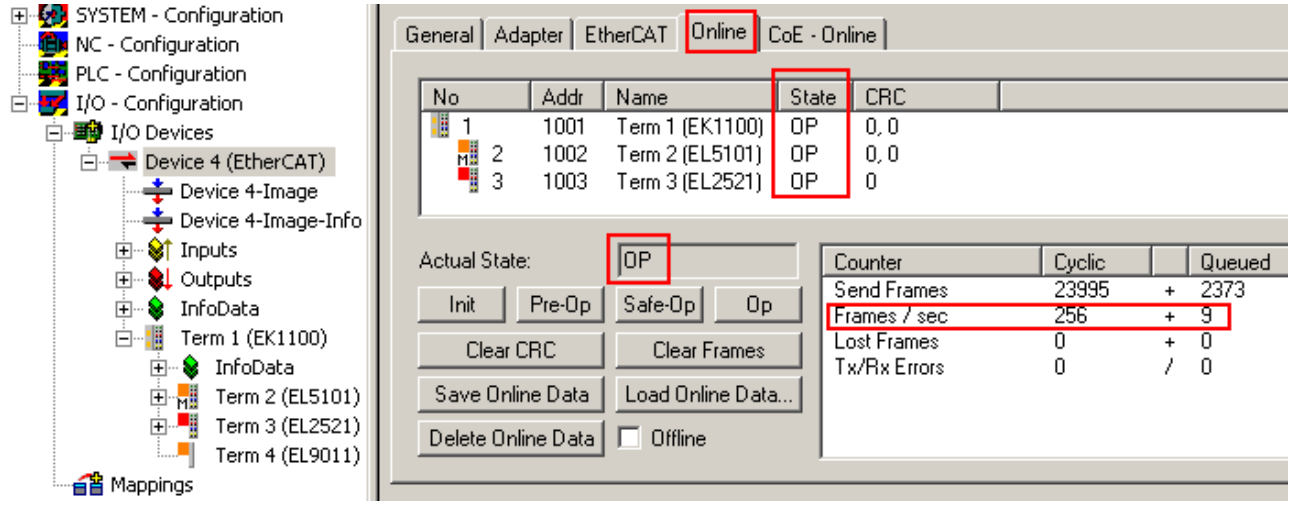


Fig. 68: Online display example

Please note:

- all slaves should be in OP state
- the EtherCAT master should be in "Actual State" OP
- "frames/sec" should match the cycle time taking into account the sent number of frames
- no excessive "LostFrames" or CRC errors should occur

The configuration is now complete. It can be modified as described under manual procedure [▶ 33].

Troubleshooting

Various effects may occur during scanning.

- An **unknown device** is detected, i.e. an EtherCAT slave for which no ESI XML description is available. In this case the System Manager offers to read any ESI that may be stored in the device. This case is described in the chapter "Notes regarding ESI device description".
- **Device are not detected properly**
  Possible reasons include:
  - faulty data links, resulting in data loss during the scan
  - slave has invalid device description
  The connections and devices should be checked in a targeted manner, e.g. via the emergency scan. Then re-run the scan.

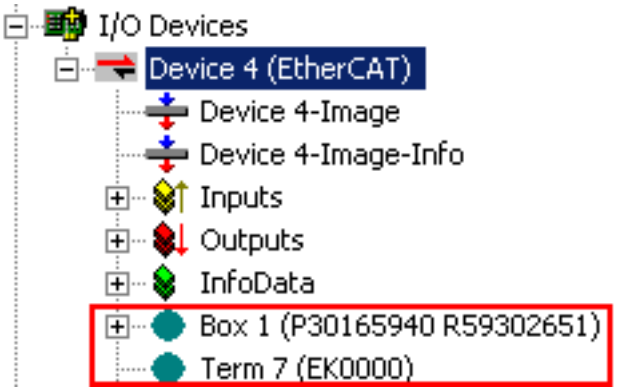


Fig. 69: Faulty identification

In the System Manager such devices may be set up as EK0000 or unknown devices. Operation is not possible or meaningful.

Scan over existing Configuration

Attention	**Change of the configuration after comparison** With this scan (TwinCAT 2.11 or 3.1) only the device properties vendor (manufacturer), device name and revision are compared at present! A ‘ChangeTo’ or ‘Copy’ should only be carried out with care, taking into consideration the Beckhoff IO compatibility rule (see above). The device configuration is then replaced by the revision found; this can affect the supported process data and functions.

If a scan is initiated for an existing configuration, the actual I/O environment may match the configuration exactly or it may differ. This enables the configuration to be compared.

Fig. 70: Identical configuration

If differences are detected, they are shown in the correction dialog, so that the user can modify the configuration as required.

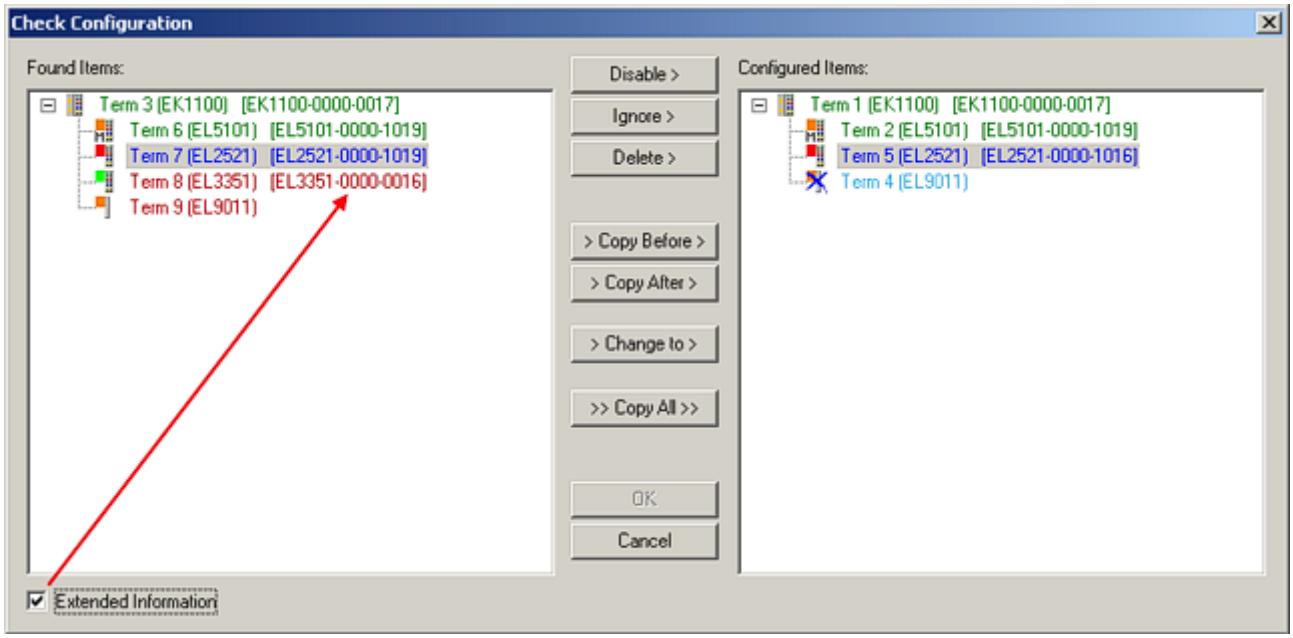


Fig. 71: Correction dialog

It is advisable to tick the “Extended Information” check box to reveal differences in the revision.

Colour	Explanation
green	This EtherCAT slave matches the entry on the other side. Both type and revision match.
blue	This EtherCAT slave is present on the other side, but in a different revision. This other revision can have other default values for the process data as well as other/additional functions. If the found revision is higher than the configured revision, the slave may be used provided compatibility issues are taken into account. If the found revision is lower than the configured revision, it is likely that the slave cannot be used. The found device may not support all functions that the master expects based on the higher revision number.
light blue	This EtherCAT slave is ignored (“Ignore” button)
red	• This EtherCAT slave is not present on the other side. • It is present, but in a different revision, which also differs in its properties from the one specified. The compatibility principle then also applies here: if the found revision is higher than the configured revision, use is possible provided compatibility issues are taken into account, since the successor devices should support the functions of the predecessor devices. If the found revision is lower than the configured revision, it is likely that the slave cannot be used. The found device may not support all functions that the master expects based on the higher revision number.

Note

Device selection based on revision, compatibility

The ESI description also defines the process image, the communication type between master and slave/device and the device functions, if applicable. The physical device (firmware, if available) has to support the communication queries/settings of the master. This is backward compatible, i.e. newer devices (higher revision) should be supported if the EtherCAT master addresses them as an older revision. The following compatibility rule of thumb is to be assumed for Beckhoff EtherCAT Terminals/Boxes:

device revision in the system >= device revision in the configuration

This also enables subsequent replacement of devices without changing the configuration (different specifications are possible for drives).

Example:

If an EL2521-0025-1018 is specified in the configuration, an EL2521-0025-1018 or higher (-1019, -1020) can be used in practice.

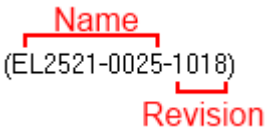


Fig. 72: Name/revision terminal

If current ESI descriptions are available in the TwinCAT system, the last revision offered in the selection dialog matches the Beckhoff state of production. It is recommended to use the last device revision when creating a new configuration, if current Beckhoff devices are used in the real application. Older revisions should only be used if older devices from stock are to be used in the application.

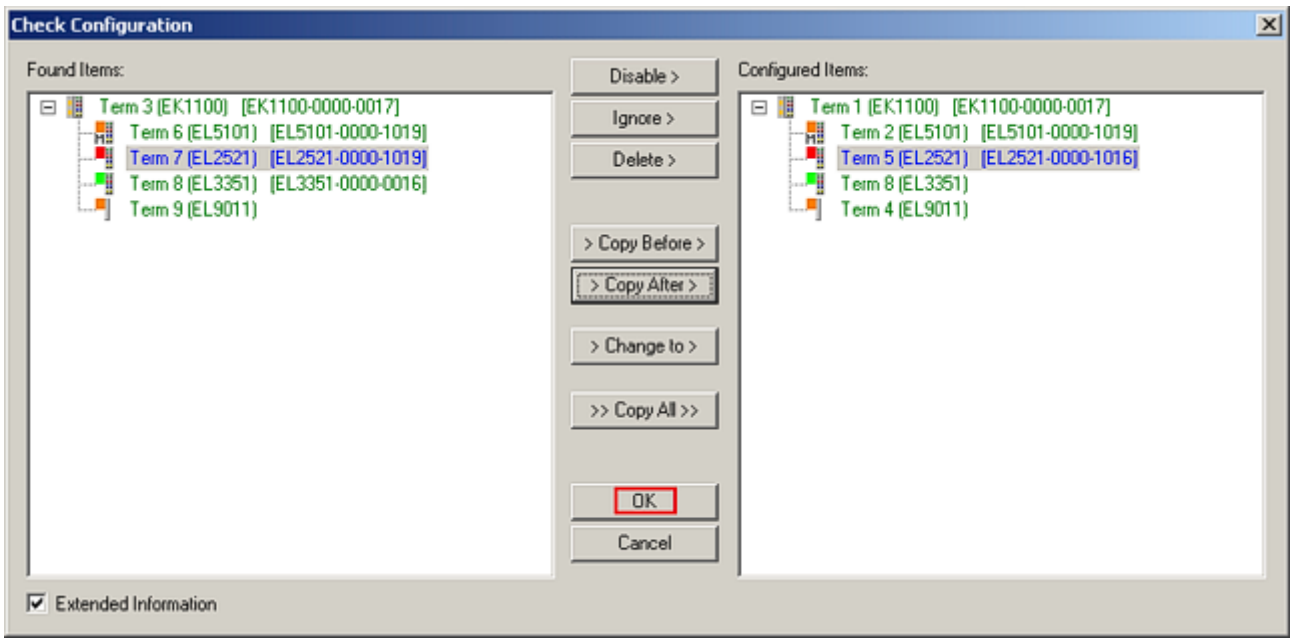


Fig. 73: Correction dialog with modifications

Once all modifications have been saved or accepted, click “OK” to transfer them to the real *.tsm configuration.

Change to compatible device

The TwinCAT System Manager offers a function for the exchange of a device whilst retaining the links in the task: *Change to compatible device.*

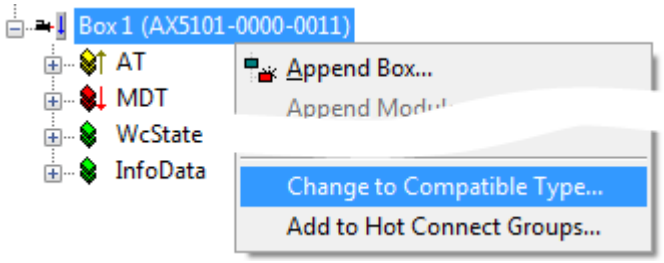


Fig. 74: TwinCAT 2 Dialog ChangeToCompatibleDevice

This function is preferably to be used on AX5000 devices. If called, the System Manager suggests the devices that it finds in the associated sub-folder; in the case of the AX5000, for example, in *\TwiNCAT\IO \EtherCAT\Beckhoff AX5xxx*.

Change to Alternative Type

The TwinCAT System Manager offers a function for the exchange of a device: *Change to Alternative Type*

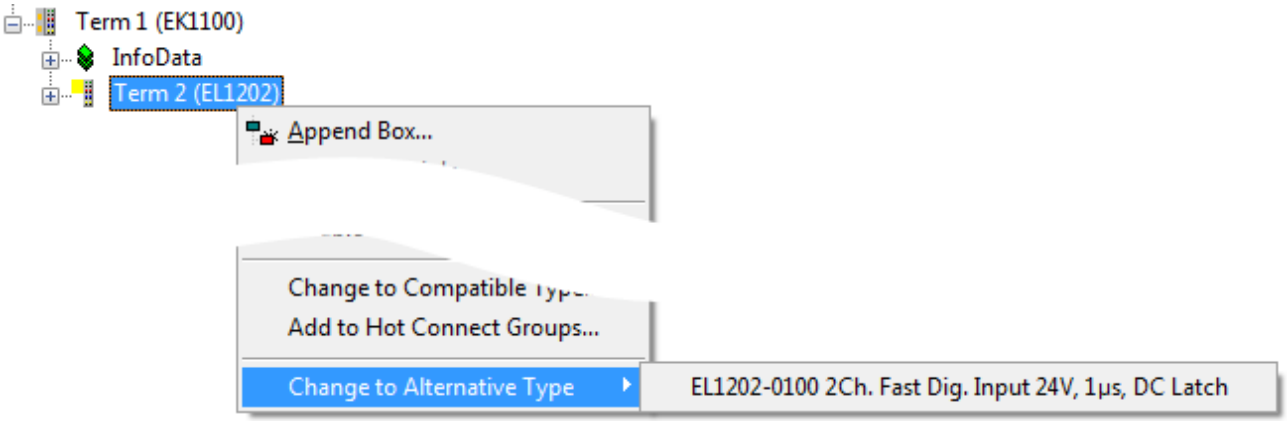


Fig. 75: TwinCAT 2 Dialog ChangeToCompatibleDevice

If called, the System Manager searches in the procured device ESI (in this example: EL1202-0000) for details of compatible devices contained there. The configuration is changed and the ESI-EEPROM is overwritten at the same time – therefore this process is possible only in the online state (ConfigMode).

5.2.5 Configuration by means of the TwinCAT System Manager

(with TwinCAT from version 2.10.0 (Build 1241), using EL5001 from firmware version 0.7 as an example)

In the left-hand window of the TwinCAT System Manager, click on the branch you wish to configure (in the example: EL5001 Terminal 6).

Fig. 76: Branch of EL5001

In the right-hand window of the TwinCAT System manager, various tabs are now available for configuring the terminal.

„General“ tab

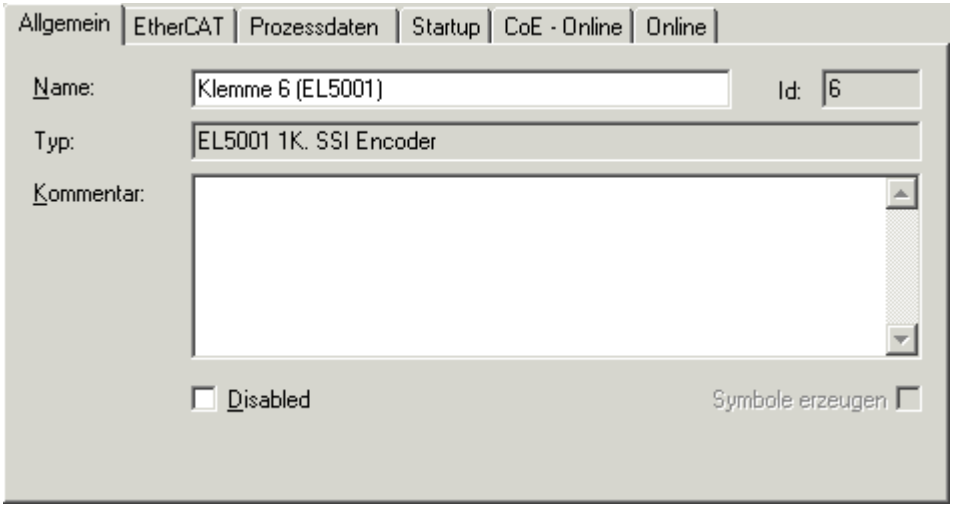


Fig. 77: “General” tab

Name	Name of the EtherCAT device
Id	Number of the EtherCAT device
Type	EtherCAT device type
Comment	Here you can add a comment (e.g. regarding the system).
Disabled	Here you can deactivate the EtherCAT device.
Create symbols	Access to this EtherCAT slave via ADS is only available if this control box is activated.

„EtherCAT“ tab

Fig. 78: „EtherCAT“ tab

Type	EtherCAT device type
Product/Revision	Product and revision number of the EtherCAT device
Auto Inc Addr.	Auto increment address of the EtherCAT device. The auto increment address can be used for addressing each EtherCAT device in the communication ring through its physical position. Auto increment addressing is used during the start-up phase when the EtherCAT master allocates addresses to the EtherCAT devices. With auto increment addressing the first EtherCAT slave in the ring has the address 0000_{hex}. For each further slave the address is decremented by 1 ($FFFF_{hex}$, $FFFE_{hex}$ etc.).
EtherCAT Addr.	Fixed address of an EtherCAT slave. This address is allocated by the EtherCAT master during the start-up phase. Tick the control box to the left of the input field in order to modify the default value.
Previous Port	Name and port of the EtherCAT device to which this device is connected. If it is possible to connect this device with another one without changing the order of the EtherCAT devices in the communication ring, then this combination field is activated and the EtherCAT device to which this device is to be connected can be selected.
Advanced Settings	This button opens the dialogs for advanced settings.

The link at the bottom of the tab points to the product page for this EtherCAT device on the web.

“Process Data” tab

Indicates the configuration of the process data. The input and output data of the EtherCAT slave are represented as CANopen process data objects (PDO). The user can select a PDO via PDO assignment and modify the content of the individual PDO via this dialog, if the EtherCAT slave supports this function.

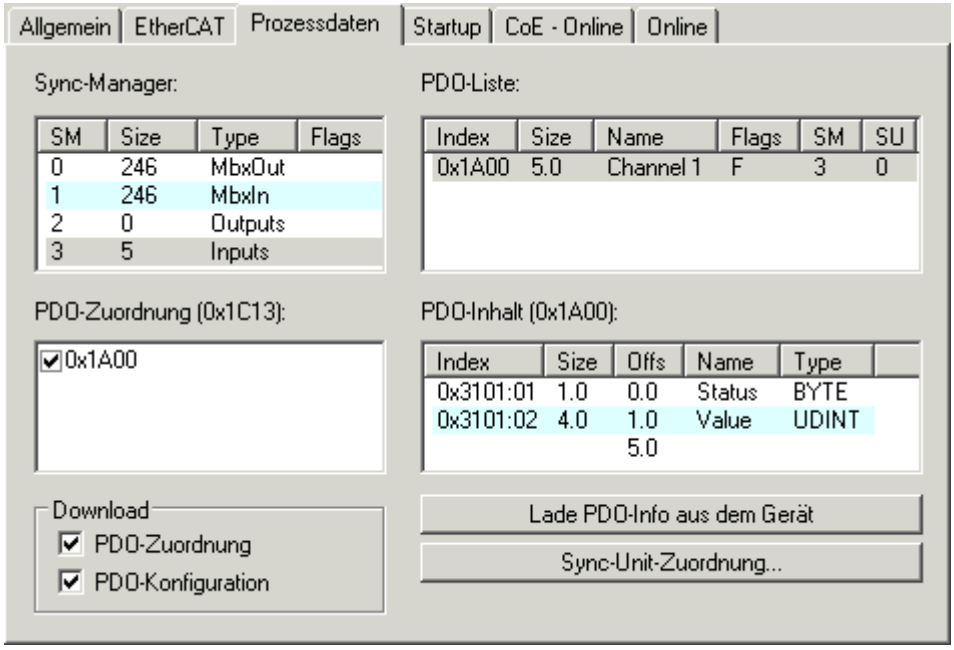


Fig. 79: “Process Data” tab

Sync Manager

Lists the configuration of the Sync Manager (SM).
If the EtherCAT device has a mailbox, SM0 is used for the mailbox output (MbxOut) and SM1 for the mailbox input (MbxIn).
SM2 is used for the output process data (outputs) and SM3 (inputs) for the input process data.

If an input is selected, the corresponding PDO assignment is displayed in the *PDO Assignment* list below.

PDO Assignment

PDO assignment of the selected Sync Manager. All PDOs defined for this Sync Manager type are listed here:

- If the output Sync Manager (outputs) is selected in the Sync Manager list, all RxPDOs are displayed.
- If the input Sync Manager (inputs) is selected in the Sync Manager list, all TxPDOs are displayed.

The selected entries are the PDOs involved in the process data transfer. In the tree diagram of the System Manager these PDOs are displayed as variables of the EtherCAT device. The name of the variable is identical to the *Name* parameter of the PDO, as displayed in the PDO list. If an entry in the PDO assignment list is deactivated (not selected and greyed out), this indicates that the input is excluded from the PDO assignment. In order to be able to select a greyed out PDO, the currently selected PDO has to be deselected first.

Note	**Activation of PDO assignment** ✓ If you have changed the PDO assignment, in order to activate the new PDO assignment, a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see Online tab [▶ 55]), b) and the System Manager has to reload the EtherCAT slaves (button)

PDO list

List of all PDOs supported by this EtherCAT device. The content of the selected PDOs is displayed in the *PDO Content* list. The PDO configuration can be modified by double-clicking on an entry.

Column	Description	
Index	PDO index.	
Size	Size of the PDO in bytes.	
Name	Name of the PDO. If this PDO is assigned to a Sync Manager, it appears as a variable of the slave with this parameter as the name.	
Flags	F	Fixed content: The content of this PDO is fixed and cannot be changed by the System Manager.
	M	Mandatory PDO. This PDO is mandatory and must therefore be assigned to a Sync Manager! Consequently, this PDO cannot be deleted from the *PDO Assignment* list
SM	Sync Manager to which this PDO is assigned. If this entry is empty, this PDO does not take part in the process data traffic.	
SU	Sync unit to which this PDO is assigned.	

PDO Content

Indicates the content of the PDO. If flag F (fixed content) of the PDO is not set the content can be modified.

Download

If the device is intelligent and has a mailbox, the configuration of the PDO and the PDO assignments can be downloaded to the device. This is an optional feature that is not supported by all EtherCAT slaves.

PDO Assignment

If this check box is selected, the PDO assignment that is configured in the PDO Assignment list is downloaded to the device on startup. The required commands to be sent to the device can be viewed in the Startup [▶ 51] tab.

PDO Configuration

If this check box is selected, the configuration of the respective PDOs (as shown in the PDO list and the PDO Content display) is downloaded to the EtherCAT slave.

„Startup“ tab

The *Startup* tab is displayed if the EtherCAT slave has a mailbox and supports the *CANopen over EtherCAT* (CoE) or *Servo drive over EtherCAT* protocol. This tab indicates which download requests are sent to the mailbox during startup. It is also possible to add new mailbox requests to the list display. The download requests are sent to the slave in the same order as they are shown in the list.

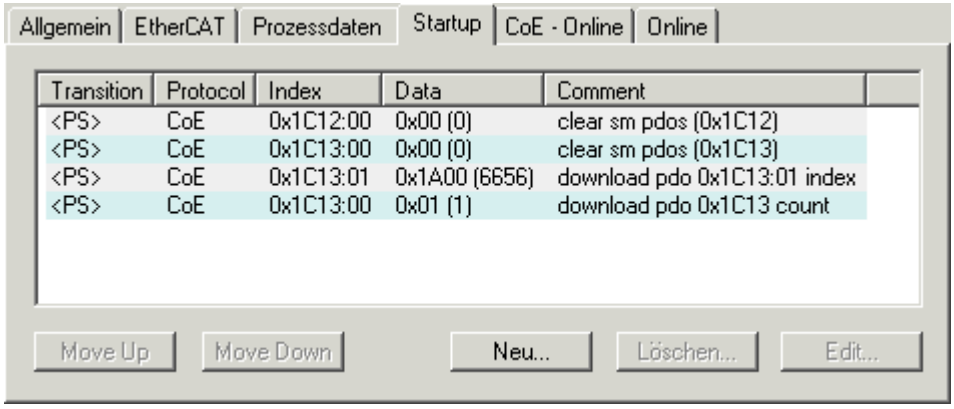


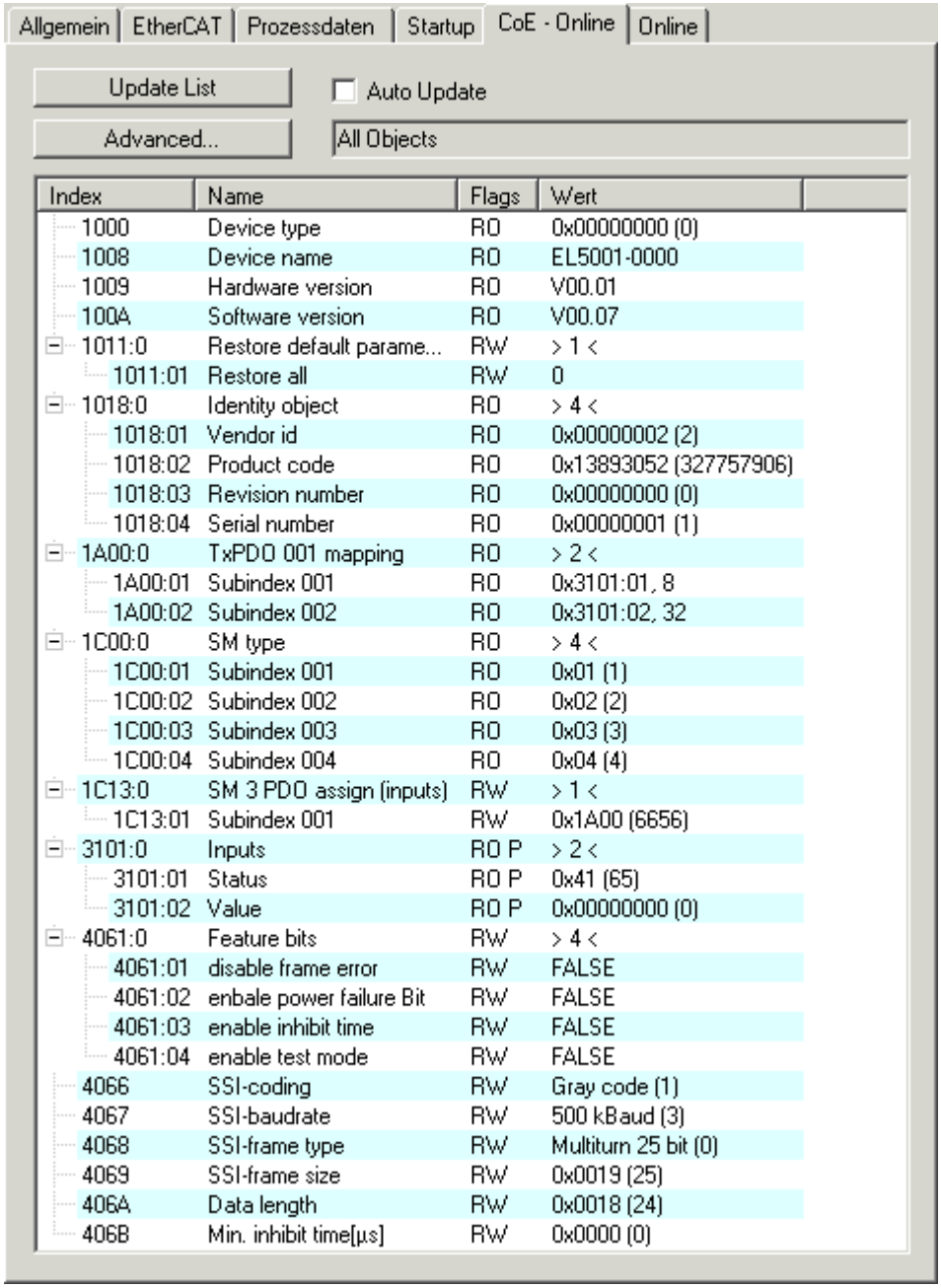
Fig. 80: „Startup“ tab

Column	Description
Transition	Transition to which the request is sent. This can either be • the transition from pre-operational to safe-operational (PS), or • the transition from safe-operational to operational (SO). If the transition is enclosed in "<>" (e.g. <PS>), the mailbox request is fixed and cannot be modified or deleted by the user.
Protocol	Type of mailbox protocol
Index	Index of the object
Data	Date on which this object is to be downloaded.
Comment	Description of the request to be sent to the mailbox

Move Up	This button moves the selected request up by one position in the list.
Move Down	This button moves the selected request down by one position in the list.
New	This button adds a new mailbox download request to be sent during startup.
Delete	This button deletes the selected entry.
Edit	This button edits an existing request.

“CoE – Online” tab

The additional *CoE - Online* tab is displayed if the EtherCAT slave supports the *CANopen over EtherCAT* (CoE) protocol. This dialog lists the content of the object list of the slave (SDO upload) and enables the user to modify the content of an object from this list. Details for the objects of the individual EtherCAT devices can be found in the device-specific object descriptions.



Index	Name	Flags	Wert
1000	Device type	RO	0x00000000 (0)
1008	Device name	RO	EL5001-0000
1009	Hardware version	RO	V00.01
100A	Software version	RO	V00.07
1011:0	Restore default parame...	RW	> 1 <
1011:01	Restore all	RW	0
1018:0	Identity object	RO	> 4 <
1018:01	Vendor id	RO	0x00000002 (2)
1018:02	Product code	RO	0x13893052 (327757906)
1018:03	Revision number	RO	0x00000000 (0)
1018:04	Serial number	RO	0x00000001 (1)
1A00:0	TxPDO 001 mapping	RO	> 2 <
1A00:01	Subindex 001	RO	0x3101:01, 8
1A00:02	Subindex 002	RO	0x3101:02, 32
1C00:0	SM type	RO	> 4 <
1C00:01	Subindex 001	RO	0x01 (1)
1C00:02	Subindex 002	RO	0x02 (2)
1C00:03	Subindex 003	RO	0x03 (3)
1C00:04	Subindex 004	RO	0x04 (4)
1C13:0	SM 3 PDO assign (inputs)	RW	> 1 <
1C13:01	Subindex 001	RW	0x1A00 (6656)
3101:0	Inputs	RO P	> 2 <
3101:01	Status	RO P	0x41 (65)
3101:02	Value	RO P	0x00000000 (0)
4061:0	Feature bits	RW	> 4 <
4061:01	disable frame error	RW	FALSE
4061:02	enbale power failure Bit	RW	FALSE
4061:03	enable inhibit time	RW	FALSE
4061:04	enable test mode	RW	FALSE
4066	SSI-coding	RW	Gray code (1)
4067	SSI-baudrate	RW	500 kBaud (3)
4068	SSI-frame type	RW	Multiturn 25 bit (0)
4069	SSI-frame size	RW	0x0019 (25)
406A	Data length	RW	0x0018 (24)
406B	Min. inhibit time[µs]	RW	0x0000 (0)

Fig. 81: “CoE – Online” tab

Object list display

Column	Description	
Index	Index and sub-index of the object	
Name	Name of the object	
Flags	RW	The object can be read, and data can be written to the object (read/write)
	RO	The object can be read, but no data can be written to the object (read only)
	P	An additional P identifies the object as a process data object.
Value	Value of the object	

Update List	The *Update list* button updates all objects in the displayed list
Auto Update	If this check box is selected, the content of the objects is updated automatically.
Advanced	The *Advanced* button opens the *Advanced Settings* dialog. Here you can specify which objects are displayed in the list.

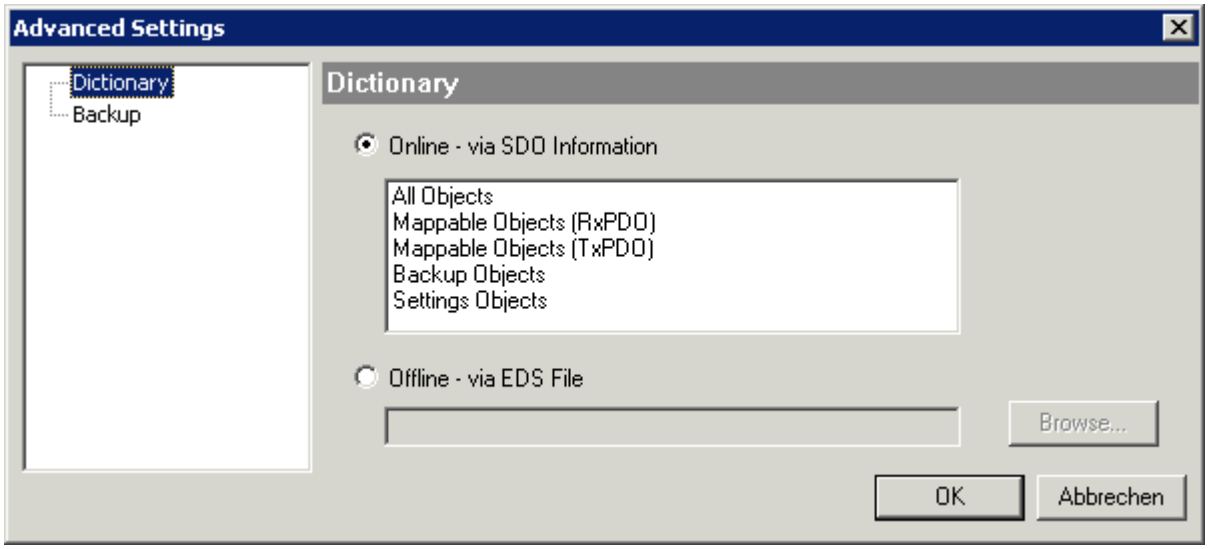


Fig. 82: Dialog “Advanced settings”

Online - via SDO Information	If this option button is selected, the list of the objects included in the object list of the slave is uploaded from the slave via SDO information. The list below can be used to specify which object types are to be uploaded.
Offline - via EDS File	If this option button is selected, the list of the objects included in the object list is read from an EDS file provided by the user.

„Online“ tab

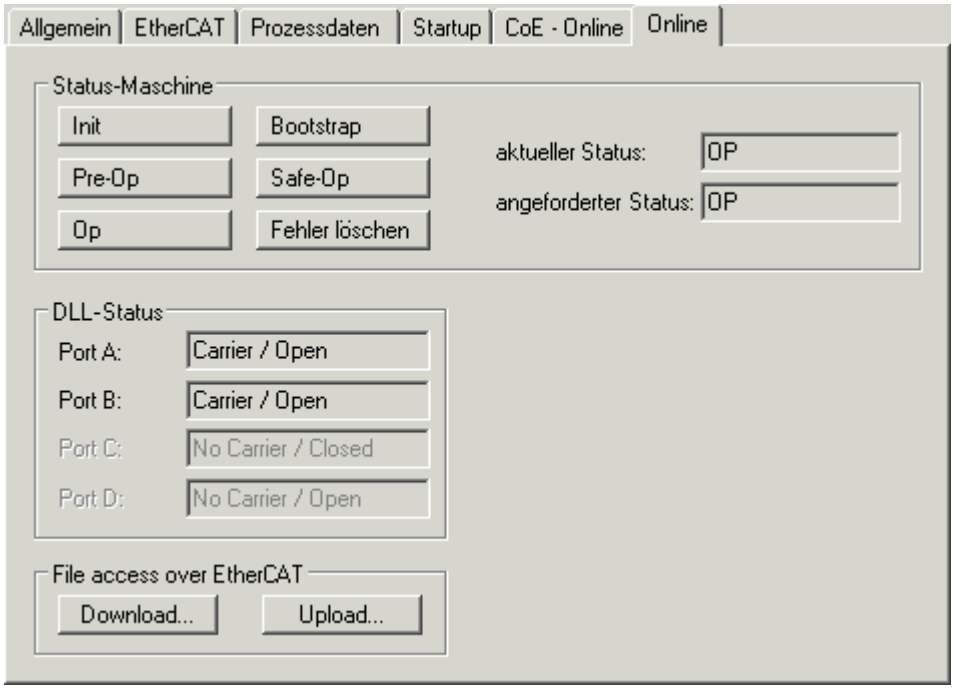


Fig. 83: „Online“ tab

State Machine

Init	This button attempts to set the EtherCAT device to the *Init* state.
Pre-Op	This button attempts to set the EtherCAT device to the *pre-operational* state.
Op	This button attempts to set the EtherCAT device to the *operational* state.
Bootstrap	This button attempts to set the EtherCAT device to the *Bootstrap* state.
Safe-Op	This button attempts to set the EtherCAT device to the *safe-operational* state.
Clear Error	This button attempts to delete the fault display. If an EtherCAT slave fails during change of state it sets an error flag. Example: An EtherCAT slave is in PREOP state (pre-operational). The master now requests the SAFEOP state (safe-operational). If the slave fails during change of state it sets the error flag. The current state is now displayed as ERR PREOP. When the *Clear Error* button is pressed the error flag is cleared, and the current state is displayed as PREOP again.
Current State	Indicates the current state of the EtherCAT device.
Requested State	Indicates the state requested for the EtherCAT device.

DLL Status

Indicates the DLL status (data link layer status) of the individual ports of the EtherCAT slave. The DLL status can have four different states:

Status	Description
No Carrier / Open	No carrier signal is available at the port, but the port is open.
No Carrier / Closed	No carrier signal is available at the port, and the port is closed.
Carrier / Open	A carrier signal is available at the port, and the port is open.
Carrier / Closed	A carrier signal is available at the port, but the port is closed.

File Access over EtherCAT

Download	With this button a file can be written to the EtherCAT device.
Upload	With this button a file can be read from the EtherCAT device.

5.3 General Notes - EtherCAT Slave Application

This summary briefly deals with a number of aspects of EtherCAT Slave operation under TwinCAT. More detailed information on this may be found in the corresponding sections of, for instance, the EtherCAT System Documentation.

Diagnosis in real time: WorkingCounter, EtherCAT State and Status

Generally speaking an EtherCAT Slave provides a variety of diagnostic information that can be used by the controlling task.

This diagnostic information relates to differing levels of communication. It therefore has a variety of sources, and is also updated at various times.

Any application that relies on I/O data from a fieldbus being correct and up to date must make diagnostic access to the corresponding underlying layers. EtherCAT and the TwinCAT System Manager offer comprehensive diagnostic elements of this kind. Those diagnostic elements that are helpful to the controlling task for diagnosis that is accurate for the current cycle when in operation (not during commissioning) are discussed below.

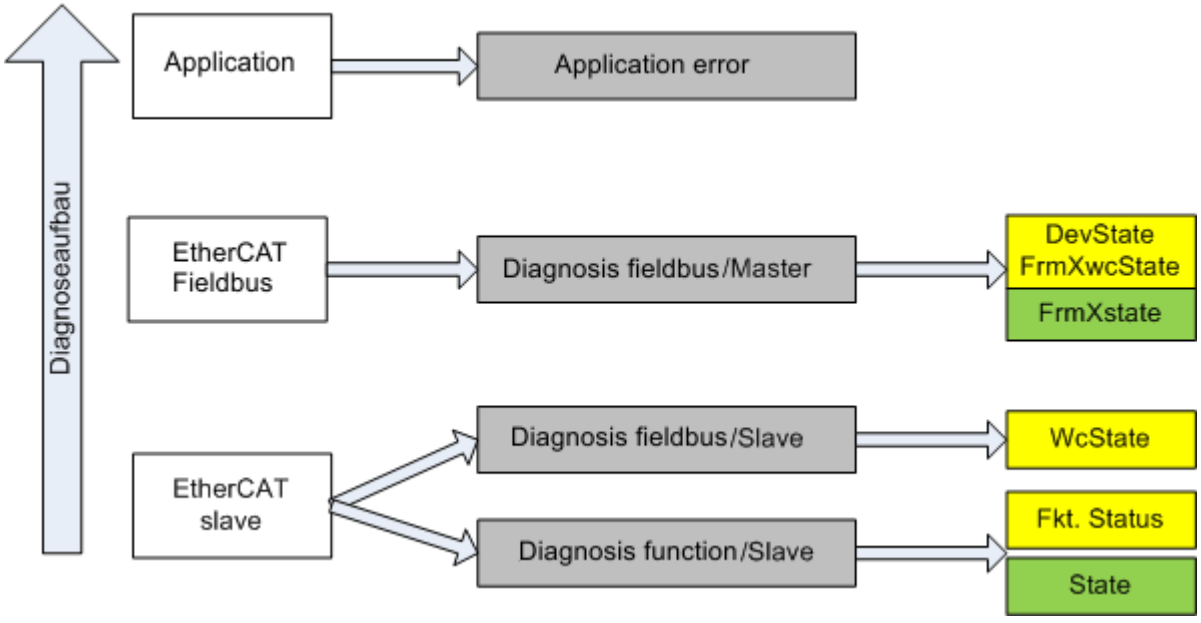


Fig. 84: Selection of the diagnostic information of an EtherCAT Slave

In general, an EtherCAT Slave offers

- communication diagnosis typical for a slave (diagnosis of successful participation in the exchange of process data, and correct operating mode)
  This diagnosis is the same for all slaves.

  as well as
- function diagnosis typical for a channel (device-dependent)
  See the corresponding device documentation

The colors in Fig. *"Selection of the diagnostic information of an EtherCAT Slave"* also correspond to the variable colors in the System Manager, see Fig. *"Basic EtherCAT Slave Diagnosis in the PLC"*.

Colour	Meaning
yellow	Input variables from the Slave to the EtherCAT Master, updated in every cycle
red	Output variables from the Slave to the EtherCAT Master, updated in every cycle
green	Information variables for the EtherCAT Master that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible status. It is therefore useful to read such variables through ADS.

Fig. *"Basic EtherCAT Slave Diagnosis in the PLC"* shows an example of an implementation of basic EtherCAT Slave Diagnosis. A Beckhoff EL3102 (2-channel analogue input terminal) is used here, as it offers both the communication diagnosis typical of a slave and the functional diagnosis that is specific to a channel. Structures are created as input variables in the PLC, each corresponding to the process image.

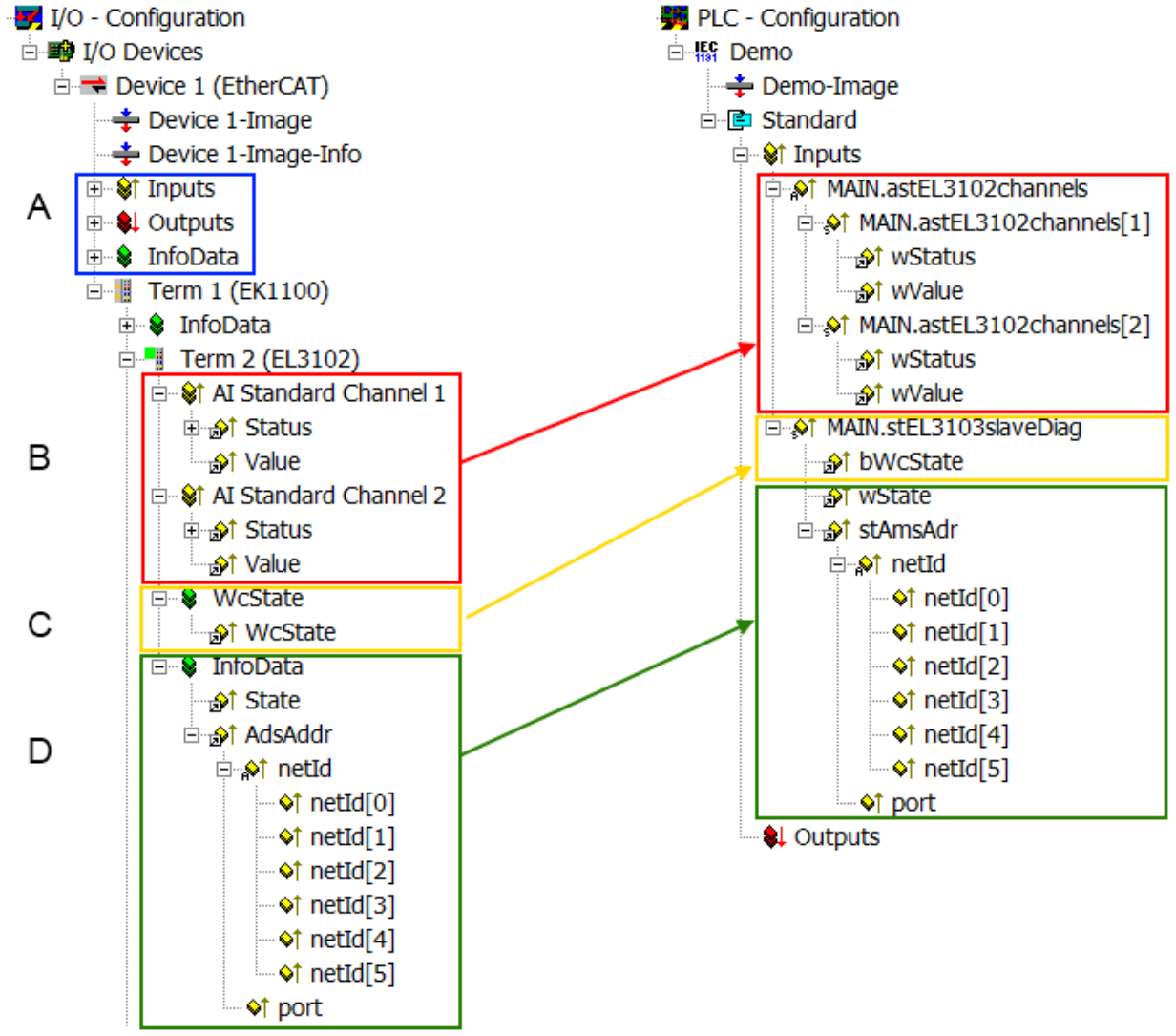


Fig. 85: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

Code	Function	Implementation	Application/evaluation
A	The EtherCAT Master's diagnostic information updated acyclically (yellow) or provided acyclically (green).		At least the DevState is to be evaluated for the most recent cycle in the PLC. The EtherCAT Master's diagnostic information offers many more possibilities than are treated in the EtherCAT System Documentation. A few keywords: • CoE in the Master for communication with/through the Slaves • Functions from *TcEtherCAT.lib* • Perform an OnlineScan
B	In the example chosen (EL3102) the EL3102 comprises two analogue input channels that transmit a single function status for the most recent cycle.	Status • the bit significations may be found in the device documentation • other devices may supply more information, or none that is typical of a slave	In order for the higher-level PLC task (or corresponding control applications) to be able to rely on correct data, the function status must be evaluated there. Such information is therefore provided with the process data for the most recent cycle.
C	For every EtherCAT Slave that has cyclic process data, the Master displays, using what is known as a WorkingCounter, whether the slave is participating successfully and without error in the cyclic exchange of process data. This important, elementary information is therefore provided for the most recent cycle in the System Manager 1. at the EtherCAT Slave, and, with identical contents 2. as a collective variable at the EtherCAT Master (see Point A) for linking.	WcState (Working Counter) 0: valid real-time communication in the last cycle 1: invalid real-time communication This may possibly have effects on the process data of other Slaves that are located in the same SyncUnit	In order for the higher-level PLC task (or corresponding control applications) to be able to rely on correct data, the communication status of the EtherCAT Slave must be evaluated there. Such information is therefore provided with the process data for the most recent cycle.
D	Diagnostic information of the EtherCAT Master which, while it is represented at the slave for linking, is actually determined by the Master for the Slave concerned and represented there. This information cannot be characterized as real-time, because it • is only rarely/never changed, except when the system starts up • is itself determined acyclically (e.g. EtherCAT Status)	State current Status (INIT..OP) of the Slave. The Slave must be in OP (=8) when operating normally. *AdsAddr* The ADS address is useful for communicating from the PLC/task via ADS with the EtherCAT Slave, e.g. for reading/writing to the CoE. The AMS-NetID of a slave corresponds to the AMS-NetID of the EtherCAT Master; communication with the individual Slave is possible via the *port* (= EtherCAT address).	Information variables for the EtherCAT Master that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible status. It is therefore possible to read such variables through ADS.

Attention

Diagnostic information

It is strongly recommended that the diagnostic information made available is evaluated so that the application can react accordingly.

CoE Parameter Directory

The CoE parameter directory (CanOpen-over-EtherCAT) is used to manage the set values for the slave concerned. Changes may, in some circumstances, have to be made here when commissioning a relatively complex EtherCAT Slave. It can be accessed through the TwinCAT System Manager, see Fig. *"EL3102, CoE directory"*:

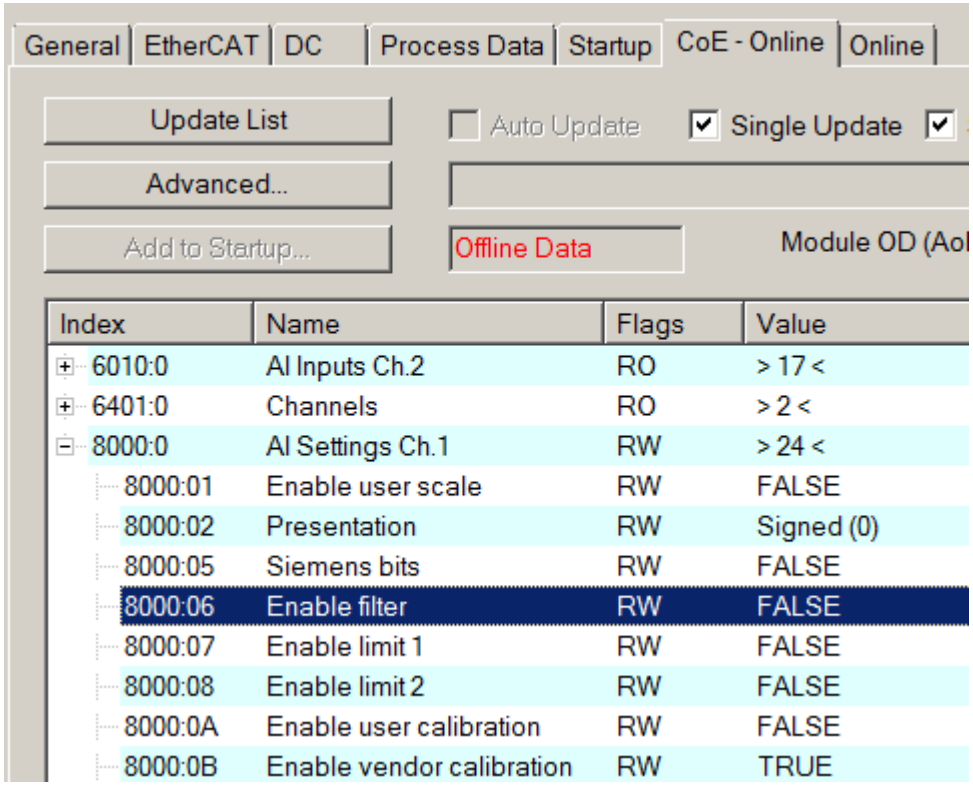


Fig. 86: EL3102, CoE directory

Note	**EtherCAT System Documentation** The comprehensive description in the EtherCAT System Documentation (EtherCAT Basics --> CoE Interface) must be observed!

A few brief extracts:

- Whether changes in the online directory are saved locally in the slave depends on the device. EL terminals (except the EL66xx) are able to save in this way.
- The user must manage the changes to the StartUp list.

Commissioning aid in the TwinCAT System Manager

Commissioning interfaces are being introduced as part of an ongoing process for EL/EP EtherCAT devices. These are available in TwinCAT System Managers from TwinCAT 2.11R2 and above. They are integrated into the System Manager through appropriately extended ESI configuration files.

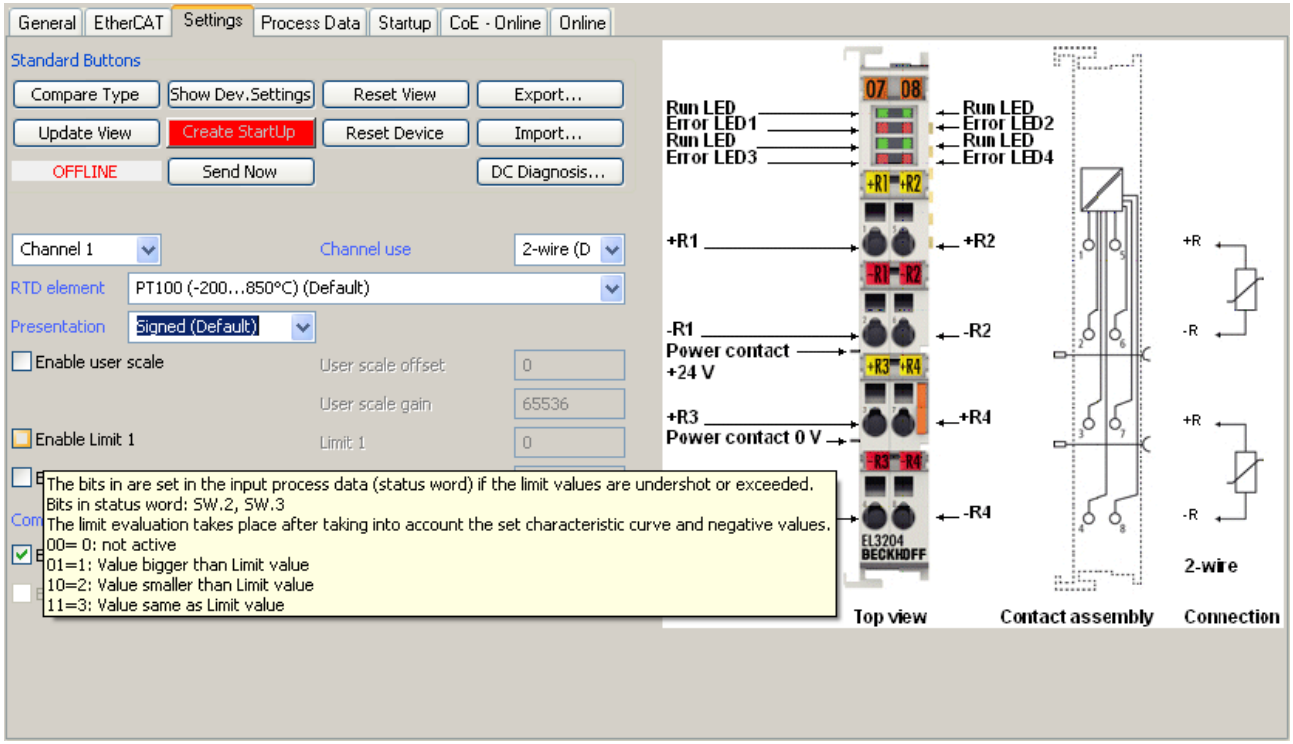


Fig. 87: Example of commissioning aid for a EL3204

This commissioning process simultaneously manages

- CoE Parameter Directory
- DC/FreeRun mode
- the available process data records (PDO)

Although the "Process Data", "DC", "Startup" and "CoE-Online" that used to be necessary for this are still displayed, it is recommended that, if the commissioning aid is used, the automatically generated settings are not changed by it.

The commissioning tool does not cover every possible application of an EL/EP device. If the available setting options are not adequate, the user can make the DC, PDO and CoE settings manually, as in the past.

EtherCAT State: automatic default behaviour of the TwinCAT System Manager and manual operation

After the operating power is switched on, an EtherCAT Slave must go through the following statuses

- INIT
- PREOP
- SAFEOP
- OP

to ensure sound operation. The EtherCAT Master directs these statuses in accordance with the initialization routines that are defined for commissioning the device by the ES/XML and user settings (Distributed Clocks (DC), PDO, CoE). See also the section on "Principles of Communication, EtherCAT State Machine" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

The EtherCAT Master itself must go through these routines when starting, until it has reached at least the OP target state.

The target state wanted by the user, and which is brought about automatically at start-up by TwinCAT, can be set in the System Manager. As soon as TwinCAT reaches the status RUN, the TwinCAT EtherCAT Master will approach the target states.

Standard setting

The advanced settings of the EtherCAT Master are set as standard:

- EtherCAT Master: OP
- Slaves: OP
  This setting applies equally to all Slaves.

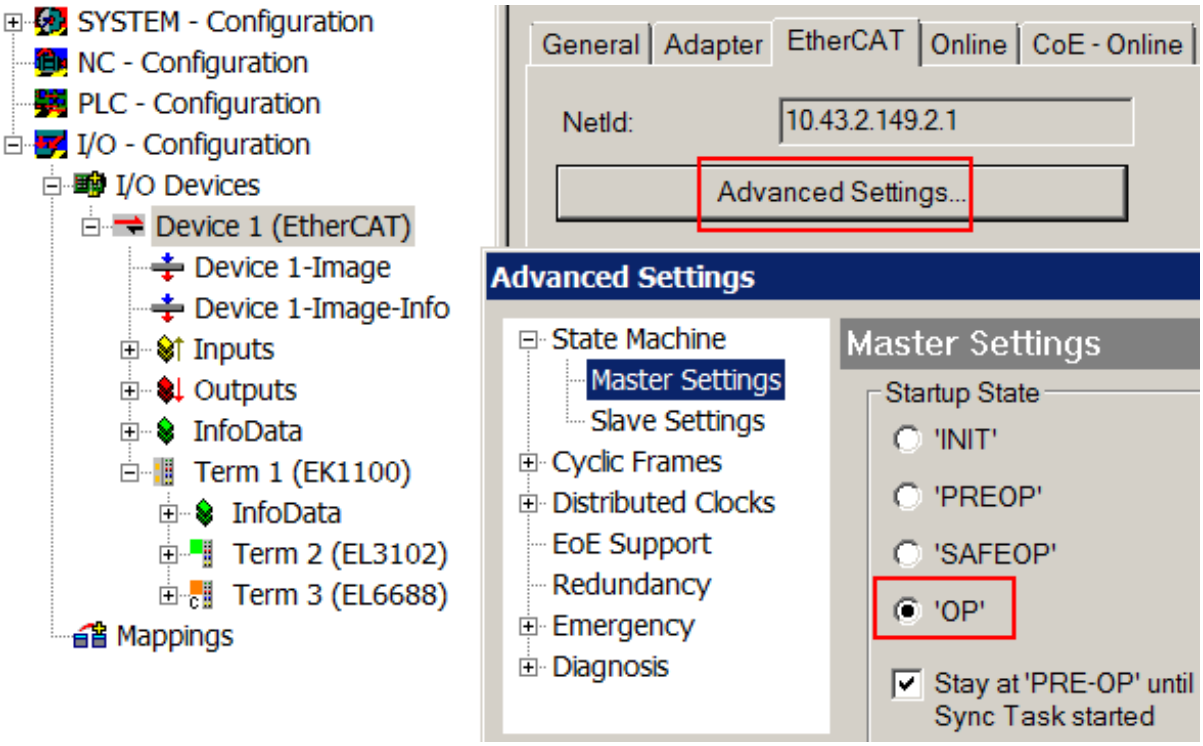


Fig. 88: Default behaviour of the System Manager

In addition, the target state of any particular Slave can be set in the "Advanced Settings" dialogue; the standard setting is again OP.

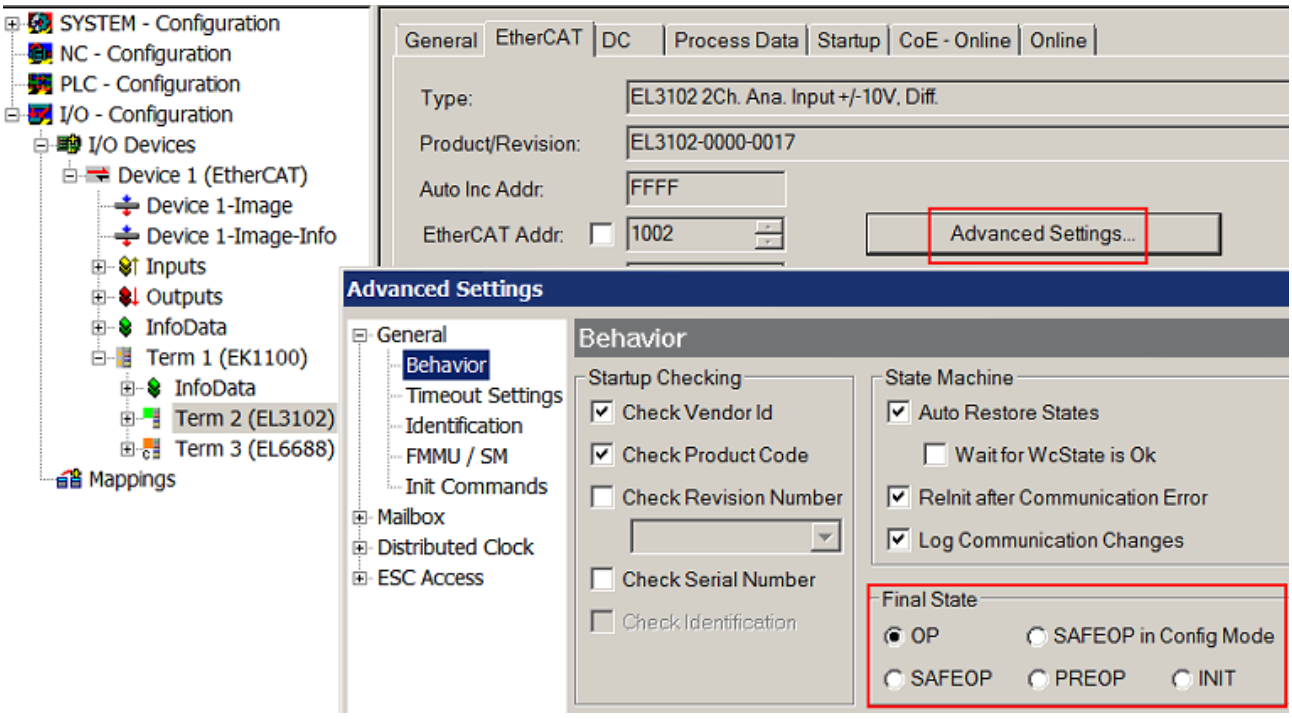


Fig. 89: Default target state in the Slave

Manual Control

There are particular reasons why it may be appropriate to control the states from the application/task/PLC. For instance:

- for diagnostic reasons
- to induce a controlled restart of axes
- because a change in the times involved in starting is desirable

In that case it is appropriate in the PLC application to use the PLC function blocks from the *TcEtherCAT.lib*, which is available as standard, and to work through the states in a controlled manner using, for instance, *FB_EcSetMasterState*.

It is then useful to put the settings in the EtherCAT Master to INIT for master and slave.

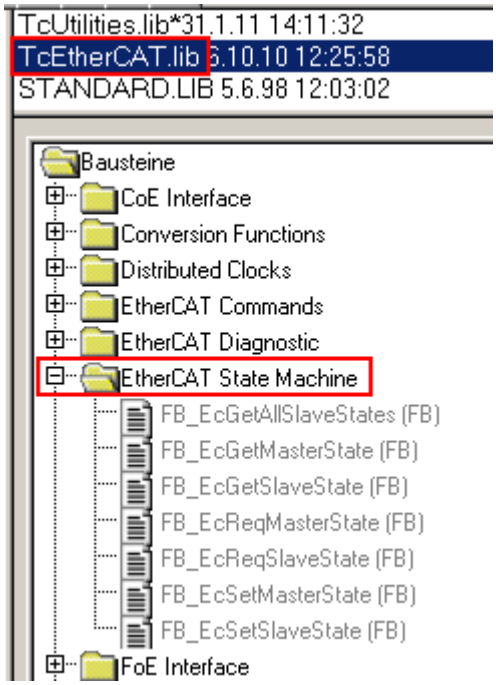
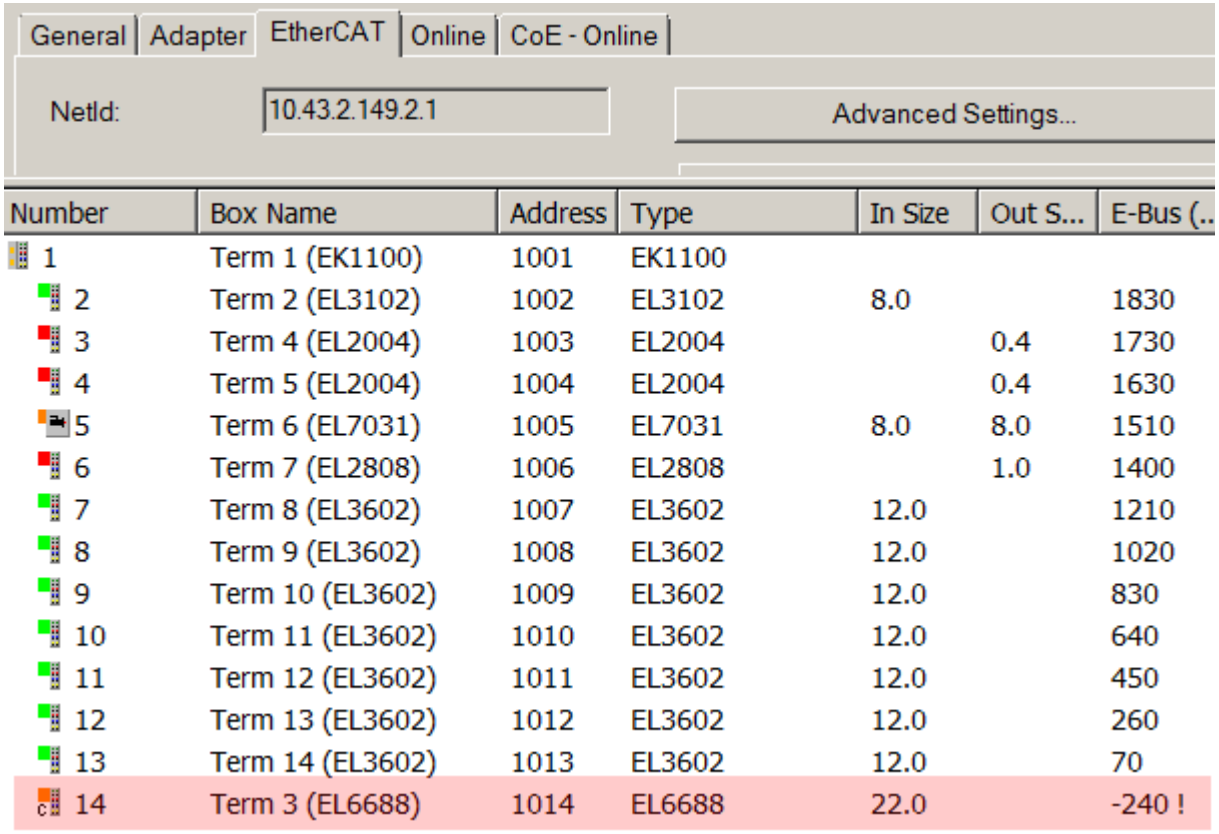


Fig. 90: PLC function blocks

Note regarding E-Bus current

EL/ES terminals are placed on the DIN rail at a coupler on the terminal strand. A Bus Coupler can supply the EL terminals added to it with the E-bus system voltage of 5 V; a coupler is thereby loadable up to 2 A as a rule. Information on how much current each EL terminal requires from the E-bus supply is available online and in the catalogue. If the added terminals require more current than the coupler can supply, then power feed terminals (e.g. EL9410) must be inserted at appropriate places in the terminal strand.

The pre-calculated theoretical maximum E-Bus current is displayed in the TwinCAT System Manager as a column value. A shortfall is marked by a negative total amount and an exclamation mark; a power feed terminal is to be placed before such a position.



Number	Box Name	Address	Type	In Size	Out S...	E-Bus (..
1	Term 1 (EK1100)	1001	EK1100			
2	Term 2 (EL3102)	1002	EL3102	8.0		1830
3	Term 4 (EL2004)	1003	EL2004		0.4	1730
4	Term 5 (EL2004)	1004	EL2004		0.4	1630
5	Term 6 (EL7031)	1005	EL7031	8.0	8.0	1510
6	Term 7 (EL2808)	1006	EL2808		1.0	1400
7	Term 8 (EL3602)	1007	EL3602	12.0		1210
8	Term 9 (EL3602)	1008	EL3602	12.0		1020
9	Term 10 (EL3602)	1009	EL3602	12.0		830
10	Term 11 (EL3602)	1010	EL3602	12.0		640
11	Term 12 (EL3602)	1011	EL3602	12.0		450
12	Term 13 (EL3602)	1012	EL3602	12.0		260
13	Term 14 (EL3602)	1013	EL3602	12.0		70
14	Term 3 (EL6688)	1014	EL6688	22.0		-240 !

Fig. 91: Illegally exceeding the E-Bus current

From TwinCAT 2.11 and above, a warning message "E-Bus Power of Terminal..." is output in the logger window when such a configuration is activated:

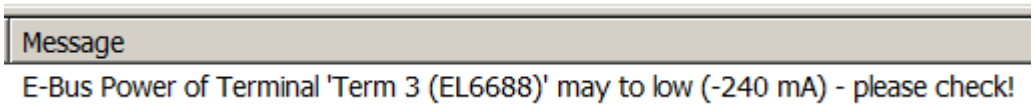


Fig. 92: Warning message for exceeding E-Bus current

Attention	**Caution! Malfunction possible!** The same ground potential must be used for the E-Bus supply of all EtherCAT terminals in a terminal block!

5.4 Distributed clocks reference clock

The distributed clock unit in the CU1128 is activated by default. The CU1128 can therefore be used as distributed clocks (DC) reference clock. The EtherCAT reference clock should be located near the start of the system hierarchy, in particular before flexible components (HotConnect). It is therefore advisable to use couplers or junction components such as the CU1128 as DC reference clock.

Without further settings, the TwinCAT System Manager uses the first DC-capable terminal/coupler as the reference clock; see Fig. *example topology with EL7201 as reference clock*.

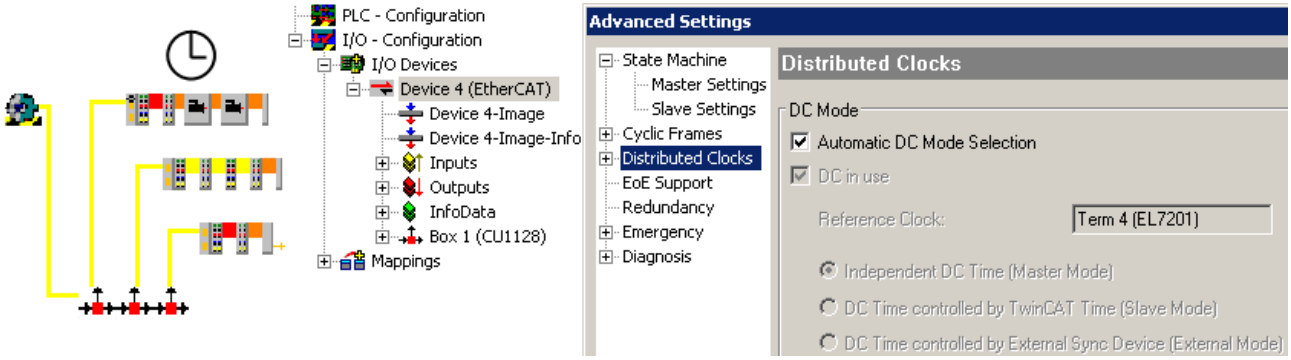


Fig. 93: example topology with EL7201 as reference clock

If the junction with the EL7201 was uncoupled, e.g. due to topology changes during machine runtime or a fault, the EtherCAT system would lose its reference clock. Central components should therefore deal with this function.

Under Advanced settings for first CU1128 (ESC1), tick the *UseAsPotentialReferenceClock* checkbox.

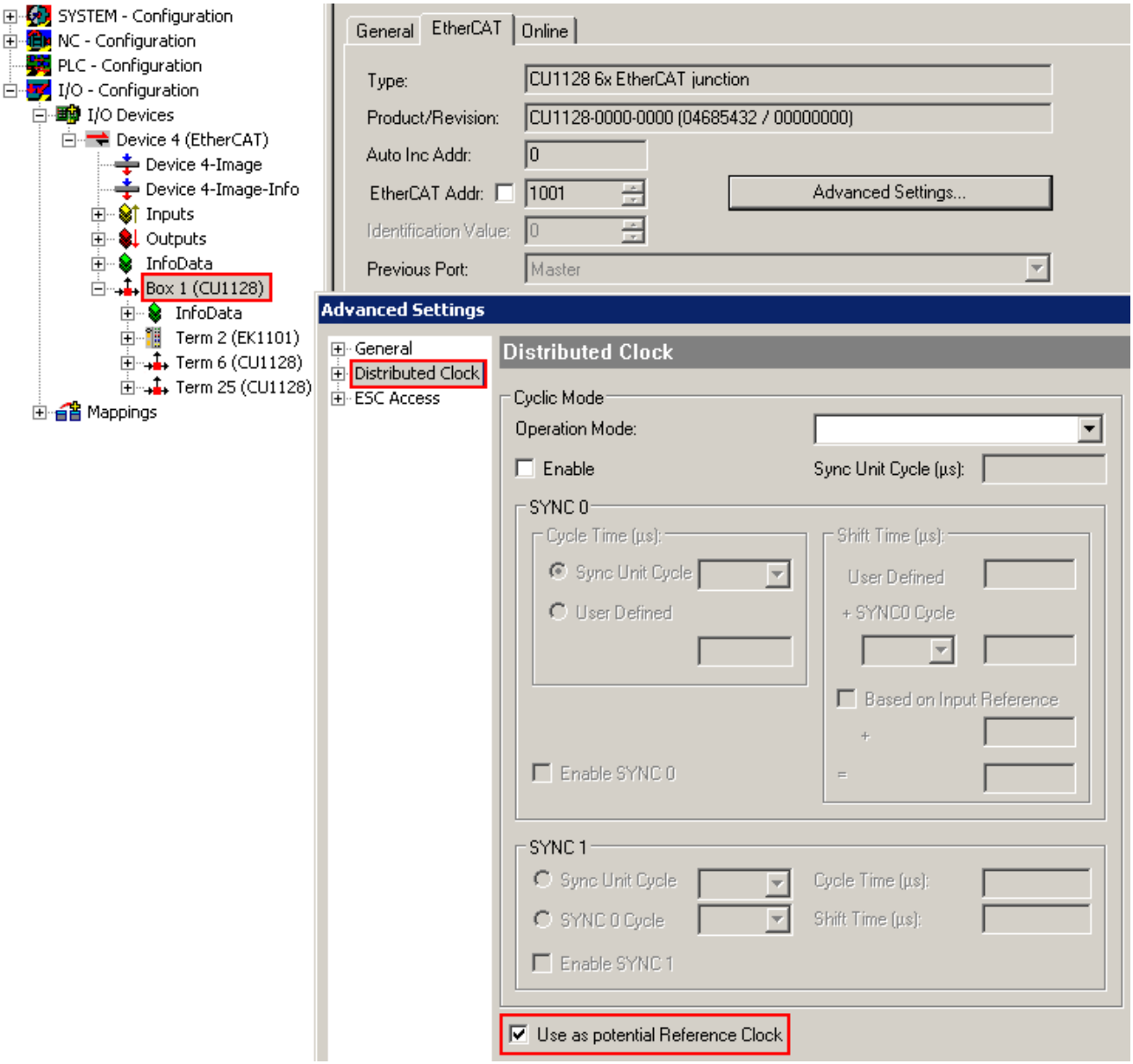


Fig. 94: activating the CU1128 DC function

Once activated, TwinCAT selects the CU1128 as reference clock.

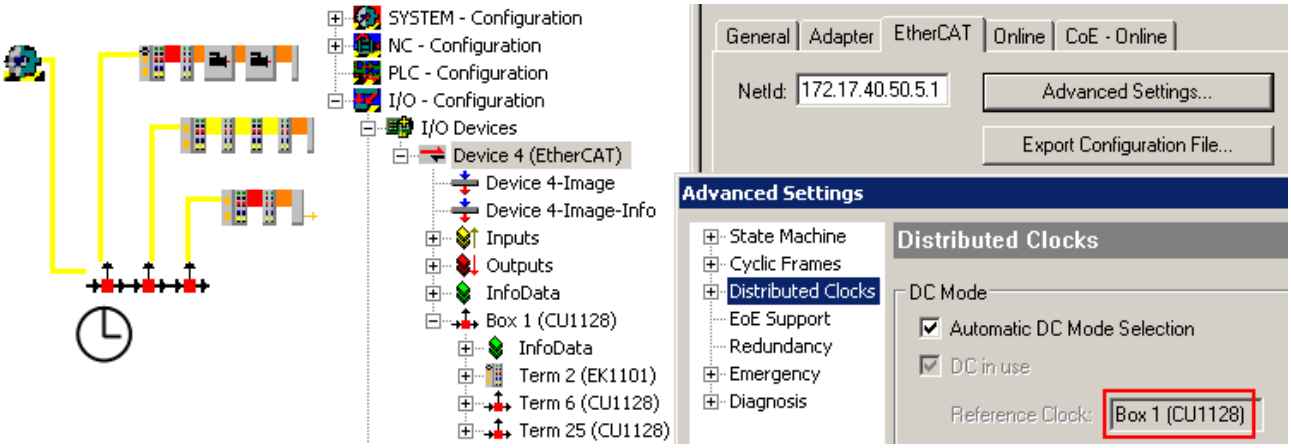


Fig. 95: CU1128 as reference clock

This function can be activated in components, refer here to the notices in the *EK1100* documentation.

5.5 Configuration of the CU1128 in the TwinCAT System Manager

Like any other EtherCAT slave, the CU1128 can be configured in two different ways in the TwinCAT System Manager:

- if it already exists and is available online and connected to the configuration PC: through scanning
- if it does not exist: through manual configuration setup

See chapter Configuration [▶ 26].

Since the CU1128 with its 3 integrated slaves is a little different from other slaves, some further information is provided below.

Configuration setup through online scan

From TwinCAT 2.10 b1340 (and therefore also TwinCAT 2.11 and higher) the CU1128 is detected during the scan, and the main device plus the two slaves (CU1128-0001 and CU1128-0002) are created.

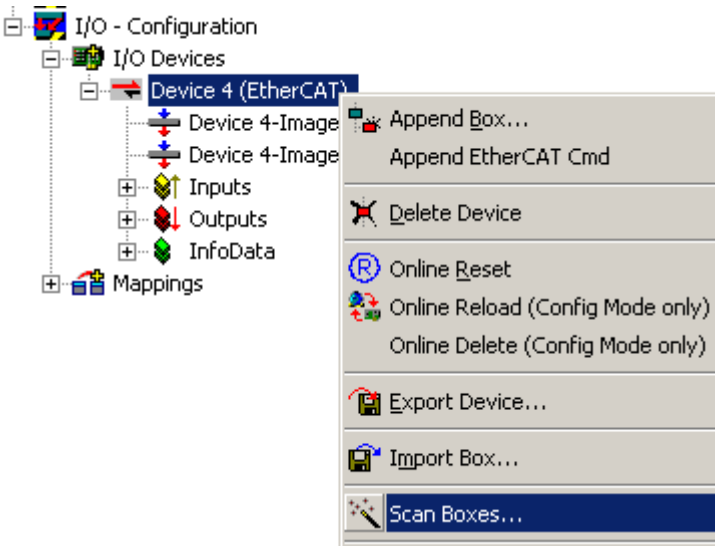


Fig. 96: scanning EtherCAT devices in Config mode

To check, make sure in the topology view (see Fig. *CU1128 topology display in the System Manager*) that the CU1128 is displayed as 3 junctions in one level.

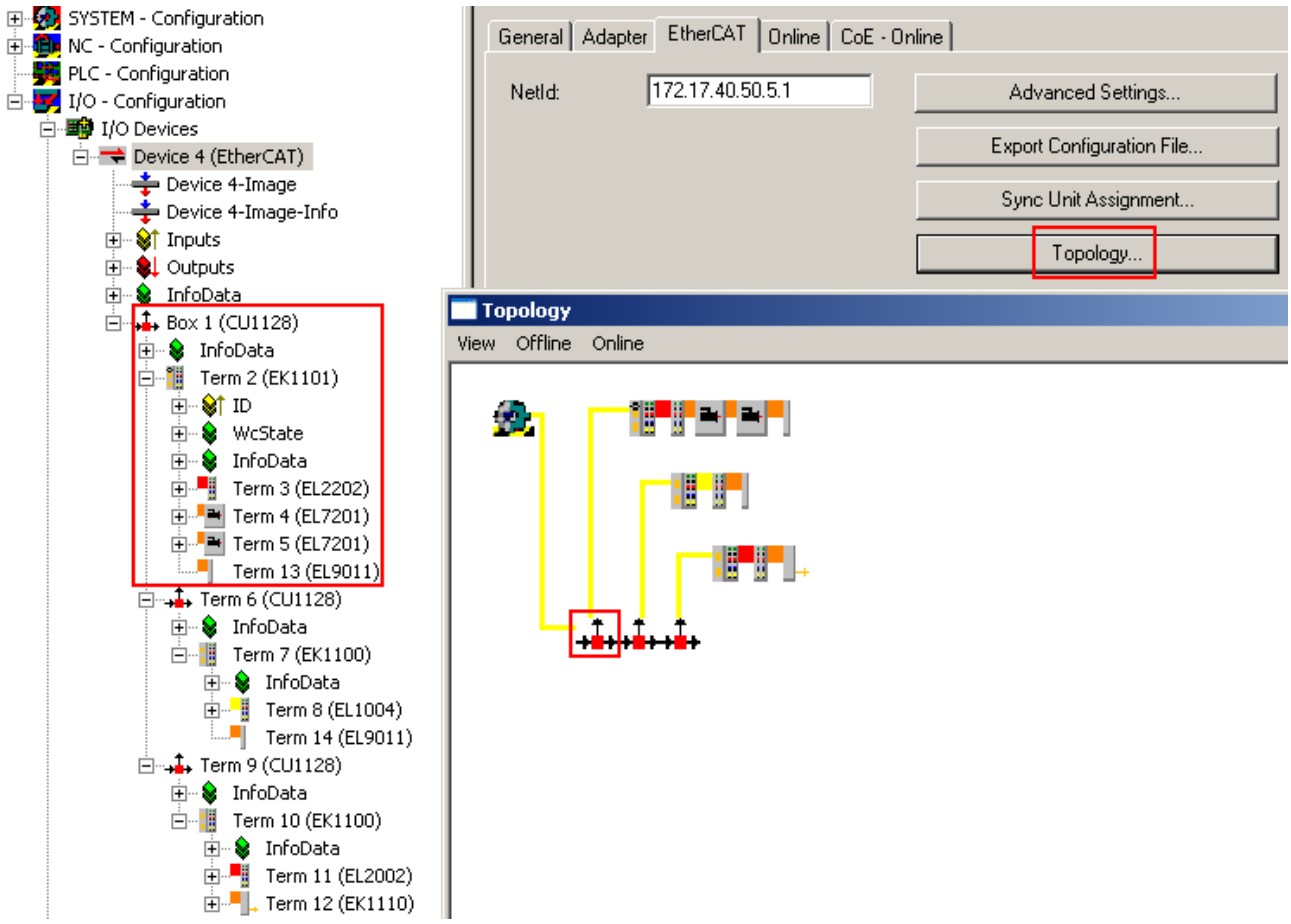


Fig. 97: CU1128 topology display in the System Manager

For comparison, Fig. 3 shows an incorrect display, caused by inadmissible relocation of the slaves in the configuration, for example.

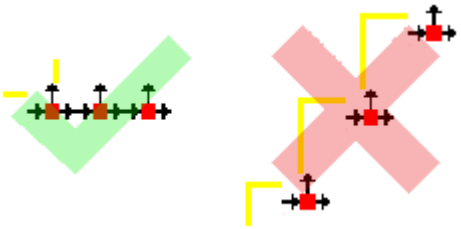


Fig. 98: CU1128 topology display

Manual configuration setup

For manual configuration setup of the CU1128 and attached devices, the correct ESI/XML file must be present, see here [▶ 33].

The process in detail:

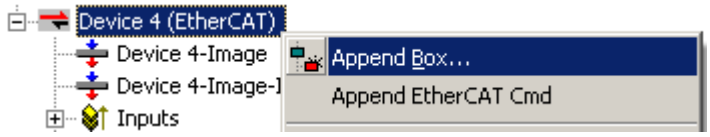


Fig. 99: right-click and select Append Box

A new device is added in the Ethernet layer. The port specification in the selection dialog plays an important part in the following; see Fig. *selection of the main device CU1128-000* on the right-hand side: it indicates at which outgoing port of the selected device the new device is to be connected. The selection options depend on the hardware (Ethernet or E-bus) available at ports A, B, C, D. In this example only Ethernet, port B is available at the EtherCAT master.

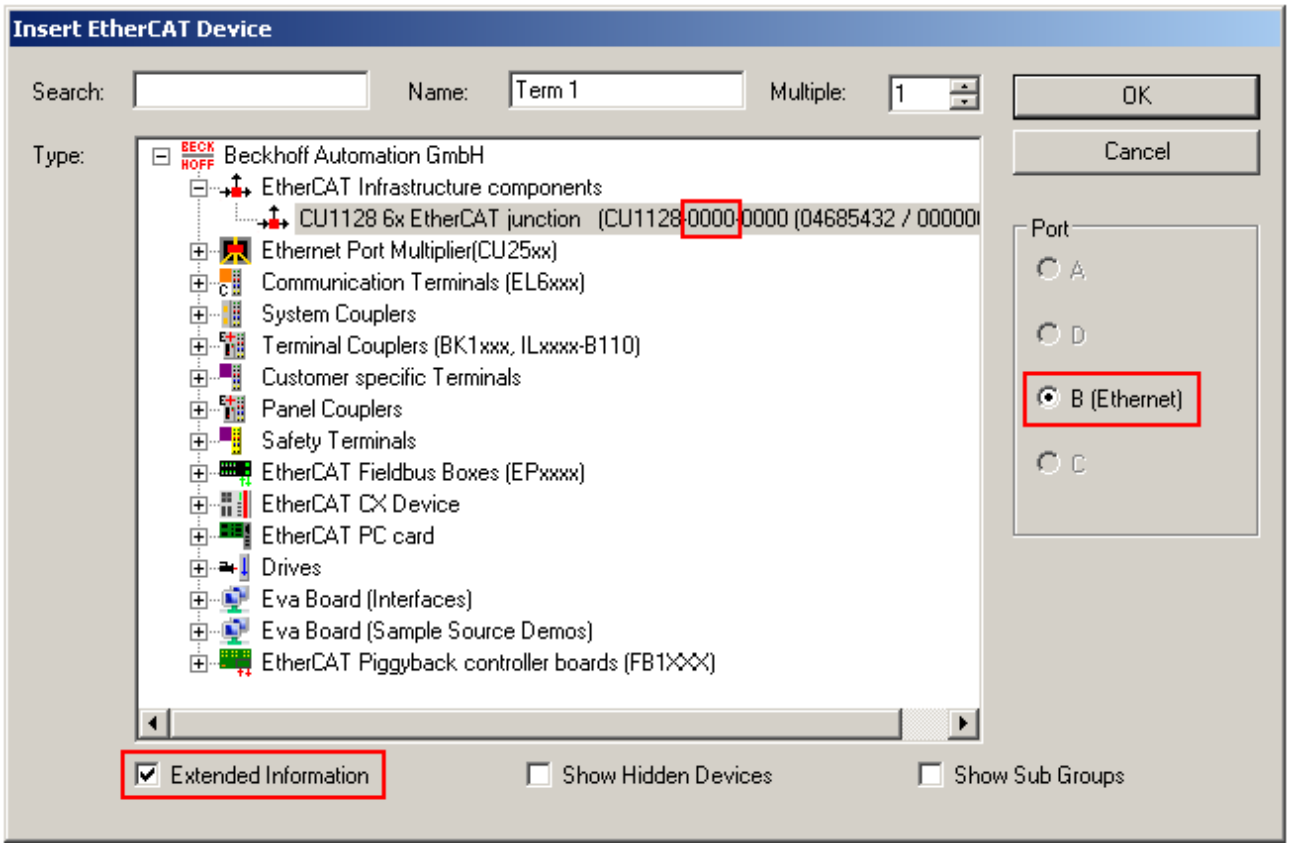


Fig. 100: selection of the main device CU1128-0000

The revision number can be shown in the selection dialog by ticking *ExtendedInformation*.

Note

Inserting CU1128

To add a CU1128 to a configuration, always use DeviceEtherCAT -> AppendBox and then move it to the required position in the configuration via drag & drop.

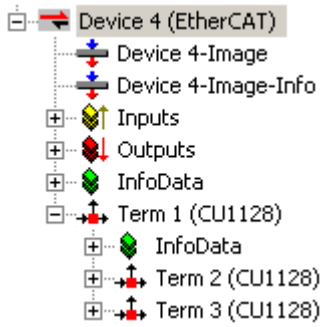


Fig. 101: added CU1128 with secondary devices Dev A and Dev B

The *PreviousPort* field of the secondary devices indicates that they are connected to port C of the previous device.

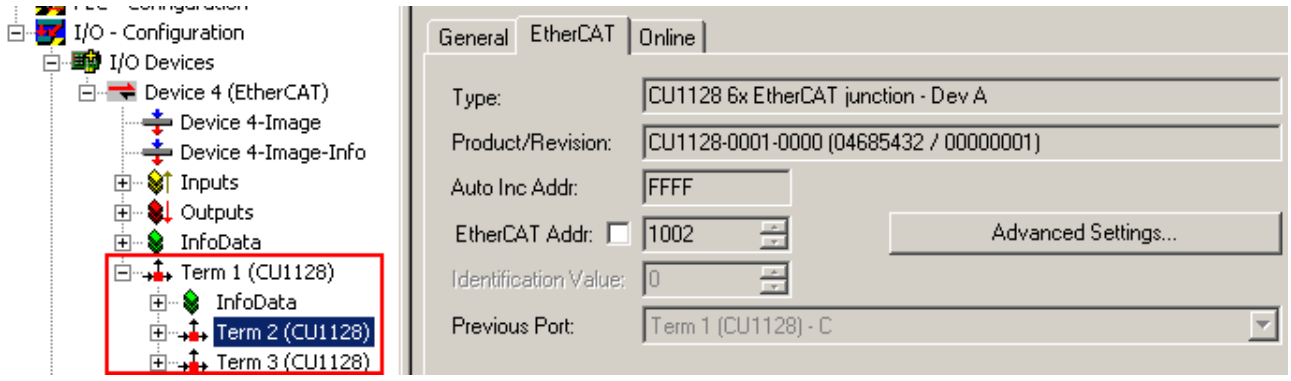


Fig. 102: previous port specification for CU1128, ESC2

Topology view shows the 3 devices as a chain:

Fig. 103: CU1128 in Topology view

Adding further CU1128 devices

It is common for further devices to be connected to the outgoing ports of the CU1128 at this stage. This is explained below, using a further CU1128 as an example. Make sure that the correct connection point is selected.

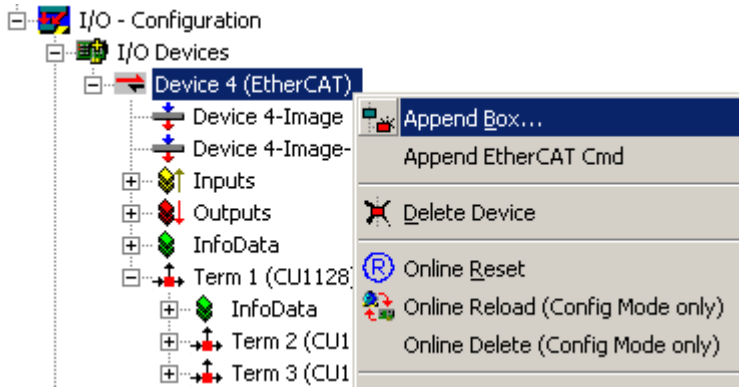


Fig. 104: adding further CU1128 devices

Add the CU1128-0000 via *DeviceEtherCAT*.

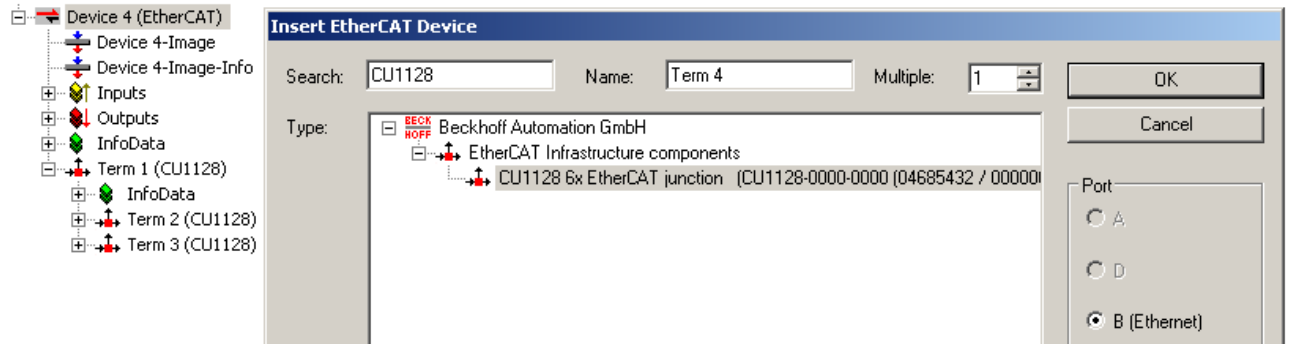


Fig. 105: selecting the CU1128-0000

The new *Term4* is added after the existing CU1128.

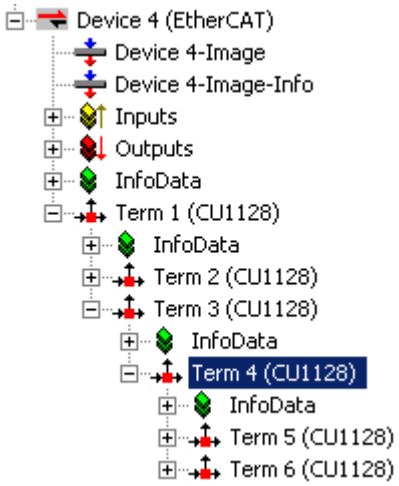


Fig. 106: added CU1128

Then use drag & drop to move it to the required position. In this example the intention is to connect it to port 4 of the CU1128, i.e. port D of ESC2 = *Term2*. This can be specified via *PreviousPort*.

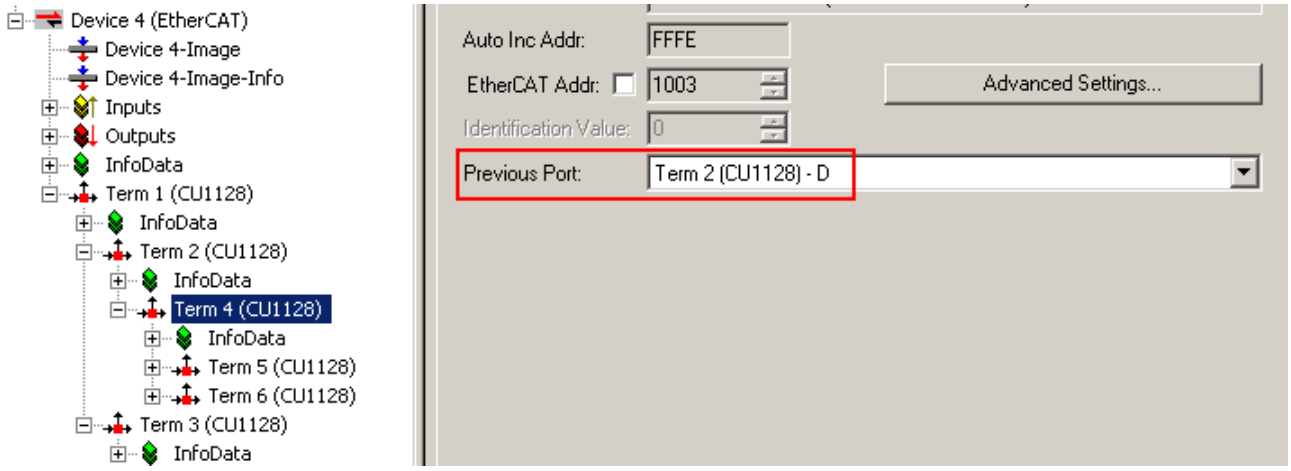


Fig. 107: selecting PreviousPort after moving the CU1128

Select Topology view to verify the configuration.

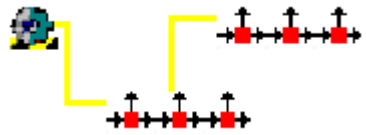


Fig. 108: topology display for 2x CU1128

Further EtherCAT devices

Other EtherCAT devices with Ethernet hardware can also be connected to any ports via *AppendBox*.

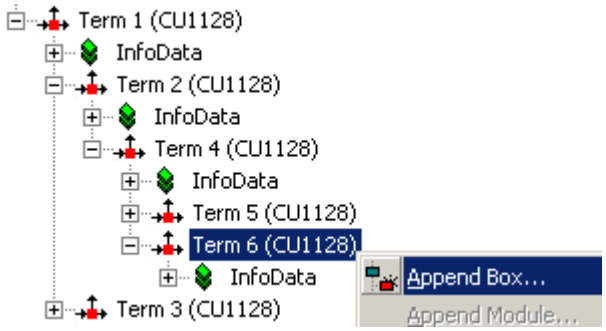


Fig. 109: adding further devices

Only Ethernet devices can be connected to a CU1128. Therefore, please make sure that *Ethernet* is selected under Port (see Fig. *adding a device*, right).

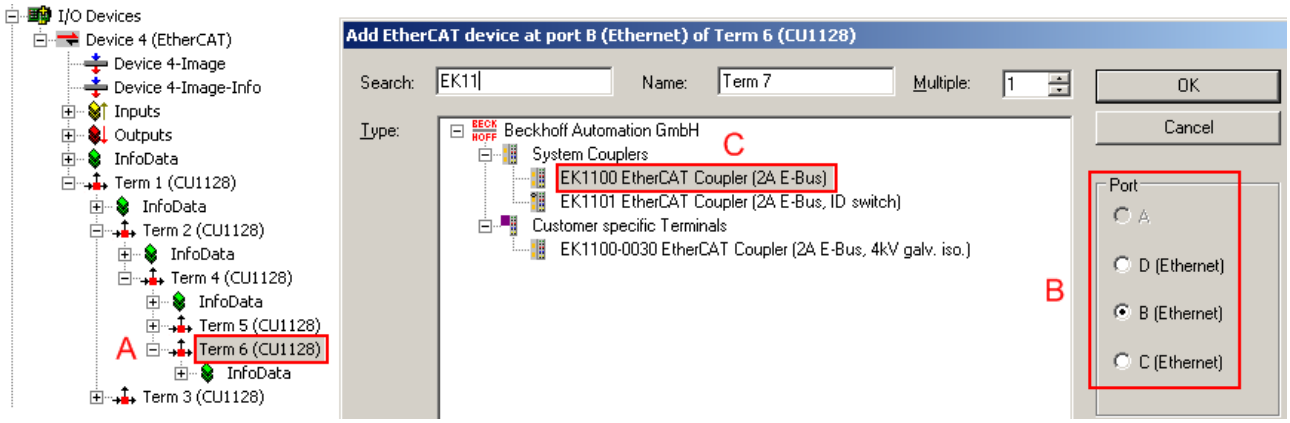


Fig. 110: adding a device

Sequence:

- A: once the CU1128 has been selected.
- B: and one of the Ethernet ports has been selected.
- C: the new device can be selected.

The E-Bus port (see Fig. *port selection for CU1128*) cannot be chosen, because the next CU1128 secondary device is attached to it.

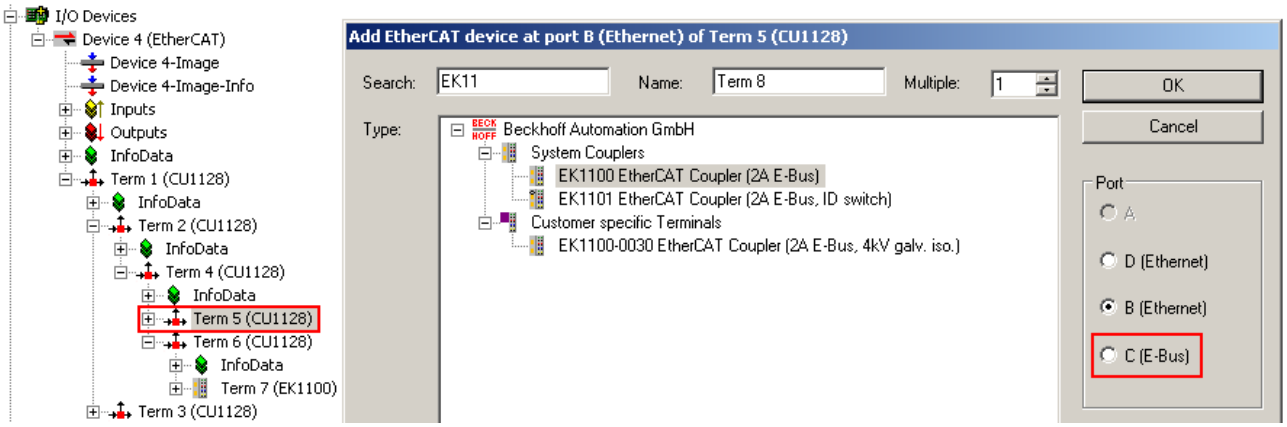


Fig. 111: port selection for CU1128

6 Appendix

6.1 EEPROM Update

The CU1128 features 3 internal EtherCAT slave controllers (ESCs). This means that 3 EEPROMs are available for write access. These are different in the CU1128, since all 3 ESCs have different port technologies (Ethernet or E-Bus); see Fig. *the 3 internal ESC names of the CU1128*.

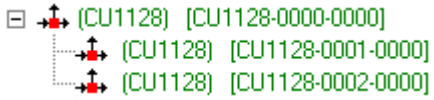


Fig. 112: the 3 internal ESC names of the CU1128

The main slave is the CU1128-0000, in this case revision -0000. The two following internal slaves are CU1128-0001 and CU1128-0002.

The EEPROM can be updated via the EEPROM Update dialog in the System Manager. This is only necessary in special cases and should only be done if instructed by Beckhoff support/service.

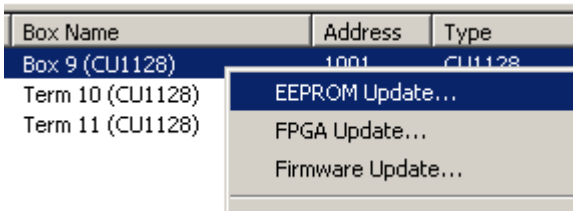


Fig. 113: EEPROM Update TwinCAT

All 3 internal ESCs can be write-accessed sequentially, followed by an off/on restart, in order to load the EEPROM. For assignment in the TwinCAT selection dialog, see Fig. *Currently available EEPROM descriptions for CU1128 and assignment.*

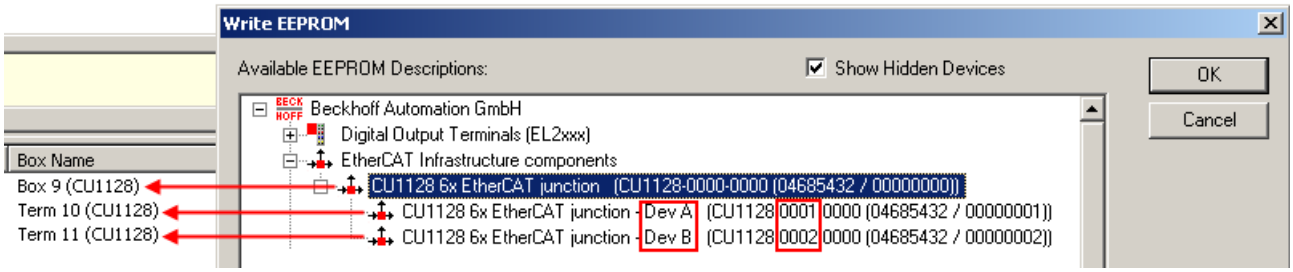


Fig. 114: currently available EEPROM descriptions for CU1128 and assignment

Dev A is intended for ESC2 CU1128-0001, *Dev B* for ESC3, CU1128-0002.

6.2 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

Two UL certificates are met in the Beckhoff EtherCAT product range, depending upon the components:

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

c UL US

Almost all current EtherCAT products (as at 2010/05) are UL certified without restrictions.

- UL certification according to UL508 with limited power consumption
  The current consumed by the device is limited to a max. possible current consumption of 4 A. Devices with this kind of certification are marked by this sign:

Almost all current EtherCAT products (as at 2010/05) are UL certified without restrictions.

Application

If terminals certified *with restrictions* are used, then the current consumption at 24 V_{DC} must be limited accordingly by means of supply

- from an isolated source protected by a fuse of max. 4A (according to UL248) or
- from a voltage supply complying with *NEC class 2*.
  A voltage source complying with *NEC class 2* may not be connected in series or parallel with another *NEC class 2* compliant voltage supply!

These requirements apply to the supply of all EtherCAT bus couplers, power adaptor terminals, Bus Terminals and their power contacts.

6.3 Support and Service

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

Beckhoff's branch offices and representatives

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

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

You will also find further documentation for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

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

Beckhoff Support

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

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

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

Beckhoff Service

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

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

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

List of illustration