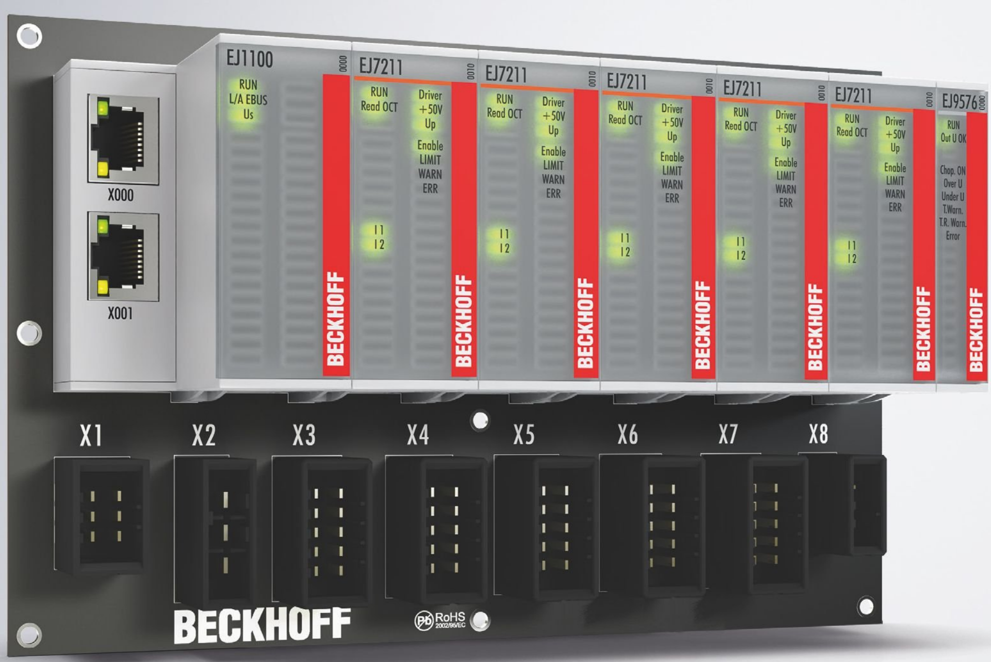
Dokumentation | EN

EJ2564

EtherCAT plug-in module, 4-channel LED output, 5...48 V DC, 4 A, RGBW,common anode



2022-12-20 | Version: 1.0

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 documentation and the following notes and explanations are followed when installing and commissioning these components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

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

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

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

Patent Pending

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

EtherCAT

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

Copyright

1.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

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

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

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

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

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3 Intended use

⚠ **WARNING**

Caution - Risk of injury!

EJ components may only be used for the purposes described below!

1.4 Signal distribution board

NOTE

Signal distribution board

Make sure that the EtherCAT plug-in modules are used only on a signal distribution board that has been developed and manufactured in accordance with the Design Guide.

1.5 Documentation issue status

Version	Comment
1.0	• First publication EJ2564

1.6 Marking of EtherCAT plug-in modules

Designation

A Beckhoff EtherCAT device has a 14-digit **technical designation**, made up as follows (e.g. EJ1008-0000-0017)

- **Order identifier**
  - family key: EJ
  - product designation: The first digit of product designation is used for assignment to a product group (e.g. EJ2xxx = digital output module).
  - Version number: The four digit version number identifies different product variants.
- **Revision number:**
  It is incremented when changes are made to the product.

The Order identifier and the revision number are printed on the side of EtherCAT plug-in modules (s. following illustration (A and B).

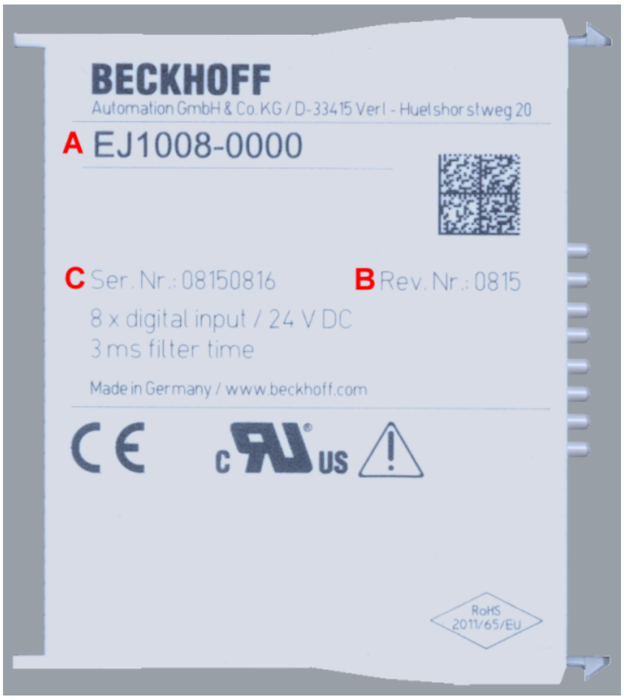


Fig. 1: Order identifier (A), Revision number (B) and serial number (C) using the example of EJ1008

Product group	Example		
	Product designation	Version	Revision
EtherCAT Coupler EJ11xx	EJ1101	-0022 (Coupler with external connectors, power supply module and optional ID switches	-0016
Digital input modules EJ1xxx	EJ1008 8-channel	-0000 (basic type)	-0017
Digital output modules EJ2xxx	EJ2521 1-channel	-0224 (2 x 24 V outputs)	-0016
Analog input modules EJ3xxx	EJ3318 8-channel thermocouple	-0000 (basic type)	-0017
Analog output modules EJ4xxx	EJ4134 4-channel	-0000 (basic type)	-0019
Special function modules EJ5xxx, EJ6xxx	EJ6224 IO-Link master	-0090 (with TwinSAFE SC)	-0016
Motion modules EJ7xxx	EJ7211 servomotor	-9414 (with ECT, STO and TwinSAFE SC)	-0029

Notes

- The elements mentioned above result in the **technical designation**. EJ1008-0000-0017 is used in the example below.
- EJ1008-0000 is the **order identifier**, in the case of "-0000" usually abbreviated to EJ1008.
- The **revision** -0017 shows the technical progress, such as the extension of features with regard to the EtherCAT communication, and is managed by Beckhoff.
  In principle, a device with a higher revision can replace a device with a lower revision, unless specified otherwise, e.g. in the documentation.
  Associated and synonymous with each revision there is usually a description (ESI, EtherCAT Slave Information) in the form of an XML file, which is available for download from the Beckhoff web site.
- The product designation, version and revision are read as decimal numbers, even if they are technically saved in hexadecimal.

Serial number

The serial number for EtherCAT plug-in modules is usually the 8-digit number printed on the side of the module (see following illustration C). 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.

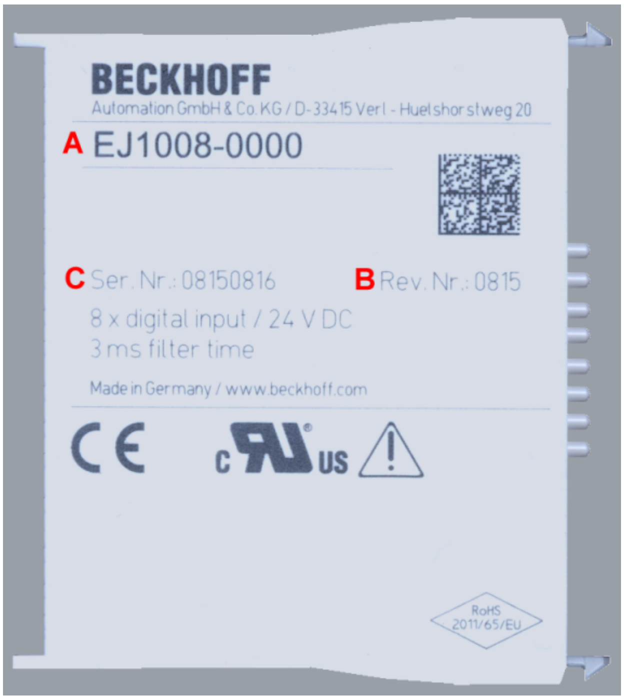


Fig. 2: Order identifier (A), revision number (B) and serial number (C) using the example of EJ1008

Serial number	Example serial number: 08 15 08 16
KK - week of production (CW, calendar week)	08 - week of production: 08
YY - year of production	15 - year of production: 2015
FF - firmware version	08 -f irmware version: 08
HH - hardware version	16 - hardware version: 16

1.6.1 Beckhoff Identification Code (BIC)

The **B**eckhoff **I**dentification **C**ode (BIC) is increasingly being applied to Beckhoff products to uniquely identify the product. The BIC is represented as a Data Matrix Code (DMC, code scheme ECC200), the content is based on the ANSI standard MH10.8.2-2016.

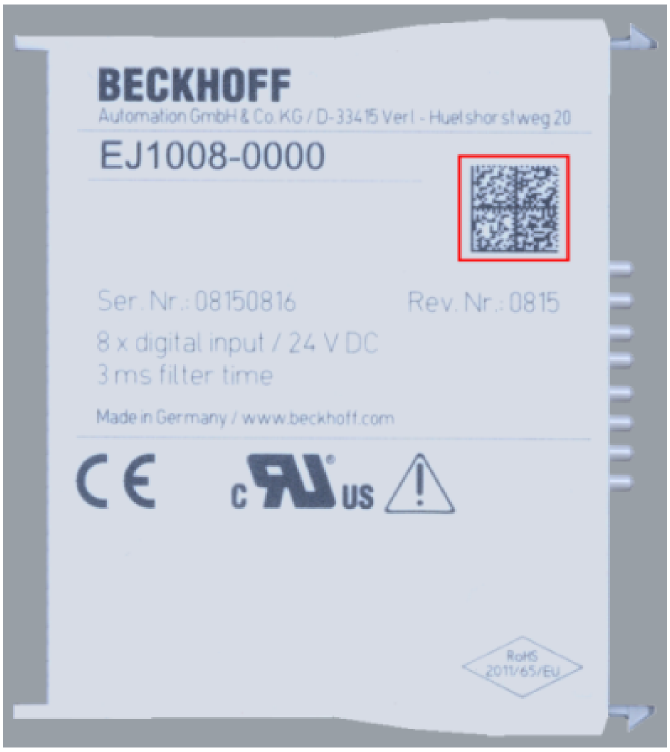


Fig. 3: BIC as data matrix code (DMC, code scheme ECC200)

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

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

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

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

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

The following information is contained:

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

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

Structure of the BIC

Example of composite information from items 1 - 4 and with the above given example value on positon 6. The data identifiers are marked in bold font for better display:

1P072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

Accordingly as DMC:

Fig. 4: Example DMC **1P**072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

BTN

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

NOTE

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

1.6.2 Electronic access to the BIC (eBIC)

Electronic BIC (eBIC)

The Beckhoff Identification Code (BIC) is applied to the outside of Beckhoff products in a visible place. If possible, it should also be electronically readable.

Decisive for the electronic readout is the interface via which the product can be electronically addressed.

K-bus devices (IP20, IP67)

Currently, no electronic storage and readout is planned for these devices.

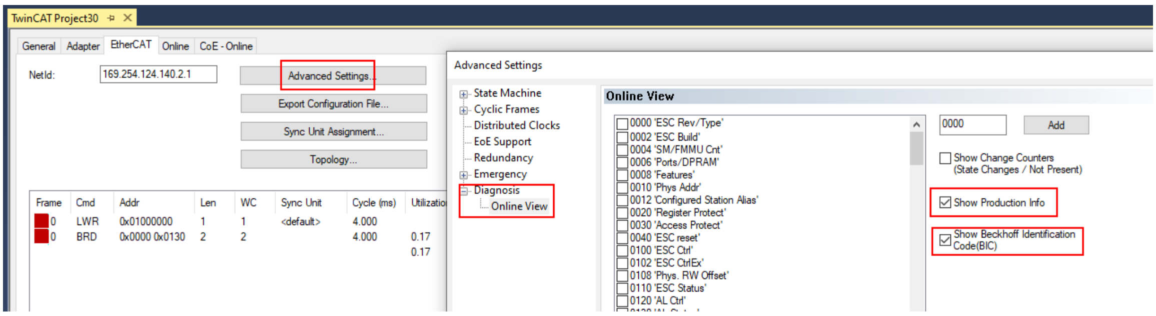
EtherCAT devices (IP20, IP67)

All Beckhoff EtherCAT devices have a so-called ESI-EEPROM, which contains the EtherCAT identity with the revision number. Stored in it is the EtherCAT slave information, also colloquially known as ESI/XML configuration file for the EtherCAT master. See the corresponding chapter in the EtherCAT system manual (Link) for the relationships.

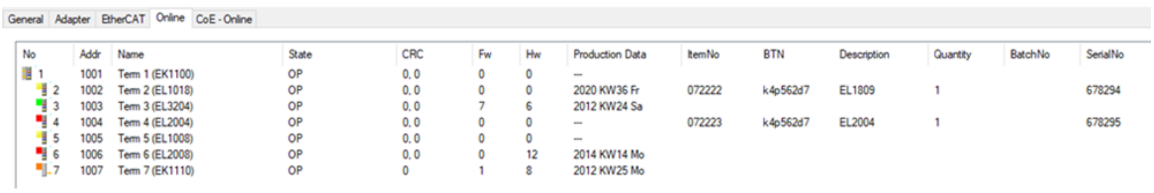
The eBIC is also stored in the ESI-EEPROM. The eBIC was introduced into the Beckhoff I/O production (terminals, box modules) from 2020; widespread implementation is expected in 2021.

The user can electronically access the eBIC (if existent) as follows:

- With all EtherCAT devices, the EtherCAT master (TwinCAT) can read the eBIC from the ESI-EEPROM
  - From TwinCAT 3.1 build 4024.11, the eBIC can be displayed in the online view.
  - To do this,
    check the checkbox "Show Beckhoff Identification Code (BIC)" under
    EtherCAT → Advanced Settings → Diagnostics:



  - The BTN and its contents are then displayed:



  - Note: as can be seen in the illustration, the production data HW version, FW version and production date, which have been programmed since 2012, can also be displayed with "Show Production Info".
  - From TwinCAT 3.1. build 4024.24 the functions *FB_EcReadBIC* and *FB_EcReadBTN* for reading into the PLC and further eBIC auxiliary functions are available in the Tc2_EtherCAT Library from v3.3.19.0.
- In the case of EtherCAT devices with CoE directory, the object 0x10E2:01 can additionally by used to display the device's own eBIC; the PLC can also simply access the information here:

- The device must be in PREOP/SAFEOP/OP for access:

Index	Name	Flags	Value
1000	Device type	RO	0x015E1389 (22942601)
1008	Device name	RO	ELM3704-0000
1009	Hardware version	RO	00
100A	Software version	RO	01
100B	Bootloader version	RO	J0.1.27.0
+ 1011:0	Restore default parameters	RO	> 1 <
+ 1018:0	Identity	RO	> 4 <
- 10E2:0	Manufacturer-specific Identification C...	RO	> 1 <
10E2:01	SubIndex 001	RO	1P158442SBTN0008jekp1KELM3704 Q1 2P482001000016
+ 10F0:0	Backup parameter handling	RO	> 1 <
+ 10F3:0	Diagnosis History	RO	> 21 <
10F8	Actual Time Stamp	RO	0x170bfb277e

  - the object 0x10E2 will be introduced into stock products in the course of a necessary firmware revision.
  - From TwinCAT 3.1. build 4024.24 the functions *FB_EcCoEReadBIC* and *FB_EcCoEReadBTN* for reading into the PLC and further eBIC auxiliary functions are available in the Tc2_EtherCAT Library from v3.3.19.0.
- Note: in the case of electronic further processing, the BTN is to be handled as a string(8); the identifier "SBTN" is not part of the BTN.
- Technical background
  The new BIC information is additionally written as a category in the ESI-EEPROM during the device production. The structure of the ESI content is largely dictated by the ETG specifications, therefore the additional vendor-specific content is stored with the help of a category according to ETG.2010. ID 03 indicates to all EtherCAT masters that they must not overwrite these data in case of an update or restore the data after an ESI update.
  The structure follows the content of the BIC, see there. This results in a memory requirement of approx. 50..200 bytes in the EEPROM.
- Special cases
  - If multiple, hierarchically arranged ESCs are installed in a device, only the top-level ESC carries the eBIC Information.
  - If multiple, non-hierarchically arranged ESCs are installed in a device, all ESCs carry the eBIC Information.
  - If the device consists of several sub-devices with their own identity, but only the top-level device is accessible via EtherCAT, the eBIC of the top-level device is located in the CoE object directory 0x10E2:01 and the eBICs of the sub-devices follow in 0x10E2:nn.

Profibus/Profinet/DeviceNet… Devices

Currently, no electronic storage and readout is planned for these devices.

1.6.3 Certificates

- The EhterCAT plug-in modules meet the requirements of the EMC and Low Voltage Directive. The CE mark is printed on the side of the modules.
- The cRUus imprint identifies devices that meet product safety requirements according to U.S. and Canadian regulations.
- The warning symbol is a request to read the corresponding documentation. The documentations for EtherCAT plug-in modules can be downloaded from the Beckhoff homepage.

Fig. 5: Marking for CE and UL using EJ1008 as an example

2 System overview

Electronically, the EJxxxx EtherCAT plug-in modules are based on the EtherCAT I/O system. The EJ system consists of the signal distribution board and EtherCAT plug-in modules. It is also possible to connect an IPC to the EJ system.
The EJ system is suitable for mass production applications, applications with small footprint and applications requiring a low total weight.
The machine complexity can be extended by means of the following:

- reserve slots,
- the use of placeholder modules,
- linking of EtherCAT Terminals and EtherCAT Boxes via an EtherCAT connection.

The following diagram illustrates an EJ system. The components shown are schematic, to illustrate the functionality.

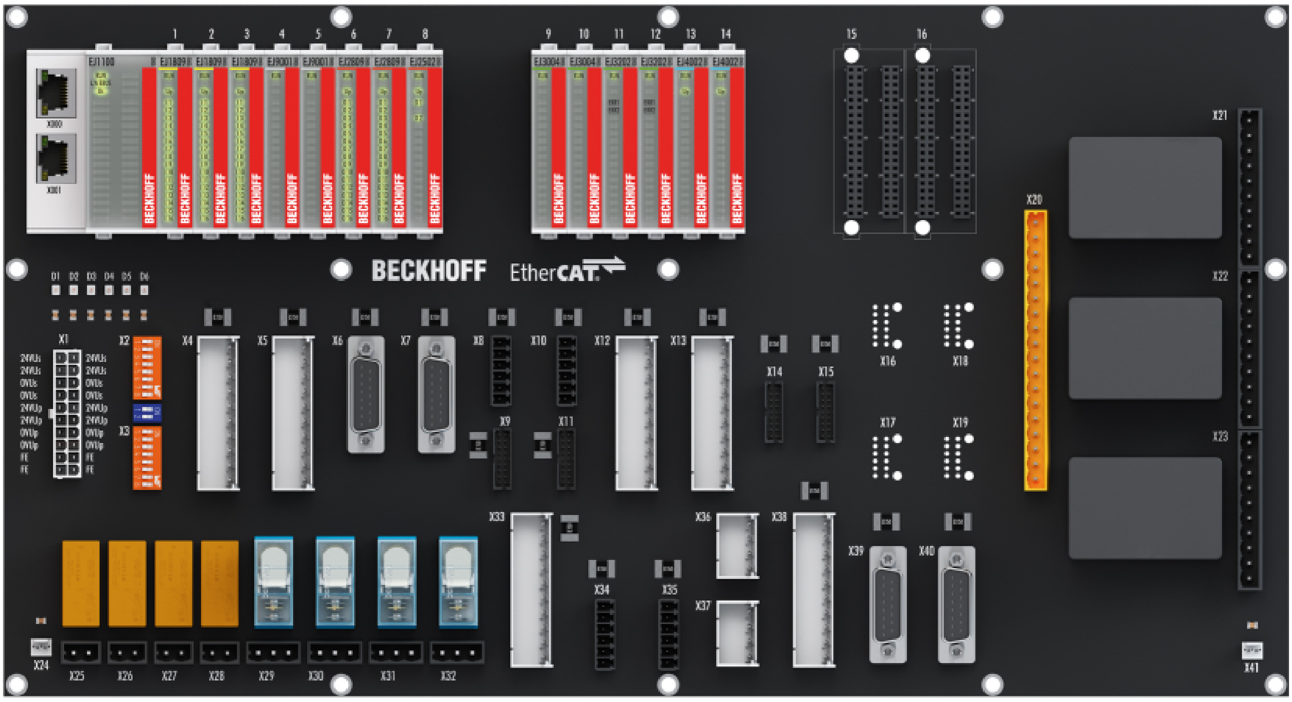


Fig. 6: EJ system sample

Signal distribution board

The signal distribution board distributes the signals and the power supply to individual application-specific plug connectors, in order to connect the controller to further machine modules. Using pre-assembled cable harnesses avoids the need for time-consuming connection of individual wires. Coded components reduce the unit costs and the risk of miswiring.
Beckhoff offers development of signal distribution boards as an engineering service. Customers have the option to develop their own signal distribution board, based on the design guide.

EtherCAT plug-in modules

Similar to the EtherCAT Terminal system, a module strand consists of a bus coupler and I/O modules. Almost all of the EtherCAT Terminals can also be manufactured in the EJ design as EtherCAT plug-in modules. The EJ modules are directly attached to the signal distribution board. The communication, signal distribution and supply take place via the contact pins at the rear of the modules and the PCB tracks of the signal distribution board. The coding pins at the rear serve as mechanical protection against incorrect connection. Color coding on the housing facilitates distinguishing of the modules.

3 EJ2564 - Product description

3.1 Introduction

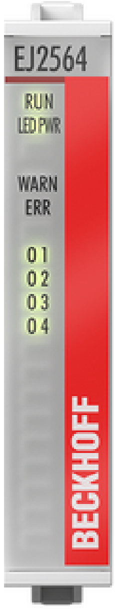


Fig. 7: EJ2564

4-channel LED output, 5...48 V_{DC}, 4 A, RGBW, common anode

The EJ2564 EtherCAT plug-in module controls LEDs with integrated series resistor via an adjustable (ground-switching) PWM signal. Common-anode RGBW LEDs can be controlled with the 4 channels. But also the operation with four LEDs of the same color is possible with the EtherCAT plug-in module.

The EJ2564 module has a flexible input voltage of 5 V_{DC}... 48 V_{DC}. The output voltage corresponds to the input voltage. For the operation of multicolor LEDs the color component per channel is adjustable. As a result, any color mixtures can be realized.

In addition to color mixing per channel, it is also possible to adjust the total brightness across all channels. Another parameter that can be set is the frequency. By setting the frequency in a range of 1 Hz ... 16,000 Hz, stroboscopic effects can be avoided. Moreover, the low frequencies allow the realization of visible flashes.

3.2 Technical data

Device functions	EJ2564
Application recommendation	General lighting, LED control, multicolor common anode LEDs
Connection technology	2-wire
Number of outputs	4
Input voltage U_{IN}	5 V_{DC}...48 V_{DC}
Load type	LED
Special features	Adjustable values: • Color component per channel (duty cycle), • Light intensity/brightness for all channels (Master Gain), • Frequency (to generate visible flashes or anti-stroboscope), • Time for dimming up/down (Ramp Time), • Scaling of the output (gamma), • Behavior in case of a bus error (Behavior in watchdog case [▸ 54])
Reverse polarity protection	-
Short-circuit proof	-

LED output	EJ2564
Output voltage U_{OUT}	U_{IN}
Output current I_{OUT}	4 A per channel (sum current limit: 15 A with supply via all contacts)
Duty cycle	0 … 100 %
Frequency	1 Hz ... 16 kHz (default: 5 kHz)

Communication	EJ2564
Configuration	via TwinCAT System Manager
Distributed Clocks	-

General data	EJ2564
Current consumption via E-bus	85 mA typ.
Weight	approx. 30 g
Dimensions (W x H x D)	approx. 12 mm x 66 mm x 55 mm
Assembly	on signal distribution board
Installation position	Standard [▸ 38]
Position of the coding pins [▸ 41]	2 and 6
Color coding	red

Environmental conditions	EJ2564
Permissible ambient temperature range during operation	0 °C ... +55 °C
Permissible ambient temperature range during storage	-25 °C ... +85 °C
Permissible relative air humidity	95 %, no condensation
Operating altitude	max. 2,000 m

Standards and approvals	EJ2564
Vibration/shock resistance	conforms to EN 60068-2-6/EN 60068-2-27 (with corresponding signal distribution board)
EMC immunity/emission	conforms to EN 61000-6-2 /EN 61000-6-4 (with corresponding signal distribution board)
Protection rating	EJ module: IP20 EJ system: dependent on the signal distribution board and housing
Approvals/markings*	CE, UKCA

*) Real applicable approvals/markings see type plate on the side (product marking).

CE approval

The CE Marking refers to the EtherCAT plug-in module mentioned above.
If the EtherCAT plug-in module is used in the production of a ready-to-use end product (PCB in conjunction with a housing), the manufacturer of the end product must check compliance of the overall system with relevant directives and CE certification.
To operate the EtherCAT plug-in modules, they must be installed in a housing.

3.3 Connection

EJ2564				
Pin#		Signal		
1	2	U_{EBUS}	U_{EBUS}	**E-Bus contacts** The power supply U_{EBUS} is provided by the coupler and supplied from the supply voltage U_S of the EtherCAT coupler.
3	4	GND	GND	
5	6	RX0+	TX1+	
7	8	RX0-	TX1-	
9	10	GND	GND	
11	12	TX0+	RX1+	
13	14	TX0-	RX1-	
15	16	GND	GND	
17	18	CH 1	CH 1	**Signals and power supply of the LEDs**
19	20	CH 2	CH 2	
21	22	CH 3	CH 3	
23	24	CH 4	CH 4	
25	26	LED Supply +	LED Supply +	
27	28	LED Supply +	LED Supply +	
29	30	LED Supply +	LED Supply -	
31	32	LED Supply -	LED Supply -	
33	34	0V Up	0V Up	**U_P-Contacts** The peripheral voltage U_P supplies the electronics on the field side.
35	36	0V Up	24V Up	
37	38	24V Up	24V Up	
39	40	SGND	SGND	

Signal	Description
U_{EBUS}	E-Bus power supply 3.3 V
GND	E-Bus GND signal Don`t connect with 0V Up!
RXn+	Positive E-Bus receive signal
RXn-	Negative E-Bus receive signal
TXn+	Positive E-Bus transmit signal
TXn-	Negative E-Bus transmit signal
CH 1 ...CH 4	Ground switching PWM output 1…4
LED Supply +	5…48 V DC supply voltage for the LEDs (this voltage does not supply the internal electronics)
LED Supply -	Ground of the LED supply voltage (internally connected to the 0 V Up-contact)
0V Up	Field side GND signal
24V Up	Field side power supply 24 V
SGND	Shield Ground

Fig. 8: EJ2564 - Connection

The PCB footprint can be downloaded from the Beckhoff homepage.

NOTE

Damage to devices possible!

Before installation and commissioning read the chapters Installation of EJ modules [▶ 34] and Commissioning [▶ 52]!

3.4 LEDs

LED No.	EJ2564
A	RUN
B	LED PWR
C	
1	WARN
2	ERR
3	
4	Out 1
5	Out 2
6	Out 3
7	Out 4
8	
9	
10	
11	
12	
13	
14	
15	
16	

Fig. 9: EJ2564-LEDs

LEDs				
LED	**Color**	**Display**	**State**	**Description**
RUN	green	off	Init	State of the EtherCAT State Machine: **INIT** = initialization of the plug-in module
		flashing	Pre - Operational	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different default settings set
		single flash	Safe - Operational	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager channels and the distributed clocks. Outputs remain in safe state
		on	Operational	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
		flickering	Bootstrap	State of the EtherCAT State Machine: **BOOTSTRAP** = function for Firmware updates of the plug-in module
PWR LED	green	on	-	The LED supply voltage is within the permissible range.
WARN	yellow	yellow	-	Warning
ERR	red	red	-	Error
Output 1 … 4	green	on	-	The LED output is active.

3.5 Usable LEDs

The EJ2564 EtherCAT plug-in module outputs the LED supply voltage PWM modulated. The current is not limited by the module. When selecting the LED or the LED strip, it is therefore important to ensure that appropriate series resistors are present or that some other current limiting device is connected upstream.

The module is particularly suitable for the use of multicolor lighting. Several differently colored LED chips are installed in one housing. Standard combinations are for example red, green and blue (RGB - Red, Green, Blue). For this combination, the EJ2564 module would need three channels, one for each color. But there are also RGB LEDs with an additional chip for white, the so-called RGBW (Red, Green, Blue, White) LEDs. All four channels of the module are then needed.

Besides the colored LEDs, there are also so-called CCT (correlated color temperature) LEDs. These are also known as "dual white" or "tunable white". Both warm white and cold white LEDs are interconnected to achieve a specific color temperature by mixing the two shades of white. This function can be used, for example, to simulate the course of the light color in the interior to the sunlight. This means that two LED chips (warm white, cold white) are installed in one housing. Therefore two channels of the EJ2564 EtherCAT plug-in module are required per CCT LED or CCT LED strip.

RGB+CCT describes a combination of the described RGB and the CCT LEDs. Five LED chips are combined in one housing (red, green, blue, warm white, cold white). RGB+CCT LEDs need two EJ2564 for control, because for the five different colors also five channels are needed.

For all color combinations it must be ensured that the LED or LED strip used has a common supply connection (anode) (so-called "common anode" LEDs), as the EJ2564 module is a ground switching PWM module.

An example of an RGB strip connected to the EtherCAT plug-in module is shown in the following figure.

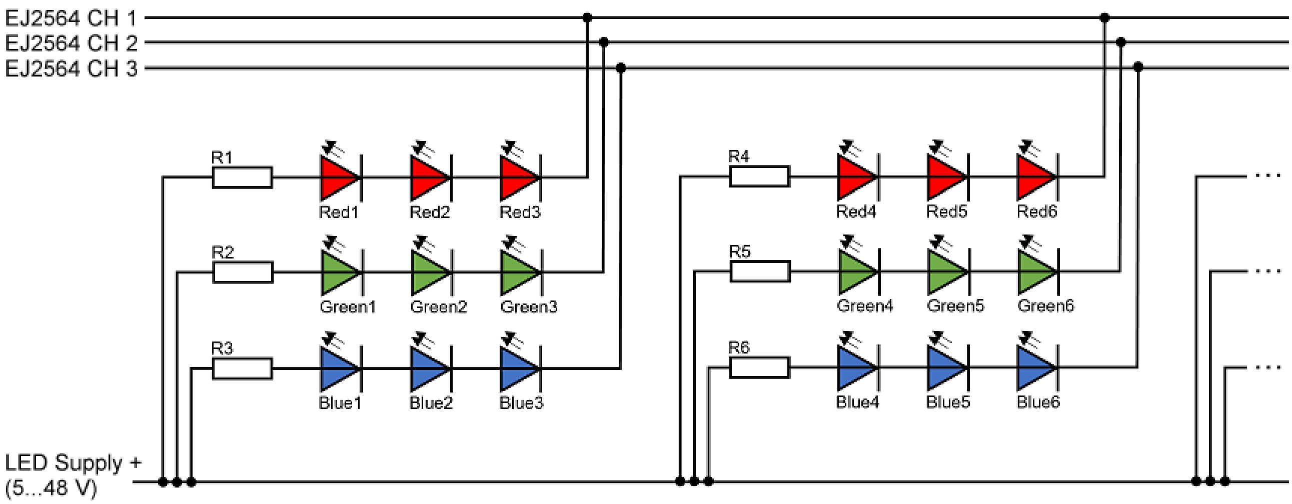


Fig. 10: Example for RGB strip connection to EJ2564

Further information on the LEDs can be found in chapter Basics of LED technology [▶ 22].

3.6 Basics of LED technology

Some basic information on light emitting diode technology (LED) is given in the following. This information is of a basic nature; therefore, please check the extent to which it applies to your application.

Beckhoff offer several products for the control of LEDs in various operating modes, including

- EL2595, EL2596: single-channel power source as an EtherCAT Terminal, also for flashing operation
- EL2564: PWM terminal for voltage operation as an EtherCAT Terminal

For others, see www.beckhoff.de

3.6.1 Definition

An LED (**l**ight-**e**mitting **d**iode) converts electrical energy into light. An LED consists of a semiconductor p-n junction. Like a conventional semiconductor diode, an LED is forward-biased and reverse-biased. When applying a voltage in the forward direction, surplus electrons in the semiconductor recombine with the electron holes. The LED becomes conductive and energy is given off in the form of light. The energy and thus the light color depend on the semiconducting material used.

The electrical circuit symbol for an LED shows a diode with two arrows. These arrows symbolize the emitted light.

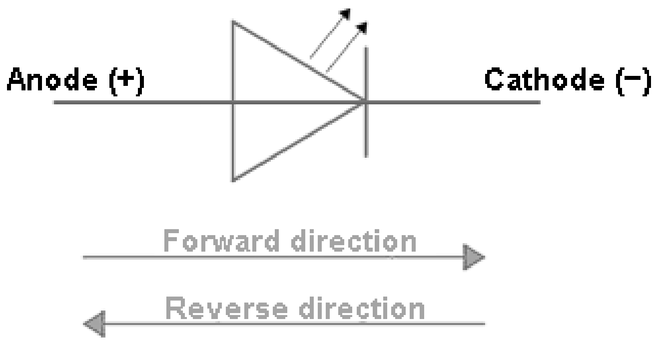


Fig. 11: Circuit symbol for an LED

3.6.2 Structure

Simple standard LEDs usually consist of an LED chip, a reflector with contact to the cathode (-), a gold wire as contact to the anode (+) and a plastic lens.

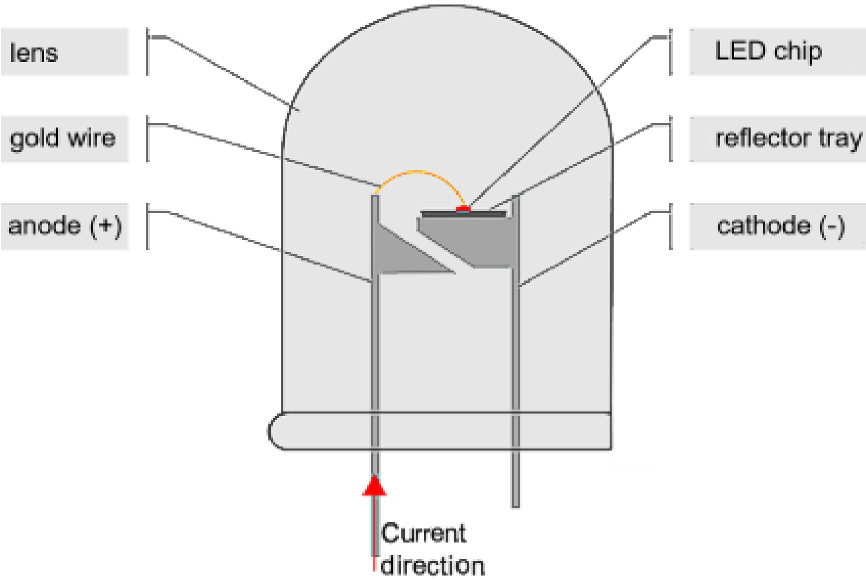


Fig. 12: Classic structure of a single-color LED

However, the structure shown above is just an example. In addition to the structure shown, there are also LEDs in high-power or SMD versions, for example.

The LED chip consists in principle of two layers. One layer has surplus electrons (n-type doping). The second, p-doped layer on the other hand has an electron deficiency; a majority of electron holes exists. This different charge distribution is achieved through the purposeful contamination (doping) of the pure semiconducting material, where other atoms such as boron or silicon are added to the semiconducting material. The illustration below shows this simplified structure of an LED chip. The emitted light is only directed by a reflector or a lens.

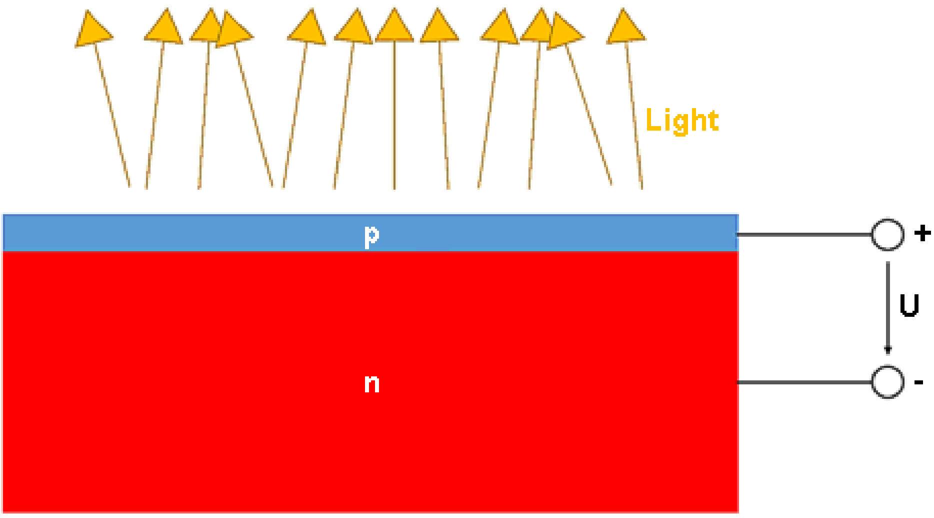


Fig. 13: Example of the structure of an LED chip

3.6.3 Properties

Forward current I_F [mA]

The forward current of an LED is the current flowing through the LED in forward direction from the anode (+) to the cathode (-). For the maximum forward current, a distinction can be made between the maximum current in continuous light mode and in pulse mode. The maximum forward current is usually higher in pulse mode than in continuous light mode.

Nominal current I_N [mA]

If the LED is operated with a forward current equal to the nominal current, the LED has the characteristics specified in its data sheet, e.g. the nominal brightness. Operation with I_F greater than I_N reduces the service life of the LED due to increased heat generation. Common nominal currents for LEDs are 20 mA, 350 mA and 1000 mA.

Conducting voltage U_D [V]

The conducting voltage indicates the level of electrical voltage required for the LED to become conductive. When the conducting voltage is applied between the anode (+) and the cathode (-), a current flows through the LED in forward direction. The conducting voltage level of an LED depends on the semiconducting material. Typical values for the conducting voltage of various LEDs are, for example, 1.6 V for red and 2.6 V for blue emitting LEDs (see Colors [▶ 30]).

Forward voltage U_F [V]

The forward voltage of an LED is the voltage applied in the forward direction between the anode (+) and the cathode (-). The forward voltage is a function of the forward current $U_F = f(I_F)$. This dependence is strongly non-linear. The relationship between U_F and I_F is shown as an example in the chapter Characteristic curve [▶ 25].

Reverse voltage U_R [V]

The reverse voltage is the electrical voltage applied to the LED in reverse direction. Data sheets usually indicate the maximum reverse voltage. This maximum reverse voltage must not be exceeded, otherwise the LED can be irreversibly damaged. A typical value for the reverse voltage of an LED is 5 V.

Typ. wavelength λ [nm]

The typical wavelength is the wavelength of the emitted light at the nominal current.

3.6.4 Characteristic curve

The characteristic curve of an LED is strongly non-linear. An LED is non-conductive if no external voltage is applied. The LED starts to conduct when the applied forward voltage U_F is at least as high than the conducting voltage U_D and the band gap is overcome by the electrons. The forward current is not proportional to the applied forward voltage. A small change in voltage can cause a large change in current. A small voltage change can lead to a strong change in light emission due to the proportionality of luminous flux and current intensity. This means that LEDs must generally be operated with a current limiter of some form or other, otherwise even slight fluctuations in the applied voltage can destroy the LED.

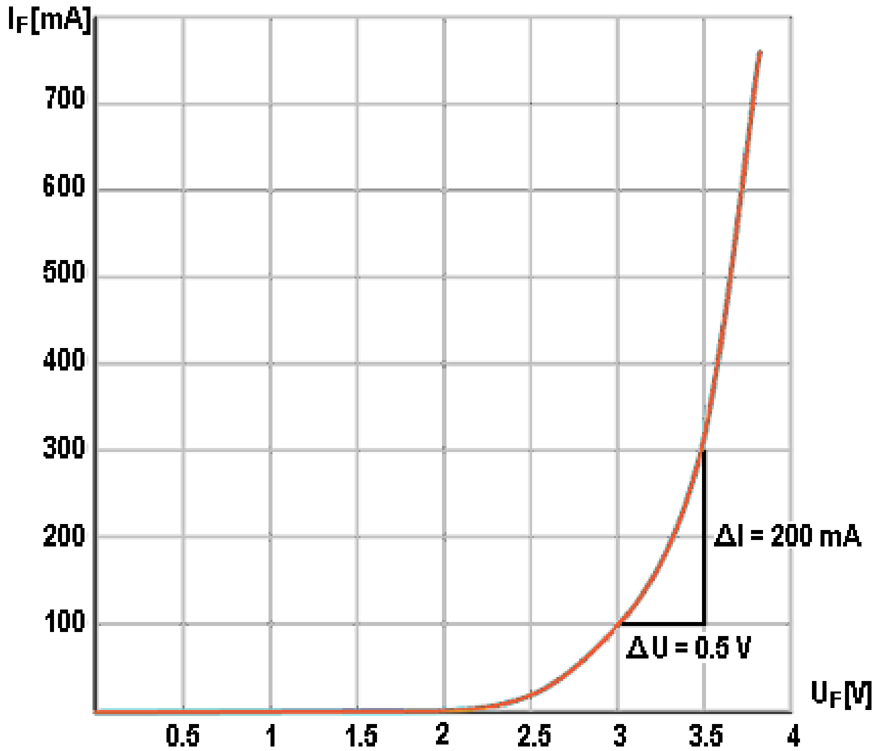


Fig. 14: Example characteristic curve of an LED

A gradient triangle is drawn on the example characteristic curve. On the basis of this gradient triangle it can be seen that a small change in the voltage of 0.5 V from 3 V to 3.5 V results in a large change in the current intensity of 200 mA from 100 mA to 300 mA. In this example case, a voltage change of less than 17 % results in a current change of 300 %.

This example shows that small voltage fluctuations lead to large changes in the current through the LED and thus to large changes in the luminous flux.

3.6.5 Control

There are two common types of control of LEDs: Current-controlled without series resistor and voltage-controlled with a series resistor. Each control method has advantages and disadvantages for certain use cases, which are explained below. Depending on the use case, a decision must be made as to which method of control is to be used.

1. Voltage mode

Voltage mode, e.g. with a battery or a power supply unit, is a simple and cost-effective way of controlling LEDs. All that is needed is an additional series resistor R_V. Due to the linear behavior of an ohmic resistance, R_V makes the overall circuit much less sensitive to voltage changes, resulting in robust LED control. Due to the ohmic resistance, however, the power loss of the control increases and is given off in the form of heat ($P_V = R_V \cdot I_{LED}^2$).

The current I_{LED} through the LED results from the ratio $I_{LED} = U/R_V$.

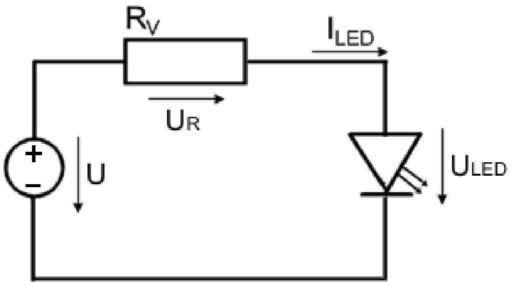


Fig. 15: Voltage-controlled LED with series resistor

The series resistor R_V is calculated as follows:

$R_V = U_R / I_{LED}$

The current through the LED is known. The voltage U_R, which is dropped across the series resistor to be calculated, is missing. This voltage is formed from the operating voltage minus the voltage dropped across the LED. The voltage U_{LED} that is dropped across the LED at I_{LED} can be read from the U/I characteristic curve for the LED in the data sheet.

$U_R = U - U_{LED}$

If the brightness of an LED with a series resistor is to be adjusted, the applied voltage must be reduced (darker) or increased (brighter).

However, the disadvantage of this type of control is that the luminous flux cannot be controlled precisely. As described at the beginning, a small change in voltage leads to a big change in current and thus to a big change in the luminous flux. In the case of voltage control, therefore, fluctuations in the supply voltage may have a direct influence on the luminous flux of the LED. It should also be borne in mind that the electrical characteristics of the resistor and the LED are temperature dependent.

- **Advantages:** simple design, simple control, the brightness of the LED can be set directly via the voltage
- **Disadvantages:** additional resistance, resulting in waste heat

2. Current mode

An LED can be operated directly if a power source (electronic circuit) is used instead of a voltage source (e.g. battery). With current control, the luminous flux of the LED can be adjusted directly via the specified current value, without resistance. Fluctuations in the supply voltage thus have no influence on the luminous flux of the LED. The luminous flux is constant and reproducible with current control. Current control is thus recommended, for example, in machine vision applications.

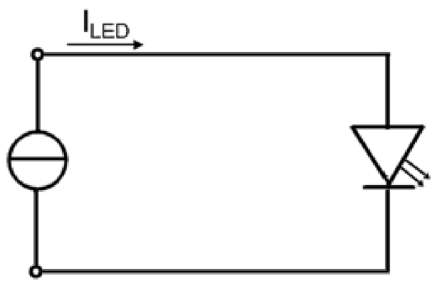


Fig. 16: Current-operated LED

- **Advantages:** no additional components; the brightness of the LED can be adjusted directly via the current
- **Disadvantages:** complex power source may be required

Glowing LED due to a leakage current

Even when the LED is switched off, a small current flow through the LED may occur, depending on the control circuit. Due to this leakage current, the LED glows noticeably in some cases.

Further information about the control of LEDs with Beckhoff components can also be found in the corresponding Application Note.

https://download.beckhoff.com/download/document/Application_Notes/DK9222-0620-0065.pdf

3.6.6 Operation modes

There are two operation modes for LEDs: continuous operation and pulse operation. Each operation mode has advantages and disadvantages, therefore a decision has to be made about which mode to use depending on the use case.

1. Continuous operation

An LED circuit can be designed for continuous operation. The LED is then switched on continuously. In this operation mode the current through the LED may not exceed the nominal current.

- **Advantages:** Simpler and cheaper circuit
- **Disadvantages:** Only a small part of the maximum possible luminous flux of the LED can be used. Continuous operation generates higher waste heat, leading to faster aging of the LED.

Continuous operation can take place in various ways:

a. Current and voltage output

The continuous switching on of a voltage or current (depending on the selected control method) leads to a continuous light. The description, as well as the advantages and disadvantages of the two control methods can be found in the chapter Control [▸ 26].

b. Pulse width modulation (PWM)

If the constant current or constant voltage with series resistor is clocked quickly in the kHz range, this is referred to as a PWM mode. The true-color brightness can then be adjusted by adapting the duty factor of the pulse width modulation (PWM). By switching the power supply on and off with a sufficiently high frequency and a preset duty cycle (0...100 %), the flashing appears to the human eye like a continuous light. By changing the duty cycle, the current averaged over time is reduced or increased by the LED, thus adjusting the brightness.

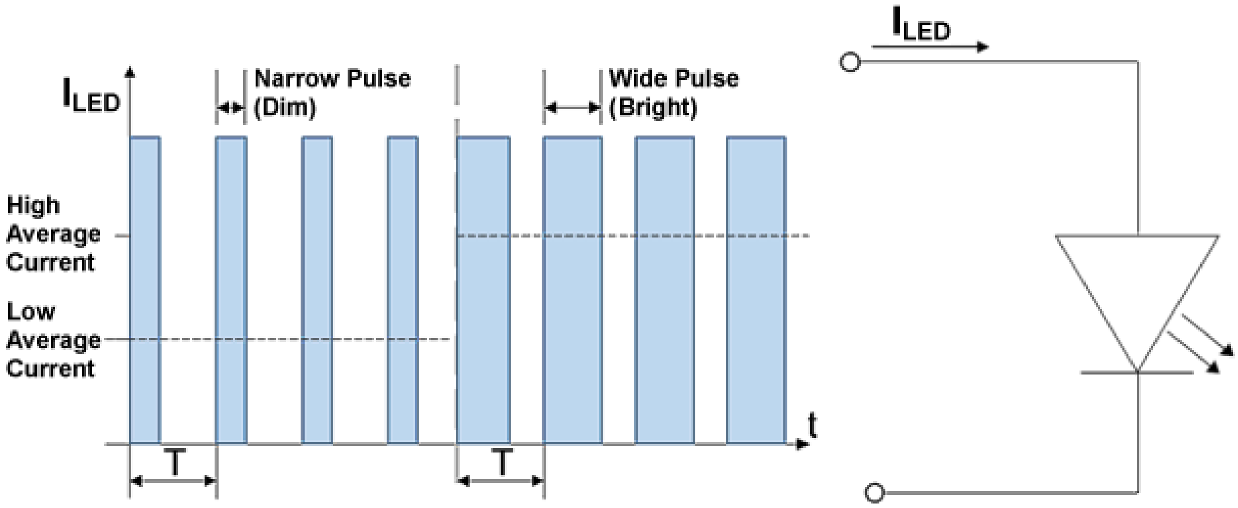


Fig. 17: Control of an LED with PWM

- **Advantages:** true-color brightness adjustment
- **Disadvantages:** supply must be able to provide rapidly increasing currents, complex supply source may be required

2. Pulse operation

In some applications an “overdrive” may be desired, as the light power in continuous operation with the nominal current is insufficient. The LED is operated briefly for a few µs to ms with a considerably higher power than in nominal operation by means of a brief and pulsating increase in the current above the nominal current. This results in briefly higher luminous fluxes. The LED can cool down again in the subsequent pause.

Overdriving leads to increased development of heat in the LEDs. The temperature of the LED chip must not increase beyond the temperature limit value during the pulse. The LED will otherwise be damaged. Following a pulse, there must be a sufficiently long pause (switch-off time) before the next pulse so that the LED can cool down. The ratio of switch-on to switch-off time is set by the duty cycle. A maximum duty cycle of 10% is often set for pulse operation. Hence, the pulse duration may not exceed 10% of the entire period. The precise values are to be taken from the manufacturer's data sheets.

$$\text{Duty Cycle} = \frac{T_{on}}{T_{on} + T_{off}} = \frac{T_{on}}{T} \leq 10\%$$

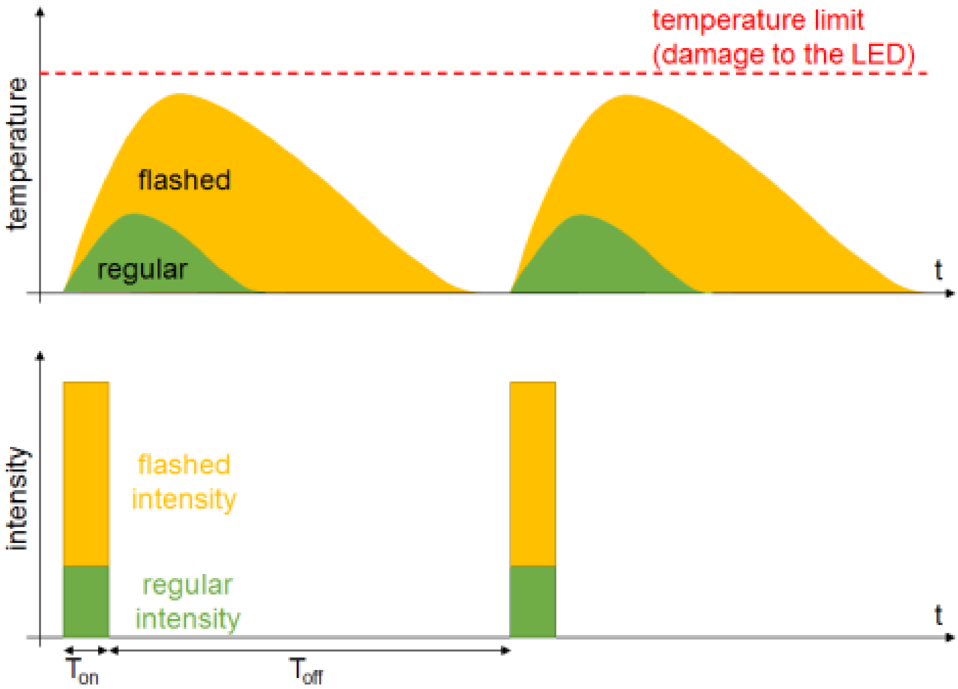


Fig. 18: Temperature and brightness as a function of time in pulse operation

- **Advantages:** The maximum luminous flux of the LED can be utilized. If the duty cycle is adhered to or undershot, the waste heat created is at the most as much as in continuous operation. If the maximum duty cycle is undershot so that the average power is less than in continuous operation, this can lead to less aging and thus to a longer service life.
- **Disadvantages:** Requires a more elaborate circuit and control solution, for example in the form of a flash controller

Here, too, implementation via a voltage or current output is possible. The output values must then be dimensioned in such a way that the output flash reaches the desired brightness. The maximum output power must always be taken into account when dimensioning the flashes.

The output of light pulses is also possible with fast PWM. A light flash with a length of 1 ms is generated with 1 kHz PWM.

3.6.7 Connection of several LEDs together

1. Series connection

The series connection of LEDs is the usual way to increase the illuminance, for example. In a series circuit, the same current flows through all consumers. It therefore makes sense if all the LEDs connected in series have the same color or, better still, are of the same type with the same characteristic values.

With a sufficiently high supply voltage, several LEDs can be connected in series. A single resistor or a current controller is then sufficient. The number of LEDs must be taken into account when calculating the series resistance, as there is a voltage drop U_{LED} across each LED, which then adds up.

2. Parallel connection

The connection of LEDs in parallel should be avoided as the U/I characteristic curve of an LED is not linear, but approximately exponential. Thus, a small change in voltage leads to a large change in current.

If two or more LEDs (with the same nominal conducting voltage) are connected in parallel, the largest current will flow through the LED with the lowest conducting voltage. As a result, this LED will be brighter and thus also warmer than the other parallel LEDs. The conducting voltage decreases as the temperature increases, as a result of which the effect is amplified and the current increases further until destruction.

Since LEDs made of different semiconducting material, i.e. with different colors, have different conducting voltages, the parallel connection of LEDs with different colors is not permitted. There is variance even in the conducting voltage of LEDs with the same color and from the same manufacturing series. When connecting LEDs in parallel, a series resistor/current controller should be used for each individual LED.

3. LED controller for pixel LEDs

The so-called pixel system is an intelligent method of LED control for several LEDs. “Pixel” LEDs are LEDs with an integrated circuit (IC). With an LED matrix or an LED strip, several LEDs are not connected classically in series; instead, each LED can receive individual signals via a bus communication. In this way, each LED can be controlled individually. These LEDs or LED strips require an LED controller, which serially transmits the communication signals with >100 kHz. Each individual LED is then assigned its own pixel controller.

3.6.8 Colors

The color of the light emitted by single-color LEDs can be set through the selection of the semiconducting material. The wavelength range of the light extends from near-infrared through the visible spectrum to the ultraviolet range. The shorter the wavelengths become, the larger the band gap of this semiconductor and the higher the conducting voltage U_D for the operation of the LED.
The following table shows sample values for colors with the associated wavelengths, possibly usable semiconducting materials and associated conducting voltages. This table merely contains example values, therefore characteristic values and materials are not fully applicable and not applicable to every LED.

Color	Wavelength λ in [nm]	Material	Conducting voltage U_D in [V]
Infrared	>760	Gallium arsenide (GaAs) Aluminum gallium arsenide (AlGaAs)	<1.6
Red	610 - 760	Aluminum gallium arsenide (AlGaAs) Gallium arsenide phosphide (GaAsP) Aluminum gallium indium phosphide (AlGaInP) Gallium phosphide (GaP)	1.6 - 1.9
Orange	590 - 610	Gallium arsenide phosphide (GaAsP) Aluminum gallium indium phosphide (AlGaInP) Gallium phosphide (GaP)	1.8 - 2.2
Yellow	570 - 590	Gallium arsenide phosphide (GaAsP) Aluminum gallium indium phosphide (AlGaInP) Gallium phosphide (GaP)	2.0 - 2.4
Green	500 - 570	Indium gallium nitride (InGaN) Gallium nitride (GaN) Gallium phosphide (GaP) Aluminum gallium indium phosphide (AlGaInP) Aluminum gallium phosphide (AlGaP)	2.2 - 2.7
Blue	450 - 500	Zinc selenide (ZnSe) Indium gallium nitride (InGaN) Silicon carbide (SiC)	2.6 - 3.3
Violet	400 - 450	Indium gallium nitride (InGaN)	3.2 - 3.6
Ultraviolet	230 - 400	Aluminum nitride (AlN) Aluminum gallium nitride (AlGaN) Aluminum gallium indium nitride (AlGaInN)	3.5 - 4.2

LEDs can generally only generate light in a small wavelength range with a width of a few tens of nanometers. White light is the sum of all colors or the sum of all wavelengths in the visible range. Therefore, colors must be mixed additively in order to generate white light with an LED. There are various methods of doing this, of which two essential methods are described below.

1. **Combination of different colored LEDs**
   Red, green and blue LEDs (RGB LEDs) can be combined with one another in a housing so that the colors mix in order to generate white light. If the LEDs are controlled appropriately, the light appears to be white. In the RGB combination of LEDs it is also possible through appropriate control of the individual LEDs to generate light of a different color with continuous color transitions.
2. **Luminescence**
   A short-wave LED (blue, violet, ultraviolet) is combined with photoluminescent dye. Photoluminescence describes the emission of light after excitation by light – usually blue or ultraviolet. The dye converts blue, higher-energy light into longer-wave light with a typically larger wavelength range. The dye used significantly influences the color temperature, so that different white tones (Cold White, Warm White) can be generated.

As the duration of use of LEDs increases, the color of the emitted light changes due to aging. These color changes proceed differently with each LED. In LEDs that emit white light by means of a photoluminescent dye, both the LED chip and the dye itself age.

3.6.9 Typical designs of multi-color LEDs

There are generally two types of LEDs, monochrome and multicolor. With monochrome LEDs, it is possible to adjust the brightness via the current in the forward direction, but the color is unchangeable as the LED is made of only one semiconducting material and therefore emits a specific wavelength. The color of the LED is not affected by the control mode. With the multi-colored LEDs, there are different types with different color options. An n-color LED consists of n individual semiconductor PN transitions combined in one housing. The individual LEDs in the multi-color LED consist of the corresponding semiconducting material, which emits the corresponding wavelength. The types RGB (red-green-blue), RGBW (red-green-blue-white) and RGBWW (red-green-blue-white-white) are common. The exact color emitted is determined by the current through the individual semiconductor transitions.

Monochrome LEDs differ in their semiconducting material, resulting in different characteristic values and colors.

The characteristic values and colors also differ with multi-color LEDs. In the case of multi-color LEDs, however, it is additionally necessary to consider how the individual monochrome LEDs are interconnected within the multi-color LED light source. Some of the possible interconnections are illustrated and explained below:

1. Inverse parallel

The “Inverse parallel” interconnection only works with two (differently colored) LEDs. With this interconnection it is possible to create different color mixtures with two LEDs.

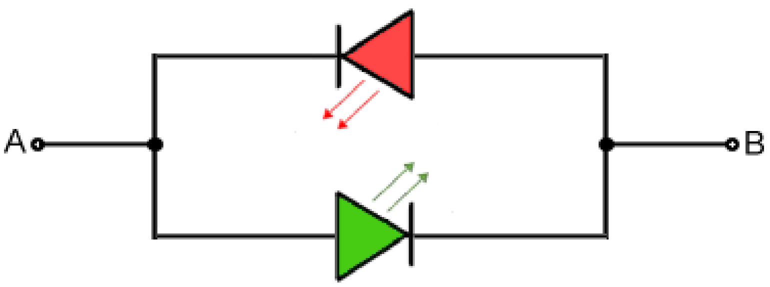


Fig. 19: Inverse parallel LEDs

When the current flows from A to B, a green light is output because the green LED is operated in the forward direction. From B to A, the red LED would light up. Since the different colors have different conducting voltages, each LED requires its own series resistor or current controller. With bidirectional current, the two

LEDs would light up alternately. If the current direction changes with a significantly shorter period than the exposure time of a camera, the individual colors of the LED mix to a mixed color. To the human eye, the colors also appear mixed in the case of a quick change.

This type of arrangement of two LEDs is used, for example, to indicate polarity, e.g. for the correct connection of batteries or power supplies.

2. Common anode

The “common anode” interconnection can be used to combine any number of LEDs. This method is common with many RGB/RGBW LEDs. In addition to switching the individual LEDs on and off, only a low current may flow through some of them. This makes any color mixture possible.

All LEDs have a common positive potential at the anode (+). In order to operate an LED in the forward direction, a lower potential must be applied to the cathode connection of the desired color than to the anode connection. If the potential at the cathode (-) is higher, the LED is operated in the reverse direction. Caution: LEDs often have very low reverse voltages of only a few volts!

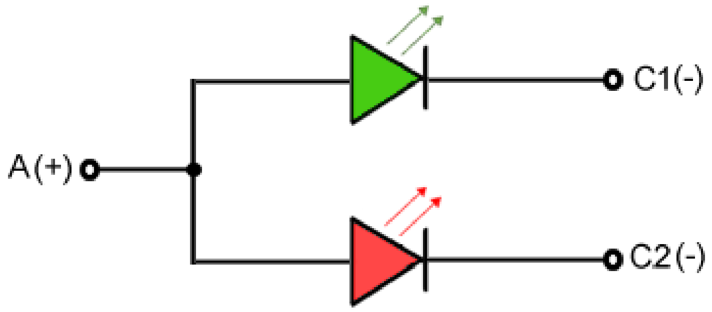


Fig. 20: Common anode LEDs

3. Common cathode

The operation of LEDs with a common cathode (-) is similar to that with a common anode (see “Common anode”). This method is used less often than common anode. Here, too, any number of LEDs can be combined in different colors. Any color can be generated by switching the LEDs on and off differently.

With the “common cathode” interconnection, all LEDs have a common negative potential. In order to switch on an LED, a higher potential must be applied to its anode (+) than to the cathode (-).

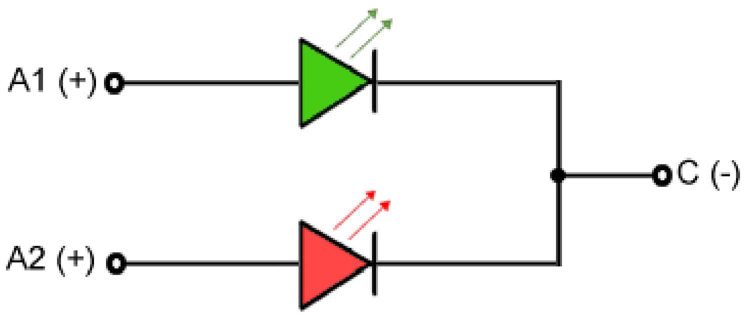


Fig. 21: Common cathode LEDs

3.6.10 Temperature and aging

As with all semiconductors, the properties of an LED are temperature-dependent. Typical changes occur mainly in the luminosity, the wavelength of the emitted light and the conducting voltage.

1. **Luminous flux**
   An increasing temperature in the LED chip leads to a reduction in the luminous flux.

2. **Wavelength λ**
   An increasing temperature in the LED chip leads to an increase in the wavelength (the extent depends on the semiconducting material)
3. **Conducting voltage U_D**
   An increasing temperature in the LED chip leads to a reduction in the conducting voltage (2 mV/°C). In contrast, the conducting voltage increases at low temperatures. A reduction in the conducting voltage leads to an increase in the current. As the current increases, the temperature of the LED chip continues to rise. This leads to a further drop in the conducting voltage.

LED circuits must be sufficiently dimensioned or cooled to prevent temperature-related changes in the current from causing damage or shortening of service life.

With falling temperatures, the current would be reduced by the increasing conducting voltage. This could lead to the required luminosity not being achieved.

The aging of LEDs is approximately exponential. The speed of aging depends on the respective semiconducting material and the operating conditions (temperature, current). If LEDs are operated at the usage limits (maximum forward voltage, maximum forward current, maximum operating temperature), the service life of the LED is shortened. The aging of LEDs is reflected in the reduction of luminosity and a change in the color temperature.

4 Installation of EJ modules

4.1 Power supply for the EtherCAT plug-in modules

⚠ **WARNING**

Power supply from SELV/PELV power supply unit!

SELV/PELV circuits (Safety Extra Low Voltage, Protective Extra Low Voltage) according to IEC 61010-2-201 must be used to supply this device.

Notes:

- SELV/PELV circuits may give rise to further requirements from standards such as IEC 60204-1 et al, for example with regard to cable spacing and insulation.
- A SELV (Safety Extra Low Voltage) supply provides safe electrical isolation and limitation of the voltage without a connection to the protective conductor,
  a PELV (Protective Extra Low Voltage) supply also requires a safe connection to the protective conductor.

The signal distribution board should have a power supply designed for the maximum possible current load of the module string. Information on the current required from the E-bus supply can be found for each module in the respective documentation in section “Technical data”, online and in the catalog. The power requirement of the module string is displayed in the TwinCAT System Manager.

E-bus power supply with EJ1100 or EJ1101-0022 and EJ940x

The EJ1100 Bus Coupler supplies the connected EJ modules with the E-bus system voltage of 3.3 V. The Coupler can accommodate a load up to 2.2 A. If a higher current is required, a combination of the coupler EJ1101-0022 and the power supply units EJ9400 (2.5 A) or EJ9404 (12 A) should be used. The EJ940x power supply units can be used as additional supply modules in the module string.

Depending on the application, the following combinations for the E-bus supply are available:

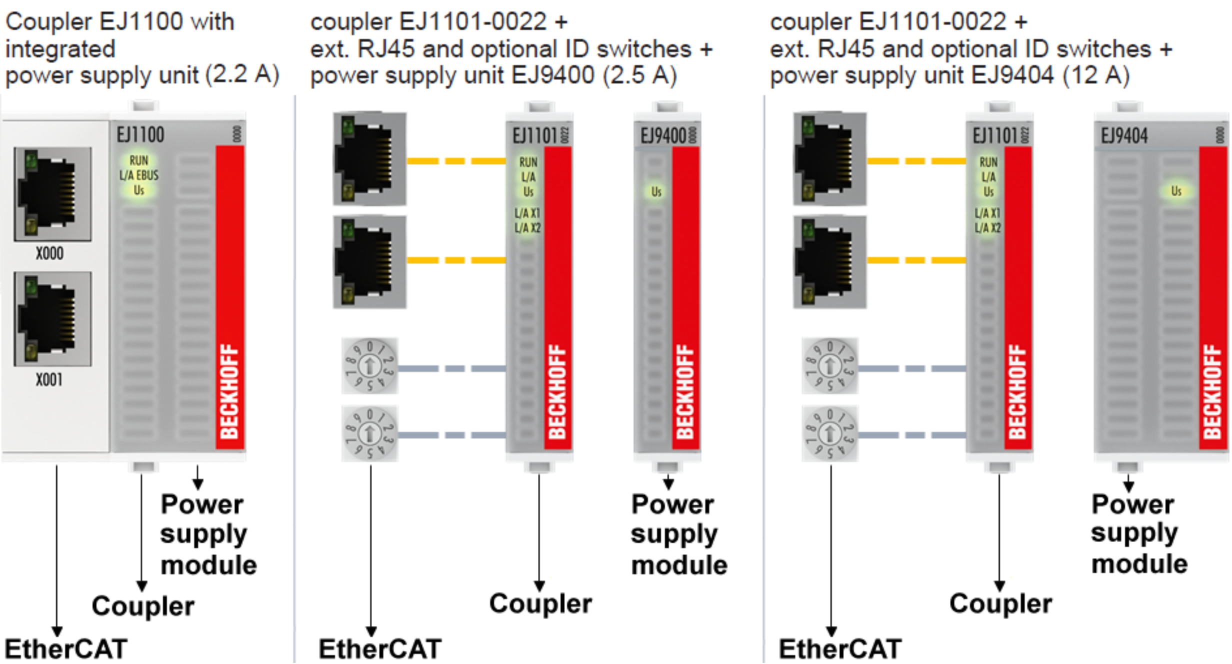


Fig. 22: E-bus power supply with EJ1100 or EJ1101-0022 + EJ940x

In the EJ1101-0022 coupler, the RJ45 connectors and optional ID switches are external and can be positioned anywhere on the signal distribution board, as required. This facilitates feeding through a housing.

The EJ940x power supply plug-in modules provide an optional reset function (see chapter Connection of the documentation for EJ9400 and EJ9404)

E-bus power supply with CXxxxx and EK1110-004x

The Embedded PC supplies the attached EtherCAT Terminals and the EtherCAT EJ coupler

- with a supply voltage Us of 24 V_{DC} (-15 %/+20%). This voltage supplies the E-bus and the bus terminal electronics.
  The CXxxxx units supply the E-bus with up to 2,000 mA E-bus current. If a higher current is required due to the attached terminals, power feed terminals or power supply plug-in modules must be used for the E-bus supply.
- with a peripheral voltage Up of 24 V_{DC} to supply the field electronics.

The EK1110-004x EtherCAT EJ couplers relay the following parameters to the signal distribution board via the rear connector:

- the E-bus signals,
- the E-bus voltage U_{EBUS} (3.3 V) and
- the peripheral voltage U_P (24 V_{DC}).

Fig. 23: PCB with Embedded PC, EK1110-0043 and EJxxxx, rear view EK1110-0043

4.2 EJxxxx - dimensions

The EJ modules are compact and lightweight thanks to their design. Their volume is approx. 50% smaller than the volume of the EL terminals. A distinction is made between four different module types, depending on the width and the height:

Module type	Dimensions (W x H x D)	Sample in figure below
Coupler	44 mm x 66 mm x 55 mm	EJ1100 (ej_44_2xrj45_coupler)
Single module	12 mm x 66 mm x 55 mm	EJ1809 (ej_12_16pin_code13)
Double module	24 mm x 66 mm x 55 mm	EJ7342 (ej_24_2x16pin_code18)
Single module (long)	12 mm x 152 mm x 55 mm	EJ1957 (ej_12_2x16pin_extended_code4747)

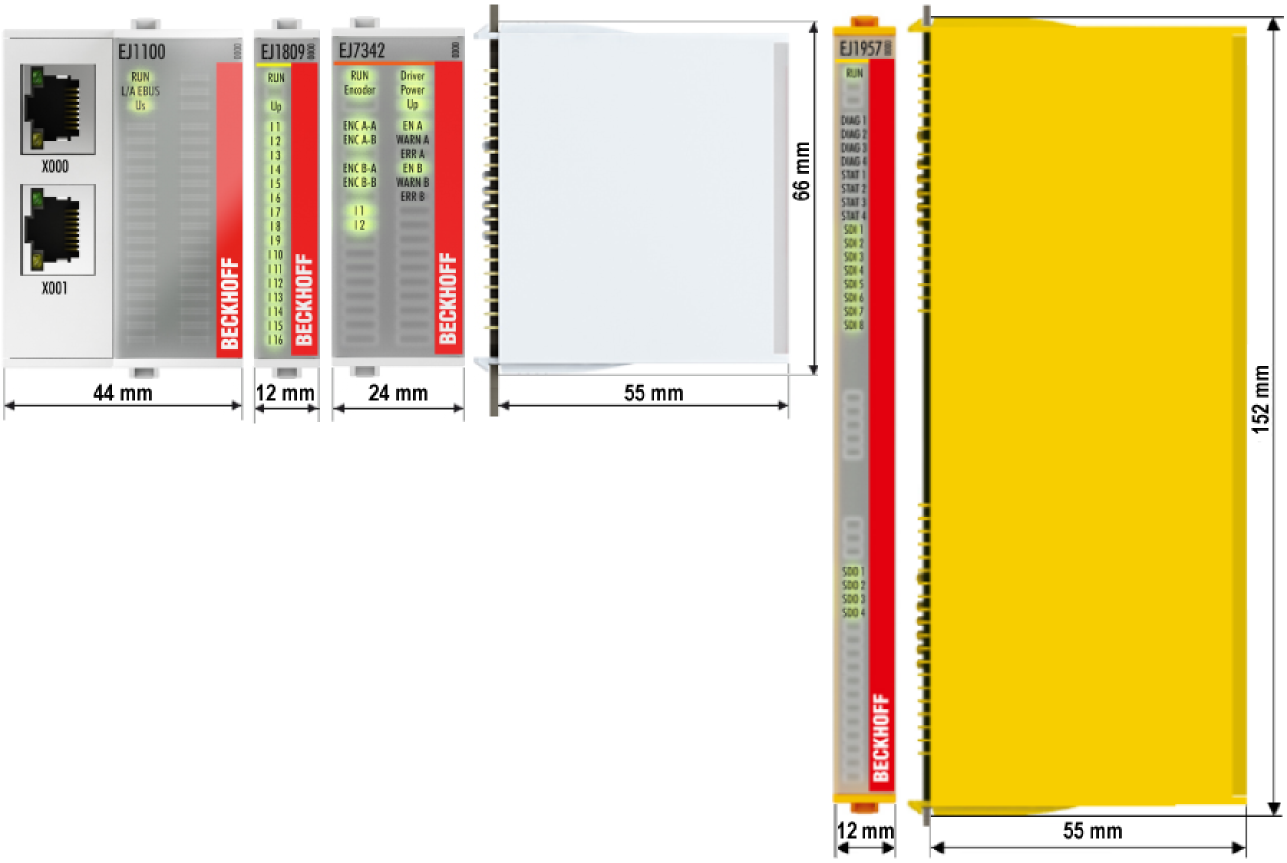


Fig. 24: EJxxxx - Dimensions

The technical drawings can be downloaded from the Beckhoff homepage. The drawings are named as described in the drawing below.

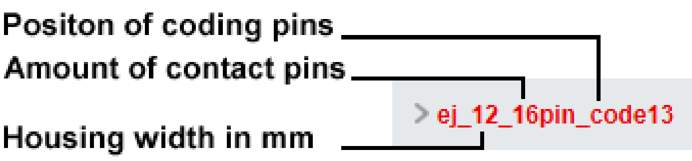


Fig. 25: Naming of the technical drawings

4.3 Installation positions and minimum distances

4.3.1 Minimum distances for ensuring installability

Note the dimensions shown in the following diagram for the design of the signal distribution board to ensure safe latching and simple assembly / disassembly of the modules.

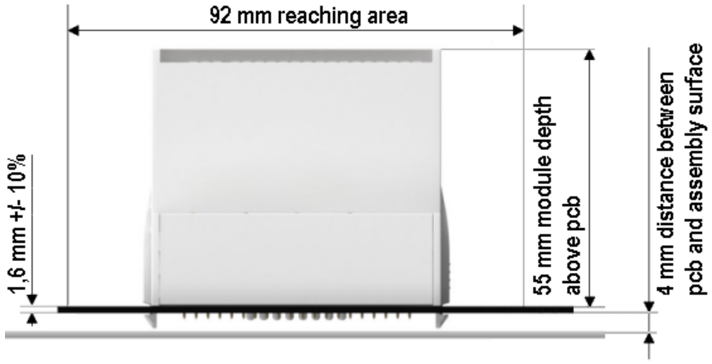


Fig. 26: Mounting distances EJ module - PCB

Observing the reaching area

A minimum reaching area of 92 mm is required for assembly / disassembly, in order to be able to reach the mounting tabs with the fingers.

Adherence to the recommended minimum distances for ventilation (see section Installation position [▸ 38]) ensures an adequate reaching area.

The signal distribution board must have a thickness of 1.6 mm and a minimum distance of 4 mm from the mounting surface, in order to ensure latching of the modules on the board.

4.3.2 Installation positions

NOTE

Constraints regarding installation position and operating temperature range

Please refer to the technical data [▶ 17] for the installed components to ascertain whether any restrictions regarding the mounting position and/or the operating temperature range have been specified. During installation of modules with increased thermal dissipation, ensure adequate distance above and below the modules to other components in order to ensure adequate ventilation of the modules during operation!

The standard installation position is recommended. If a different installation position is used, check whether additional ventilation measures are required.

Ensure that the specified conditions (see Technical data) are adhered to!

Optimum installation position (standard)

For the optimum installation position the signal distribution board is installed horizontally, and the fronts of the EJ modules face forward (see Fig. *Recommended distances for standard installation position*). The modules are ventilated from below, which enables optimum cooling of the electronics through convection. “From below” is relative to the acceleration of gravity.

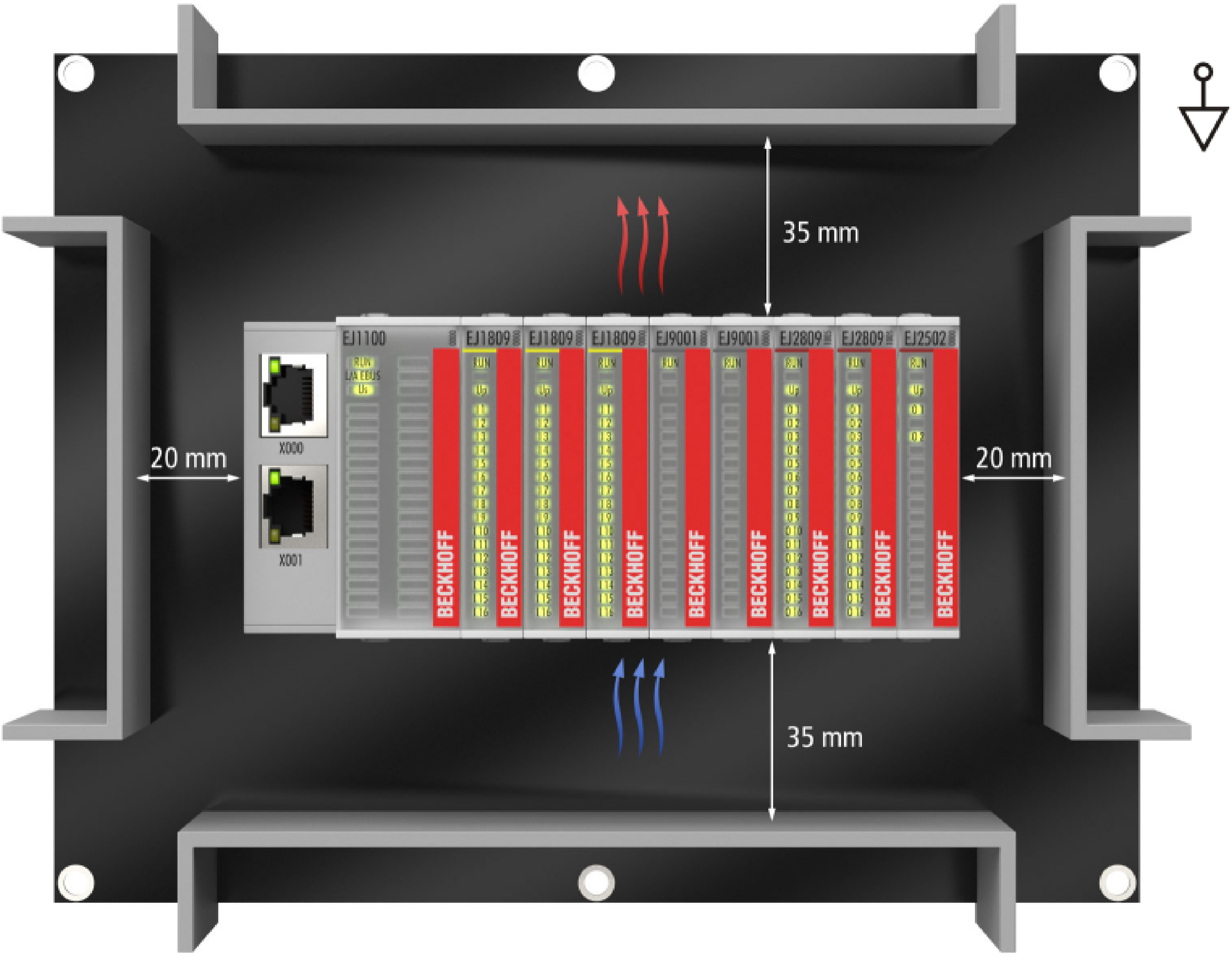


Fig. 27: Recommended distances for standard installation position

Compliance with the distances shown in Fig. *Recommended distances for standard installation position* is recommend. The recommended minimum distances should not be regarded as restricted areas for other components. The customer is responsible for verifying compliance with the environmental conditions described in the technical data. Additional cooling measures must be provided, if required.

Other installation positions

All other installation positions are characterized by a different spatial position of the signal distribution board, see Fig. *Other installation positions.*

The minimum distances to ambient specified above also apply to these installation positions.

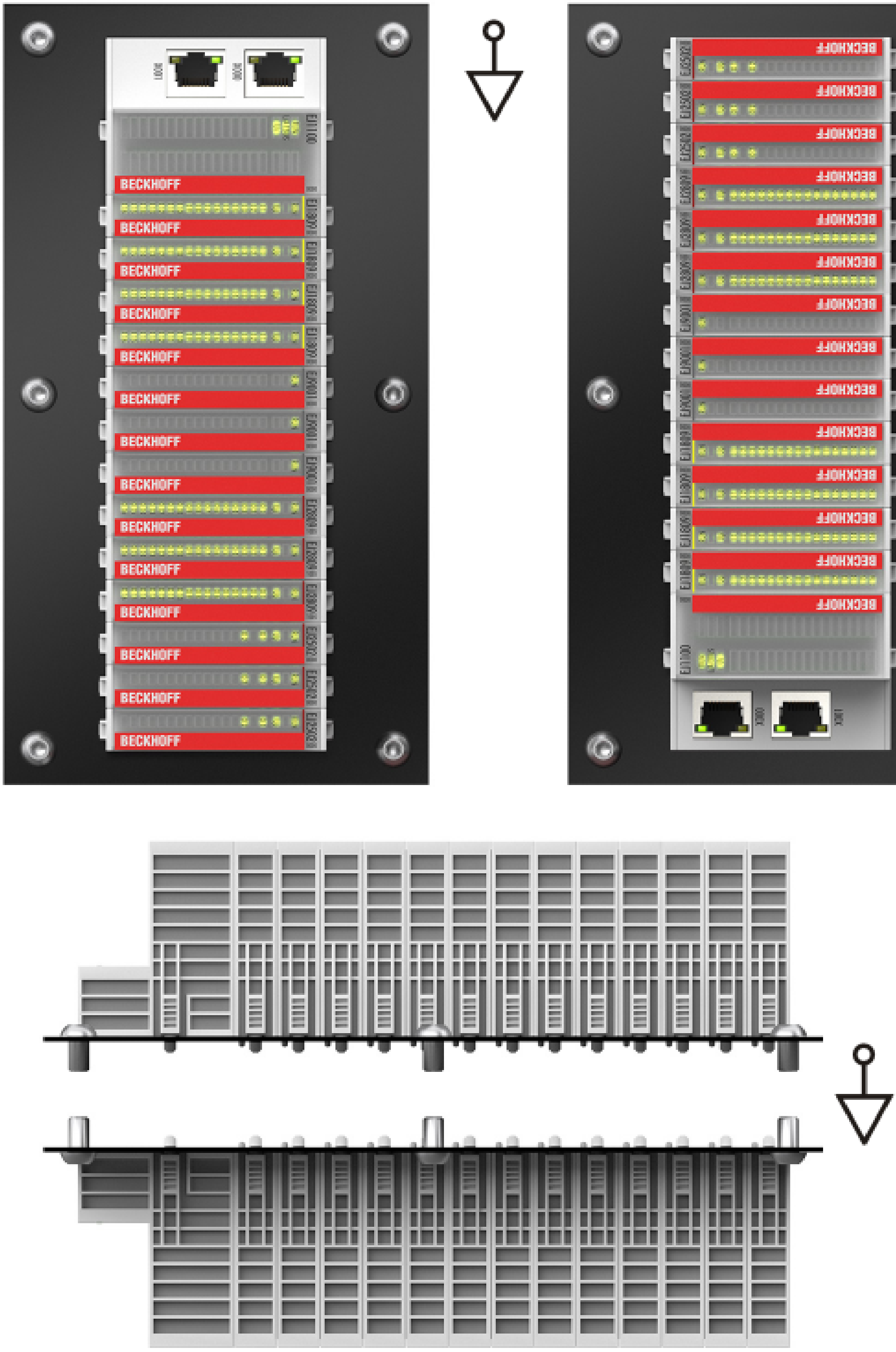


Fig. 28: Other installation positions

4.4 Codings

4.4.1 Color coding

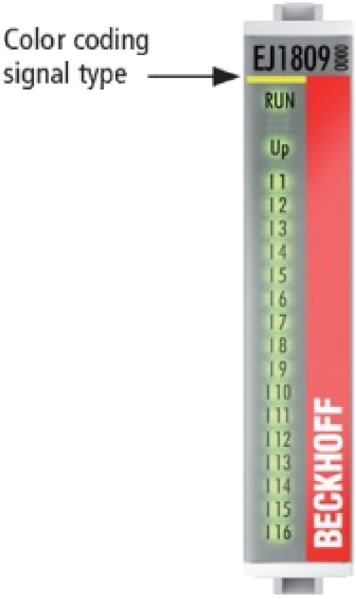


Fig. 29: EJ modules color code; sample: EJ1809

The EJ modules are color-coded for a better overview in the control cabinet (see diagram above). The color code indicates the signal type. The following table provides an overview of the signal types with corresponding color coding.

Signal type	Modules	Color
Coupler	EJ11xx	No color coding
Digital input	EJ1xxx	Yellow
Digital output	EJ2xxx	Red
Analog input	EJ3xxx	Green
Analog output	EJ4xxx	Blue
Position measurement	EJ5xxx	grey
Communication	EJ6xxx	grey
Motion	EJ7xxx	orange
System	EJ9xxx	grey

4.4.2 Mechanical position coding

The modules have two signal-specific coding pins on the underside (see Figs. B1 and B2 below). In conjunction with the coding holes in the signal distribution board (see Figs. A1 and A2 below), the coding pins provide an option for mechanical protection against incorrect connection. This significantly reduces the risk of error during installation and service.
Couplers and placeholder modules have no coding pins.

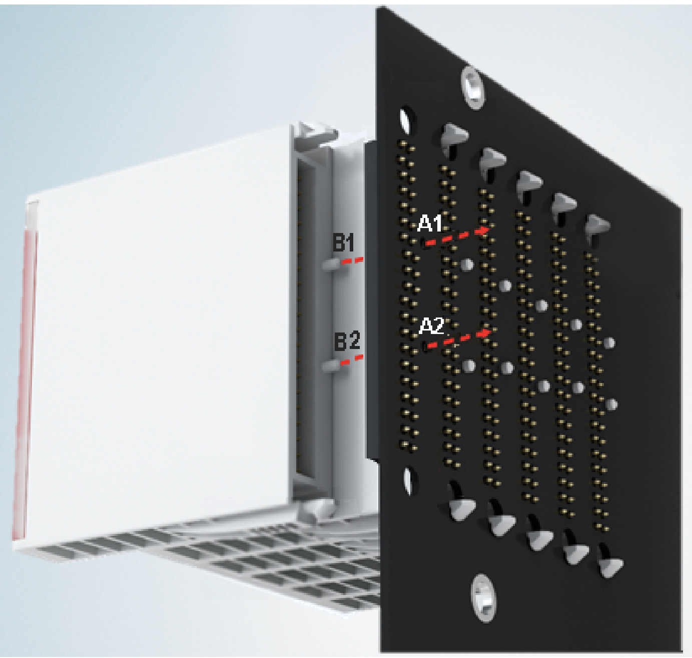


Fig. 30: Mechanical position coding with coding pins (B1 and B2) and coding holes (A1 and A2)

The following diagram shows the position of the position coding with position numbers on the left-hand side. Modules with the same signal type have the same coding. For sample, all digital input modules have the coding pins at positions one and three. There is no plug protection between modules with the same signal type. During installation the module type should therefore be verified based on the device name.

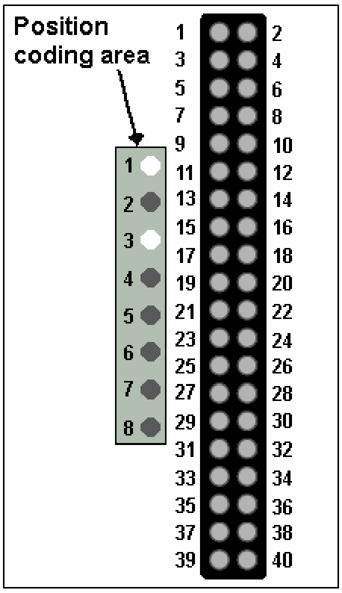


Fig. 31: Pin coding; sample: digital input modules

4.5 Installation on the signal distribution board

EJ modules are installed on the signal distribution board. The electrical connections between coupler and EJ modules are realized via the pin contacts and the signal distribution board.

The EJ components must be installed in a control cabinet or enclosure which must provide protection against fire hazards, environmental conditions and mechanical impact.

⚠ WARNING

Risk of injury through electric shock and damage to the device!

Bring the module system into a safe, de-energized state before starting installation, disassembly or wiring of the modules.

NOTE

Risk of damage to components through electrostatic discharge!

Observe the regulations for ESD protection.

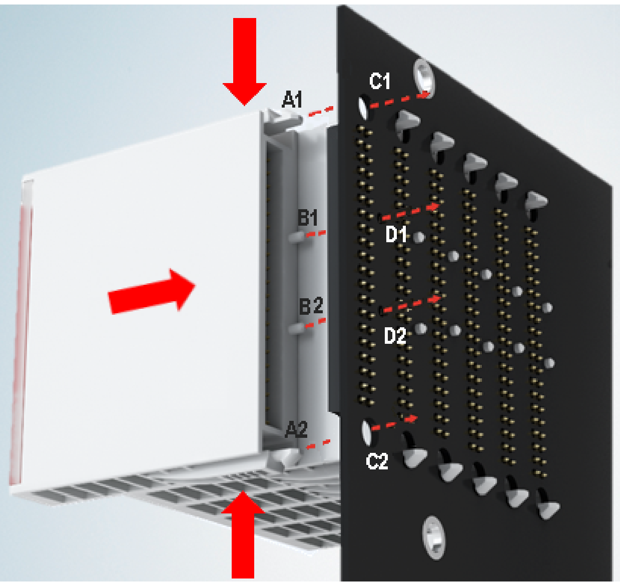


Fig. 32: Installation of EJ modules

A1 / A2	Latching lugs top / bottom	C1 / C2	Mounting holes
B1 / B2	Coding pins	D1 / D2	Coding holes

To install the modules on the signal distribution board proceed as follows:

1. Before the installation, ensure that the signal distribution board is securely connected to the mounting surface. Installation on an unsecured signal distribution board may result in damage to the board.
2. If necessary, check whether the positions of the coding pins (B) match the corresponding holes in the signal distribution board (D).
3. Compare the device name on the module with the information in the installation drawing.
4. Press the upper and the lower mounting tabs simultaneously and push the module onto the board while gently moving it up and down, until the module is latched securely.
   The required contact pressure can only be established and the maximum current carrying capacity ensured if the module is latched securely.
5. Use placeholder modules (EJ9001) to fill gaps in the module strand.

NOTE

- During installation ensure safe latching of the modules on the signal distribution board! The consequences of inadequate contact pressure include:

⇨ loss of quality of the transferred signals,

⇨ increased power dissipation of the contacts,

⇨ impairment of the service life.

4.6 Extension options

Three options are available for modifications and extensions of the EJ system.

- Replacing the placeholder modules with the function modules provided for the respective slot
- Assigning function modules specified for the respective slots for the reserve slots at the end of the module string
- Linking with EtherCAT Terminals and EtherCAT Box modules via an Ethernet/EtherCAT connection

4.6.1 Using placeholder modules for unused slots

The EJ9001 placeholder modules are used to close temporary gaps in the module strands (see Fig. A1 below). Gaps in the module strand cause interruption in EtherCAT communication and must be equipped with placeholder modules.
In contrast to the passive terminals of the EL series, the placeholder modules actively participate in the data exchange. Several placeholder modules can therefore be connected in series, without impairing the data exchange.
Unused slots at the end of the module strand can be left as reserve slots (see Fig. B1 below).

The machine complexity is extended (extended version) by allocating unused slots (see Figs. A2 below - Exchanging placeholder modules and B2 - Assigning reserve slots) according to the specifications for the signal distribution board.

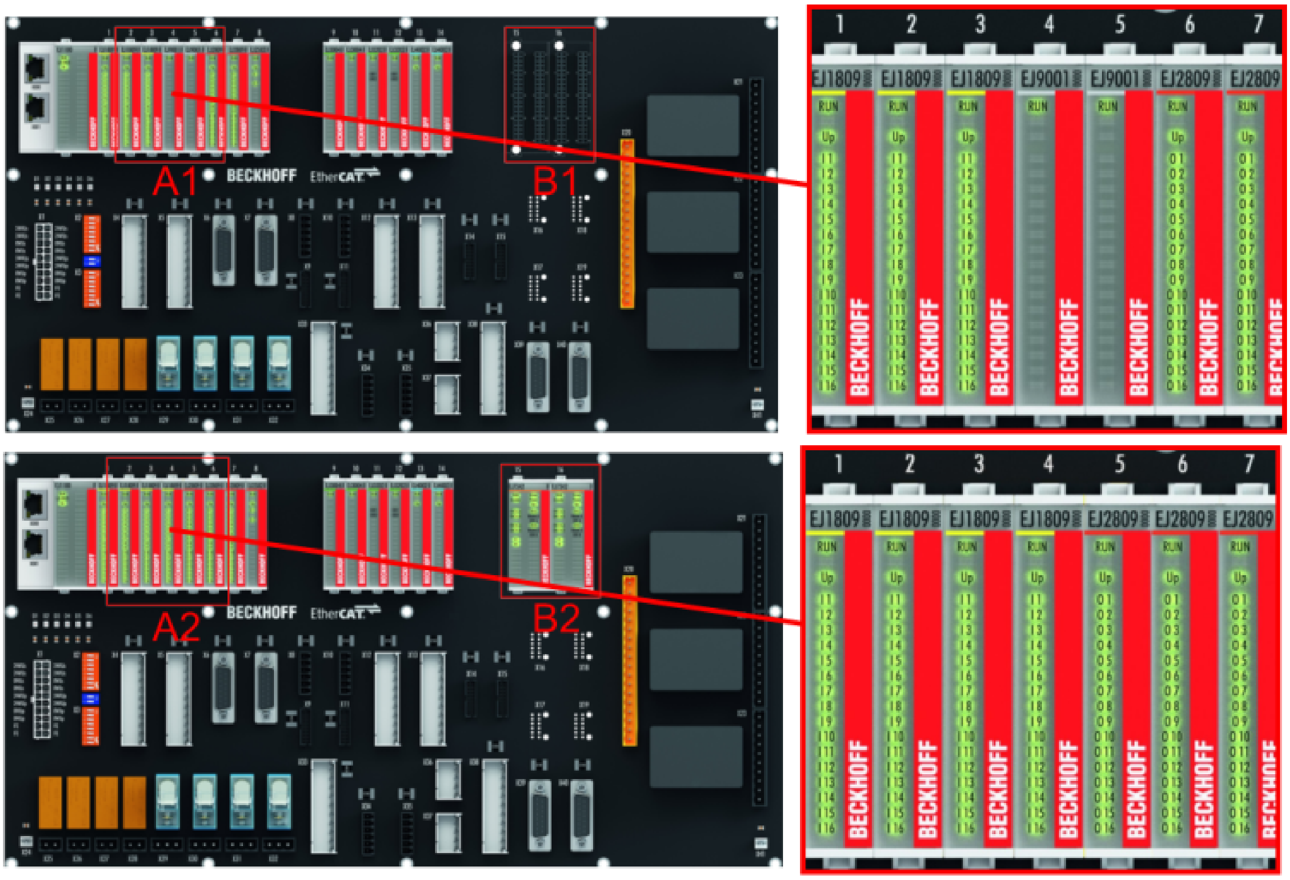


Fig. 33: Sample: Exchanging placeholder modules and assigning reserve slots

E-bus supply

Exchange the placeholder modules with other modules changes the current input from the E-Bus. Ensure that adequate power supply is provided.

4.6.2 Linking with EtherCAT Terminals and EtherCAT Box modules via an Ethernet/EtherCAT connection

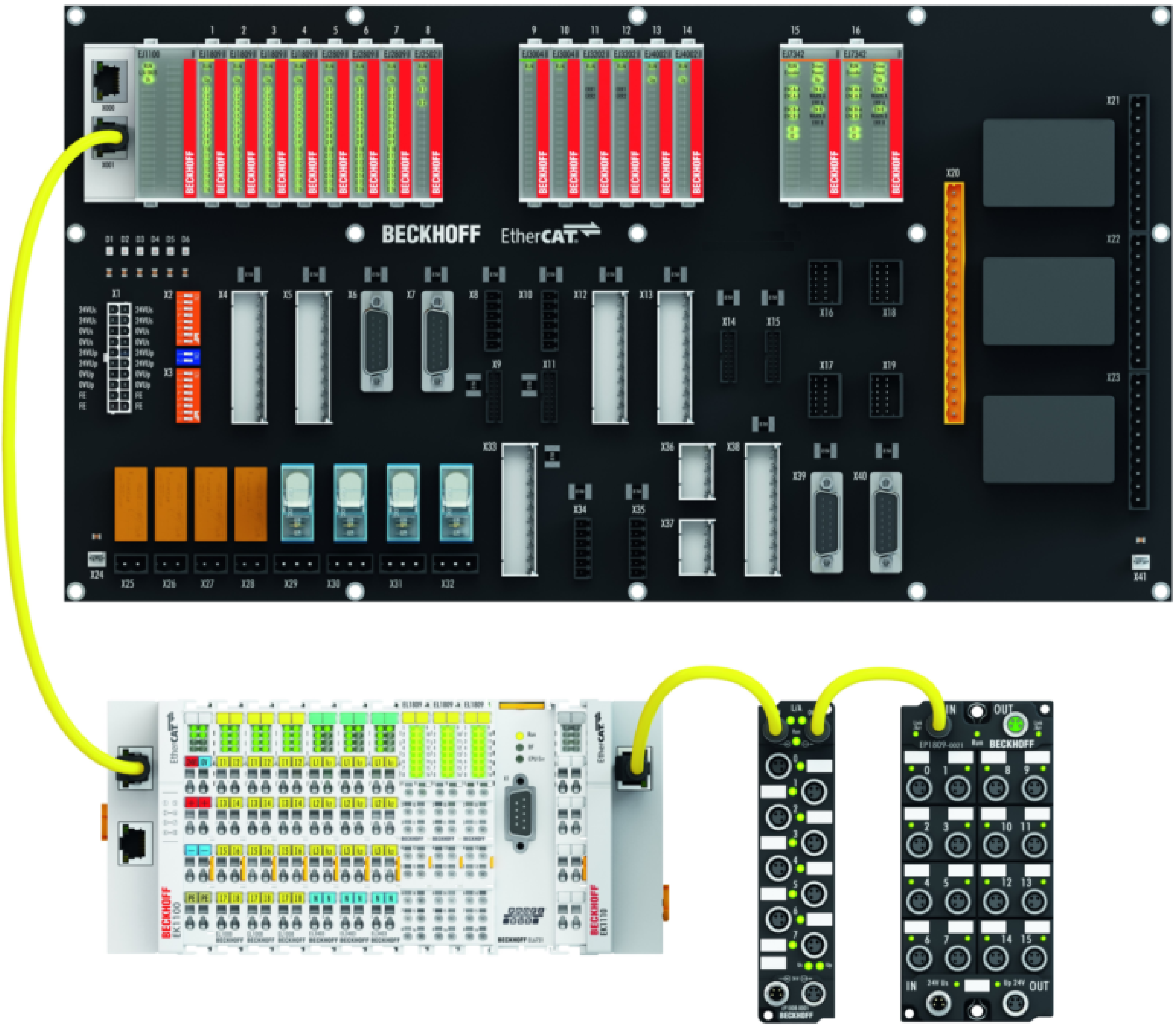


Fig. 34: Example of extension via an Ethernet/EtherCAT connection

4.7 IPC integration

Connection of CX and EL terminals via the EK1110-004x EtherCAT EJ coupler

The EK1110-0043 and EK1110-0044 EtherCAT EJ couplers connect the compact DIN-rail PCs of the CX series and attached EtherCAT Terminals (ELxxxx) with the EJ modules on the signal distribution board.

The EK1110-004x are supplied from the power supply unit of the Embedded PC.
The E-bus signals and the supply voltage of the field side U_P are routed directly to the PCB via a plug connector at the rear of the EtherCAT EJ couplers.

Due to the direct coupling of the Embedded PC and the EL terminals with the EJ modules on the PCB, no EtherCAT Extension (EK1110) or EtherCAT Coupler (EJ1100) is required.

The Embedded PC can be expanded with EtherCAT Terminals that are not yet available in the EJ system, for example.

Fig. 35: Example PCB with Embedded PC, EK1110-0043 and EJxxxx, rear view EK1110-0043

Connection of C6015 / C6017 via the EJ110x-00xx EtherCAT Coupler

Thanks to their ultra-compact design and versatile mounting options, the C6015 and C6017 IPCs are ideally suited for connection to an EJ system.

In combination with the ZS5000-0003 mounting set, it is possible to place the C6015 and C6017 IPCs compactly on the signal distribution board.
The EJ system is optimally connected to the IPC via the corresponding EtherCAT Cable (see following Fig. [A]).
The IPC can be supplied directly via the signal distribution board using the enclosed power plug (see Fig. [B] below).

NOTE

Positioning on the signal distribution board

The dimensions and distances for placement and other details can be found in the Design Guide and the documentation for the individual components.

The figure below shows the connection of a C6015 IPC to an EJ system as an example. The components shown are schematic, to illustrate the functionality.

Fig. 36: Example for the connection of a C6015 IPC to an EJ system

4.8 Disassembly of the signal distribution board

⚠ **WARNING**

Risk of injury through electric shock and damage to the device!

Bring the module system into a safe, de-energized state before starting installation, disassembly or wiring of the modules.

NOTE

Risk of damage to components through electrostatic discharge!

Observe the regulations for ESD protection.

Each module is secured through latching on the distribution board, which has to be released for disassembly.

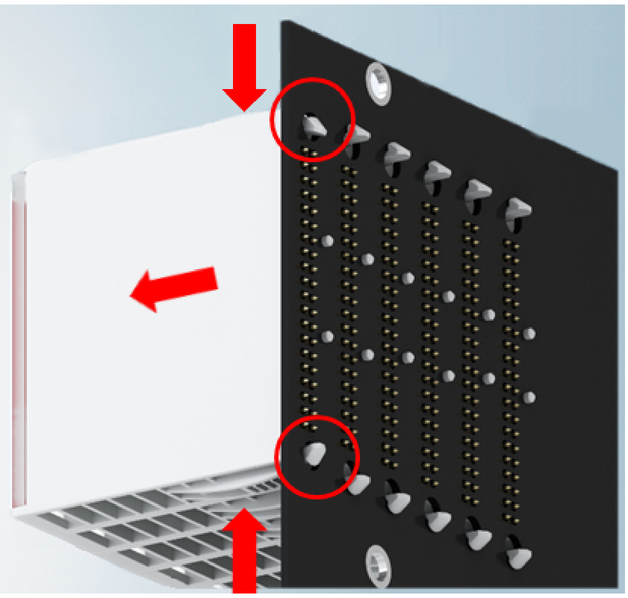


Fig. 37: Disassembly of EJ modules

To disassemble the module from the signal distribution board proceed as follows:

1. Before disassembly, ensure that the signal distribution board is securely connected to the mounting surface. Disassembly of an unsecured signal distribution board may result in damage to the board.
2. Press the upper and lower mounting tabs simultaneously and pull the module from board while gently moving it up and down.

4.9 Disposal

Products marked with a crossed-out wheeled bin shall not be discarded with the normal waste stream. The device is considered as waste electrical and electronic equipment. The national regulations for the disposal of waste electrical and electronic equipment must be observed.

5 EtherCAT basics

Please refer to the EtherCAT System Documentation for the EtherCAT fieldbus basics.

6 General notes for setting the watchdog

The ELxxxx terminals are equipped with a safety device (watchdog) which, e.g. in the event of interrupted process data traffic, switches the outputs (if present) to a presettable state after a presettable time, depending on the device and setting, e.g. to FALSE (off) or an output value.

The EtherCAT slave controller (ESC) features two watchdogs:

- SM watchdog (default: 100 ms)
- PDI watchdog (default: 100 ms)

Their times are individually parameterized in TwinCAT as follows:

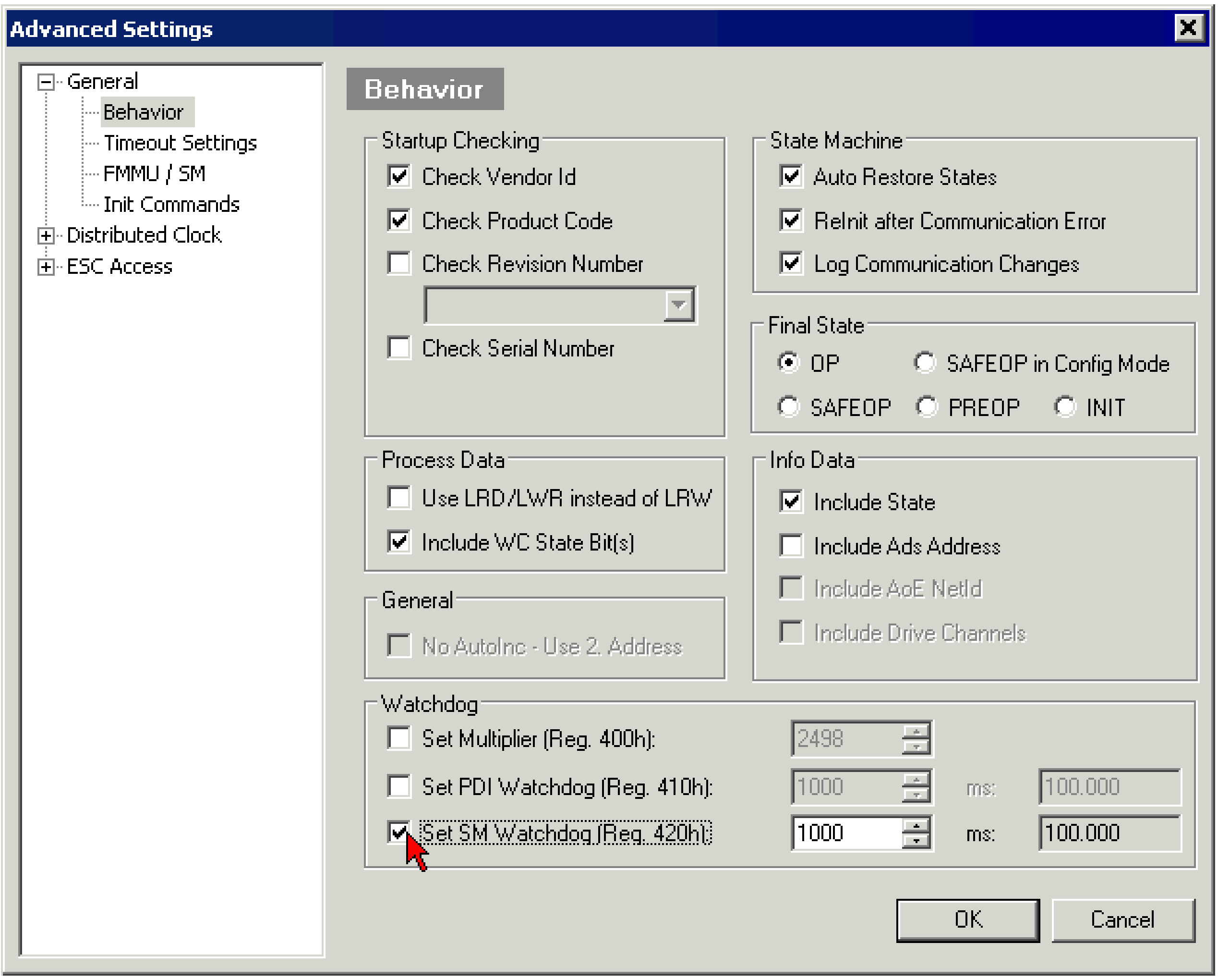


Fig. 38: eEtherCAT tab -> Advanced Settings -> Behavior -> Watchdog

Notes:

- the Multiplier Register 400h (hexadecimal, i.e. x0400) is valid for both watchdogs.
- each watchdog has its own timer setting 410h or 420h, which together with the Multiplier results in a resulting time.
- important: the Multiplier/Timer setting is only loaded into the slave at EtherCAT startup if the checkbox in front of it is activated.
- if it is not checked, nothing is downloaded and the setting located in the ESC remains unchanged.
- the downloaded values can be seen in the ESC registers x0400/0410/0420: ESC Access -> Memory

SM watchdog (SyncManager Watchdog)

The SyncManager watchdog is reset with each successful EtherCAT process data communication with the terminal. If, for example, no EtherCAT process data communication with the terminal takes place for longer than the set and activated SM watchdog time due to a line interruption, the watchdog is triggered. The status of the terminal (usually OP) remains unaffected. The watchdog is only reset again by a successful EtherCAT process data access.

The SyncManager watchdog is therefore a monitoring for correct and timely process data communication with the ESC from the EtherCAT side.

The maximum possible watchdog time depends on the device. For example, for "simple" EtherCAT slaves (without firmware) with watchdog execution in the ESC it is usually up to ~170 seconds. For "complex" EtherCAT slaves (with firmware) the SM watchdog function is usually parameterized via Reg. 400/420 but executed by the µC and can be significantly lower. In addition, the execution may then be subject to a certain time uncertainty. Since the TwinCAT dialog may allow inputs up to 65535, a test of the desired watchdog time is recommended.

PDI watchdog (Process Data Watchdog)

If there is no PDI communication with the EtherCAT slave controller (ESC) for longer than the set and activated PDI watchdog time, this watchdog is triggered.

PDI (Process Data Interface) is the internal interface of the ESC, e.g. to local processors in the EtherCAT slave. With the PDI watchdog this communication can be monitored for failure.

The PDI watchdog is therefore a monitoring for correct and timely process data communication with the ESC, but viewed from the application side.

Calculation

Watchdog time = [1/25 MHz * (Watchdog multiplier + 2)] * PDI/SM watchdog

Example: default setting Multiplier=2498, SM watchdog=1000 -> 100 ms

The value in Multiplier + 2 corresponds to the number of 40ns base ticks representing one watchdog tick.

⚠ CAUTION

Undefined state possible!

The function for switching off of the SM watchdog via SM watchdog = 0 is only implemented in terminals from version -0016. In previous versions this operating mode should not be used.

⚠ CAUTION

Damage of devices and undefined state possible!

If the SM watchdog is activated and a value of 0 is entered the watchdog switches off completely. This is the deactivation of the watchdog! Set outputs are NOT set in a safe state if the communication is interrupted.

7 Commissioning

7.1 Adjustable parameters

Duty Cycle

The duty cycle can be set individually for each channel to a value between 0 and 100 %. The value is set per channel via the output process data PWM Outputs Channel n → PWM Output. The maximum possible value in the process data is 32767.

- Duty Cycle 0 % = 0
- Duty Cycle 50 % = 16384
- Duty Cycle 100 % = 32767

The current value for the output can be read out for diagnostic purposes in the CoE object 0x60n0:13 [▶ 61] Output duty cycle. Alternatively, the objects 0x1A01, 0x1A03, 0x1A05 and 0x1A07 can also be mapped in the process data. Here the current process data value can then be read cyclically via the Output duty cycle variable.

Master Gain

The Master Gain sets the total brightness across all channels and is therefore only adjustable once for all four channels to dim all channels by the same factor. This value must be set in the process data. For this purpose the object *0x1604 PWM Outputs Device* must be mapped in the process data in *Sync Manager 2 Outputs* or alternatively the Predefined PDO Assignment *4 Ch + Master gain* must be selected. In addition, the value for the Master Gain can be permanently specified via the CoE object 0xF819:12 [▶ 60] as long as the object is not mapped in the process data. The brightness can be set between 0 and 100%.

If the value is mapped into the process data, it is initially 0 by default. This value must be changed to the desired Master Gain value > 0 in order to see a light at the output of the EJ2564 EtherCAT plug-in module. The maximum possible value is 32767. The actual output value of the duty cycle is then calculated via the value entered in the process data for the duty cycle multiplied by the factor (Master Gain/32767).

If Master Gain and the gamma value are used together to scale the output, the output value is always calculated first with the Master Gain and then with the gamma.

PWM frequency

The frequency of the PWM output for all channels can be changed in CoE object 0xF819:11 [▶ 60] *PWM frequency*. The value is specified in Hz. The default value is 5000 Hz. The frequency can be set between 1 and 16000 Hz via this object.

Depending on the set frequency, the number of steps changes from the lowest to the highest brightness. With increasing PWM frequency the resolution decreases. The resolution or step size is displayed in the CoE object 0xF919:11 [▶ 62] *Resolution*.

In the delivery state with a PWM frequency of the brightness is changed approx. every five PDO increments. (32767/6399 = 5.12)

At the low frequencies, where a visible flashing is generated, it must be considered that the individual pulses have the maximum brightness and cannot be dimmed, since dimming is usually realized at fast frequencies via the duty cycle. This means that at low frequencies no arbitrary color mixing is possible. If flashing with any color mixture is desired, it must be implemented in the PLC program.

Ramp Time

The Ramp Time specifies the time for dimming up/down from the lowest to the maximum brightness. The time can be set for each channel individually in the CoE object 0x80n0:25 [▶ 60] *Ramp time*. For values smaller than 0.05 s the ramp is inactive and the output value is controlled synchronously with the cycle. The maximum time for the ramp time is 10000 s.

Gamma - scaling of the output

The gamma value can be used to adjust the scaling of the output. The brightness behavior can, for example, be approximated to the perception of the human eye. Gamma can be set per channel in the CoE object 0x80n00:24 [▶ 60] *Gamma*. If this value is set to 1.0, the scaling is linear. Some progressions of the output at different gamma values can be taken from the following figure.

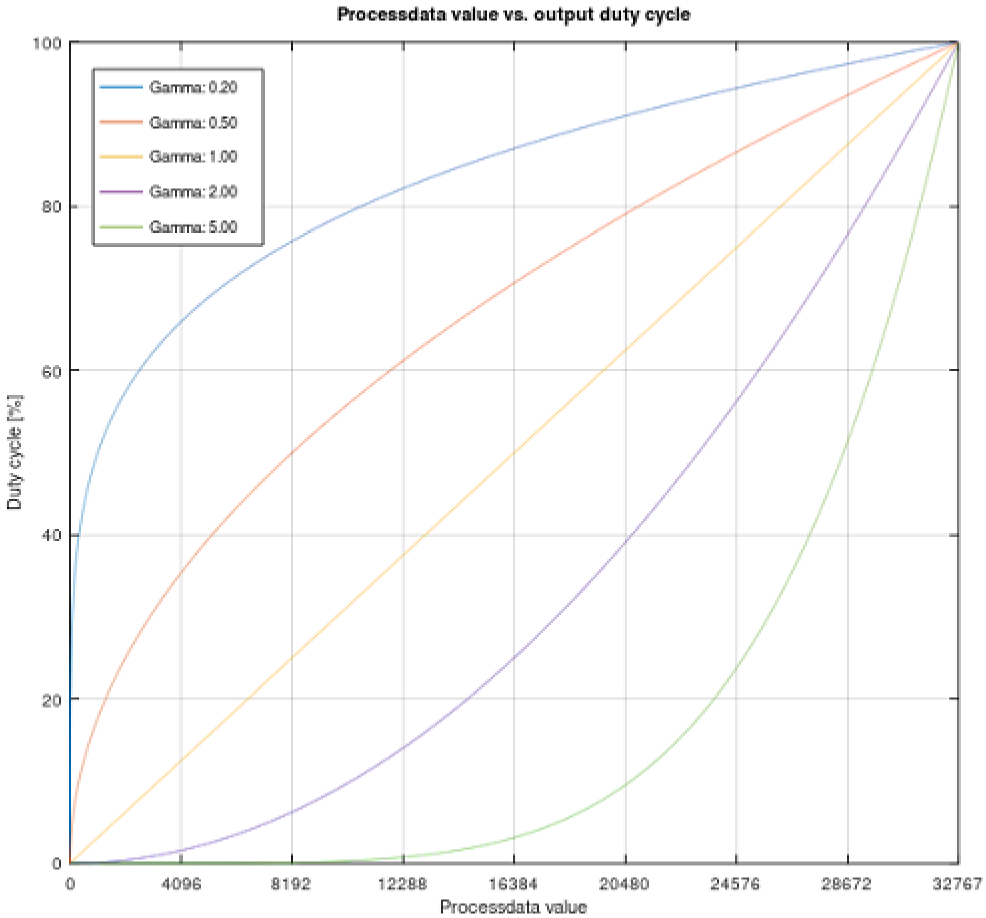


Fig. 39: Scaling of the output via the gamma value

The output is calculated using the following relationship: (Current_process_data_value/ Maximum_process_data_value)^Gamma

The following is an example calculation for the process data value 4096 and Gamma 0.5:

(4096/32767)^0.5 = 0.35

The result is therefore 35%. The actual output value of the duty cycle corresponds to the specified value from the process data when gamma is equal to 1.0.

If Master Gain and the gamma value are used together to scale the output, the output value is always calculated first with the Master Gain and then with the gamma.

Example application for the linearization of brightness for the human eye:
The brightness perceived by humans increases more steeply in dark areas and less steeply in bright areas. The human eye is assigned a gamma value of approx. 0.3 to 0.5. If you want to perceive the brightness signal of a display device (e.g. a monitor or an illumination device) linearly, you have to pre-distort it with the reciprocal of the gamma value (approx. 1/0.3 = 3.3 to 1/0.5 = 2) so that the two non-linearities cancel each other out and the course appears linear to the observer. For the gamma values of the EJ2564 EtherCAT plug-in module a value between 2 and 3.3 would have to be specified for the human eye.

Behavior in watchdog case

The behavior in the watchdog case can be specified for each channel in the CoE object 0x80n0:05 [▸ 60] *Watchdog*. It can be selected whether the output keeps the last process data value in the watchdog case or whether it assumes a value specified in CoE object 0x80n0:0D [▸ 60] *Default output*. This can be used for example for error indication, so that in the watchdog case all channels are switched off except the channel with the red LED to signal the error.

With the default output value from 0x80n0:0D [▸ 60] it must be noted that this is the default duty cycle. 32767 therefore corresponds to the maximum brightness.

In the watchdog case, the values entered in the CoE for the frequency, gamma and master gain are retained. Specified ramp times have no influence.

NOTE

Damage to the equipment and unintended behavior of the system is possible!

Observe the general notes on the watchdog setting [▸ 50].

7.2 Process data

7.2.1 Process data overview

Manual process data assignment is necessary for TwinCAT up to version 2.10.

Sync Manager (SM)

The scope of the process data offered can be changed via the "Process data" tab (see Fig. *Process Data SM2, EJ2564 tab)*.
The PDOs from the range 0x160n (0x1600 to 0x1604) can be assigned to the Output SyncManager 2, see Fig. *Process Data SM2, EJ2564 tab)*.

The PDOs from the range 0x1A0n (0x1A00 to 0x1A07) to the Input Sync Manager 3. See fig. *Process Data SM3, EJ2564 tab)*.

Not all combinations are technically useful.

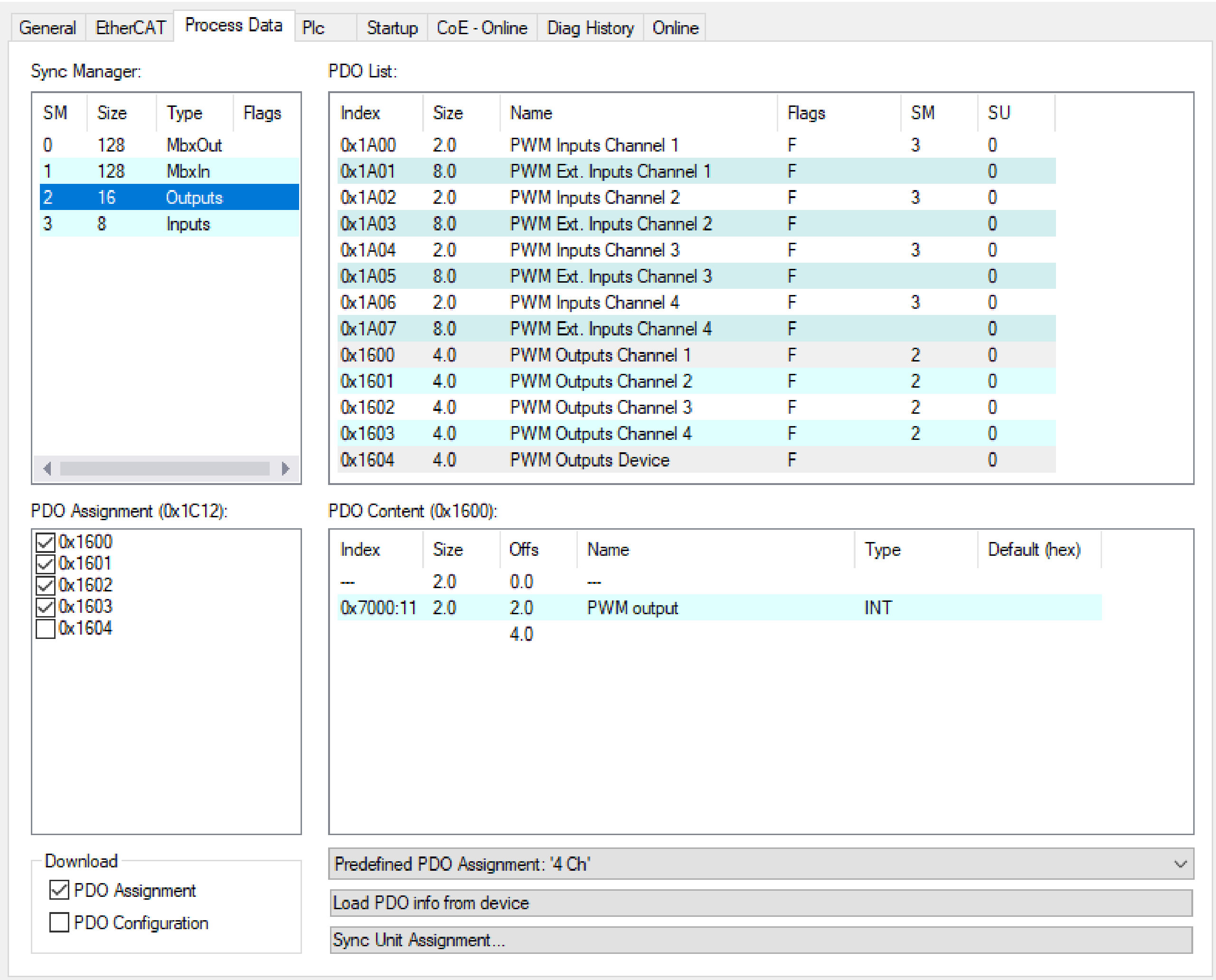


Fig. 40: Process Data SM2, EJ2564 tab

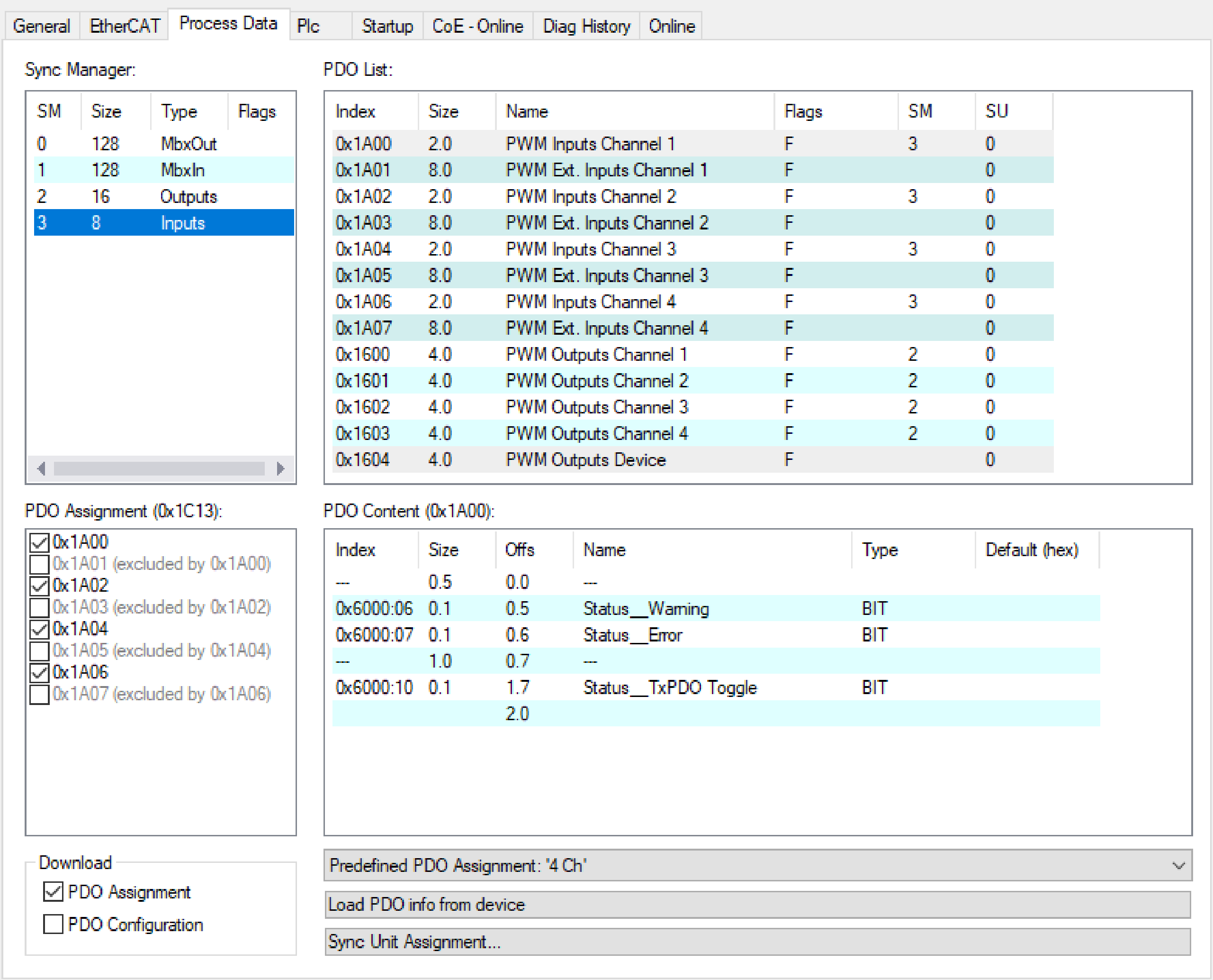


Fig. 41: Process Data SM3, EJ2564 tab

Manual PDO Assignment

1. To configure the process data, select the required Sync Manager (SM2 and SM3 can be edited here) in the "Sync Manager" box at the top left (see fig. *Process Data SM3, EJ2564 tab*).
2. The process data assigned to this Sync Manager can then be switched on or off in the “PDO Assignment” box underneath.
3. Restarting the EtherCAT system, or reloading the configuration in Config mode (F4), causes the EtherCAT communication to restart, and the process data is transferred from the module.

SM2, PDO Assignment 0x1C12				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content Index - Name**
0x1600	-	4.0	PWM Outputs Channel 1	0x7000:11 - PWM Output
0x1601	-	4.0	PWM Outputs Channel 2	0x7010:11 - PWM Output
0x1602	-	4.0	PWM Outputs Channel 3	0x7020:11 - PWM Output
0x1603	-	4.0	PWM Outputs Channel 4	0x7030:11 - PWM Output
0x1604	-	4.0	PWM Outputs Device	0xF719:11 - Master gain

SM3, PDO Assignment 0x1C13				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content Index - Name**
0x1A00	0x1A01	2.0	PWM Inputs Channel 1	0x6000:06 - Warning 0x6000:07 - Error 0x6000:10 - TxPDO Toggle
0x1A01	0x1A00	8.0	PWM Ext. Inputs Channel 1	0x6000:06 - Warning 0x6000:07 - Error 0x6000:10 - TxPDO Toggle 0x6000:13 - Output duty cycle
0x1A02	0x1A03	2.0	PWM Inputs Channel 2	0x6010:06 - Warning 0x6010:07 - Error 0x6010:10 - TxPDO Toggle
0x1A03	0x1A02	8.0	PWM Ext. Inputs Channel 2	0x6010:06 - Warning 0x6010:07 - Error 0x6010:10 - TxPDO Toggle 0x6010:13 - Output duty cycle
0x1A04	0x1A05	2.0	PWM Inputs Channel 3	0x6020:06 - Warning 0x6020:07 - Error 0x6020:10 - TxPDO Toggle
0x1A05	0x1A04	8.0	PWM Ext. Inputs Channel 3	0x6020:06 - Warning 0x6020:07 - Error 0x6020:10 - TxPDO Toggle 0x6020:13 - Output duty cycle
0x1A06	0x1A07	2.0	PWM Inputs Channel 4	0x6030:06 - Warning 0x6030:07 - Error 0x6030:10 - TxPDO Toggle
0x1A07	0x1A06	8.0	PWM Ext. Inputs Channel 4	0x6030:06 - Warning 0x6030:07 - Error 0x6030:10 - TxPDO Toggle 0x6030:13 - Output duty cycle

7.2.2 Preselection of process data

An EtherCAT device offers several different process data objects (PDO) for input and output data, which can be configured in the System Manager, i.e. they can be activated or deactivated for cyclic transmission. See further below for the corresponding overview. Attention is thereby to be paid to the compatibility of input and output PDO.

From TwinCAT 2.11 with the EtherCAT devices intended for the purpose according to the ESI/XML description, the process data for input and output can be activated simultaneously by appropriate predefined sentences, "Predefined PDO".

The EtherCAT plug-in module has in the "Process data" tab

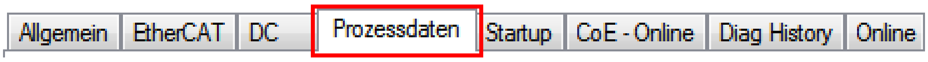


Fig. 42: "Process data" tab

the following "Predefined PDO" sentences:

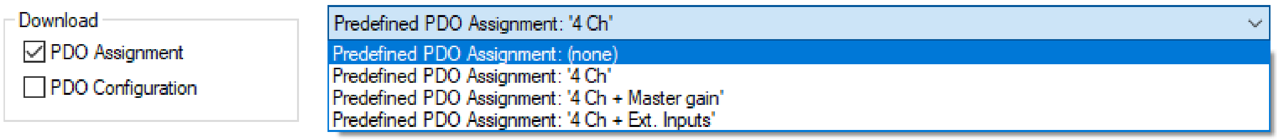


Fig. 43: TwinCAT System Manager with the PDO selection

Composition of the "Predefined PDO Assignments":

Name	SM2, PDO assignment	SM3, PDO assignment
4 Ch	0x1600 0x1601 0x1602 0x1603	0x1A00 0x1A02 0x1A04 0x1A06
4 Ch + Master gain	0x1600 0x1601 0x1602 0x1603 0x1604	0x1A00 0x1A02 0x1A04 0x1A06
4 Ch + Ext. Inputs	0x1600 0x1601 0x1602 0x1603	0x1A01 0x1A03 0x1A05 0x1A07

7.3 EJ2564 - object description and parameterization

EtherCAT XML Device Description

The display matches that of the CoE objects from the EtherCAT XML Device Description. We recommend downloading the latest XML file from the download area of the Beckhoff website and installing it according to installation instructions.

Parameterization via the CoE list (CAN over EtherCAT)

The EtherCAT device is parameterized via the CoE - Online tab (with a double click on the respective object) or via the Process Data tab (assignment of PDOs). A detailed description can be found in the EtherCAT System-Documentation in chapter “EtherCAT subscriber configuration”
Please note the general CoE notes in the EtherCAT System Documentation in chapter “CoE-interface” when using/manipulating the CoE parameters:
- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use "CoE reload" for resetting changes

Introduction

The CoE overview contains objects for different intended applications:

- Objects required for parameterization during commissioning:
  - Restore [▸ 59] Object Index 0x1011
  - Configuration data [▸ 60] Index 0x80n0, 0xF819
- Objects intended for regular operation, e.g. through ADS access.
- Profile-specific objects:
  - Input data [▸ 61] Index 0x60n0
  - Output data [▸ 61] Index 0x70n0, 0xF719
  - Information and diagnostic data [▸ 62] Index 0xF919, 0xFA19, 0xF000, 0xF008
- Standard objects [▸ 63]

The following section first describes the objects required for normal operation, followed by a complete overview of missing objects.

7.3.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore default parameters	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to **“0x64616F6C”** in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0_{dec})

7.3.2 Configuration data

Index 80n0 PWM Settings Ch. n+1 (for Ch. 1 – 4 (0 < n < 3))

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	PWM Settings Ch. n+1	Max. Subindex	UINT8	RO	0x25 (37$_{dec}$)
80n0:05	Watchdog	0: Default watchdog value • The output takes the specified value from 0x80n0:0D after an EtherCAT watchdog. 2: Last output value active • With an EtherCAT watchdog the last process data value is retained.	UINT8	RW	0x00 (0$_{dec}$)
80n0:0D	Default output	After an EtherCAT watchdog the output jumps to this value if a 0 is entered in 0x80n0:05. Permitted values: 0 … 32767 (0 … 100 %)	INT16	RW	0x0000 (0$_{dec}$)
80n0:24	Gamma	Scaling of the output • 1.0: linear scaling	REAL32	RW	1.0
80n0:25	Ramp time	Time for dimming up/down from lowest to maximum brightness in seconds. For values smaller than 0.05 s the ramp is inactive and the output value is influenced cycle-synchronous. Permitted values: 0 … 10000 s	REAL32	RW	0.0

Index F819 PWM Settings

Index (hex)	Name	Meaning	Data type	Flags	Default
F819:0	PWM settings	Max. Subindex	UINT8	RO	0x12 (18$_{dec}$)
F819:11	PWM frequency	PWM frequency of the output signal in Hz	INT32	RW	0x00001388 (5000$_{dec}$)
F819:12	Master gain	Value of the total brightness across all channels (if in the process data 0x1604 is **not** mapped)	INT16	RW	0x7FFF (32767$_{dec}$)

7.3.3 Command object

Index FB00 PWM Command

Index (hex)	Name	Meaning	Data type	Flags	Default
FB00:0	PWM Command	Module specific commands can be executed via PWM Command.	UINT8	RO	0x03 (3$_{dec}$)
FB00:01	Request		OCTET-STRING[2]	RW	{0}
FB00:02	Status		UINT8	RO	0x00 (0$_{dec}$)
FB00:03	Response		OCTET-STRING[4]	RO	{0}

7.3.4 Input data

Index 60n0 PWM Inputs Ch. n+1 (for Ch. 1 - 4 (0 ≤ n ≤ 3))

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	PWM Inputs Ch. n+1	Max. Subindex	UINT8	RO	0x13 (19_{dec})
60n0:06	Warning	A warning has occurred • Internal temperature of the module > 80 °C • No LED supply voltage	BOOLEAN	RO	0x00 (0_{dec})
60n0:07	Error	An error has occurred Internal temperature of the module > 100 °C • No LED supply voltage and min. one channel switched on	BOOLEAN	RO	0x00 (0_{dec})
60n0:10	TxPDO Toggle	Toggle bit	BOOLEAN	RO	0x00 (0_{dec})
60n0:13	Output duty cycle	Current duty cycle of the output 0 = 0 % Duty cycle 32767 (0x7FFF) = 100 % Duty cycle	INT16	RO	0x0000 (0_{dec})

7.3.5 Output data

Index 70n0 PWM Outputs Ch. n+1 (for Ch. 1 - 4 (0 ≤ n ≤ 3))

Index (hex)	Name	Meaning	Data type	Flags	Default
70n0:0	PWM Outputs Ch. n+1	Max. Subindex	UINT8	RO	0x11 (17_{dec})
70n00:11	PWM Output	Output value for the brightness: 0 = off 0x7FFF = Full brightness	INT16	RO	0x0000 (0_{dec})

Index F719 PWM Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
F719:0	PWM Outputs	Max. Subindex	UINT8	RO	0x11 (17_{dec})
F719:11	Master gain	Current value of the total brightness across all channels, if 0x1604 is mapped in the process data	INT16	RO	0x0000 (0_{dec})

7.3.6 Information / diagnostic data

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index distance of the objects of the individual channels	UINT16	RO	0x0010 (16_{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0004 (4_{dec})

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0_{dec})

Index F081 Download revision

Index (hex)	Name	Meaning	Data type	Flags	Default
F081:0	Download revision	Max. Subindex	UINT8	RO	0x01 (1_{dec})
F081:01	Revision number	The subindex 0xF081:01 (Download revision) describes the revision level of the module.	UINT32	RW	0x00000000 (0_{dec})

Index F919 PWM Info data

Index (hex)	Name	Meaning	Data type	Flags	Default
F919:0	PWM Info data	Max. Subindex	UINT8	RO	0x11 (17_{dec})
F919:11	Resolution	Number of steps from lowest to highest brightness. The resolution decreases with increasing PWM frequency (0xF800:11). The brightness is changed approx. every five PDO increments in case of default value. (32767/6399 = 5.12)	INT16	RO	0x0000 (0_{dec})

Index FA19 PWM Diag data

Index (hex)	Name	Meaning	Data type	Flags	Default
FA19:0	PWM Diag data	Max. Subindex	UINT8	RO	0x11 (17_{dec})
FA19:11	PCB temperature	Circuit board temperature in 0.1 °C	INT16	RW	0x0000 (0_{dec})

7.3.7 Standard objects (0x1000-0x1FFF)

The standard objects have the same meaning for all EtherCAT slaves.

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the used CoE profile (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x00FA1389 (16389001_{dec})

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EJ2564

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	00

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	01

Index 100B Bootloader version

Index (hex)	Name	Meaning	Data type	Flags	Default
100B:0	Bootloader version	Bootloader version	STRING	RO	

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x0E6C3052 (241971282_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the Low Word (bit 0-15) indicates the special terminal number, the High Word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the Low Byte (bit 0-7) of the Low Word contains the year of production, the High Byte (bit 8-15) of the Low Word contains the week of production, the High Word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default
10F0:0	Backup parameter handling	Information for standardized loading and saving of backup entries	UINT8	RO	0x01 (1_{dec})
10F0:01	Checksum	Checksum across all backup entries of the EtherCAT slave	UINT32	RO	0x00000000 (0_{dec})

Index 10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	Max. Subindex	UINT8	RO	0x15 (21_{dec})
10F3:01	Maximum Messages	Maximum number of stored messages. A maximum of 16 messages can be stored	UINT32	RO	0x00000000 0_{dec})
10F3:02	Newest Message	Subindex of the latest message	UINT8	RO	0x00 0_{dec})
10F3:03	Newest Acknowledged Message	Subindex of the last confirmed message	UINT8	RO	0x00 (0_{dec})
10F3:04	New Message available	Indicates that a new message is available	BOOLEAN	RO	0x00 (0_{dec})
10F3:05	Flags	not used	UINT16	RO	0x0000 (0_{dec})
10F3:06	Diagnosis Message 001	Message 1	OCTET-STRING[20]	RO	{0}
…	…	…	…	…	…
10F3:015	Diagnosis Message 016	Message 16	OCTET-STRING[20]	RO	{0}

Index 1600 PWM RxPDO-Map Outputs Ch. 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	PWM RxPDO-Map Outputs Ch. 1	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (PWM Outputs Ch.1), entry 0x11 (PWM output))	UINT32	RO	0x7000:11, 16

Index 1601 PWM RxPDO-Map Outputs Ch. 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1601:0	PWM RxPDO-Map Outputs Ch. 2	PDO Mapping RxPDO 2	UINT8	RO	0x02 (2_{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1601:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (PWM Outputs Ch.2), entry 0x11 (PWM output))	UINT32	RO	0x7010:11, 16

Index 1602 PWM RxPDO-Map Outputs Ch. 3

Index (hex)	Name	Meaning	Data type	Flags	Default
1602:0	PWM RxPDO-Map Outputs Ch. 3	PDO Mapping RxPDO 3	UINT8	RO	0x02 (2_{dec})
1602:01	SubIndex 001	1. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1602:02	SubIndex 002	2. PDO Mapping entry (object 0x7020 (PWM Outputs Ch.3), entry 0x11 (PWM output))	UINT32	RO	0x7020:11, 16

Index 1603 PWM RxPDO-Map Outputs Ch. 4

Index (hex)	Name	Meaning	Data type	Flags	Default
1603:0	PWM RxPDO-Map Outputs Ch. 4	PDO Mapping RxPDO 4	UINT8	RO	0x02 (2_{dec})
1603:01	SubIndex 001	1. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1603:02	SubIndex 002	2. PDO Mapping entry (object 0x7030 (PWM Outputs Ch.4), entry 0x11 (PWM output))	UINT32	RO	0x7030:11, 16

Index 1604 PWM RxPDO-Map Outputs Device

Index (hex)	Name	Meaning	Data type	Flags	Default
1604:0	PWM RxPDO-Map Outputs Device	PDO Mapping RxPDO 5	UINT8	RO	0x02 (2_{dec})
1604:01	SubIndex 001	1. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1604:02	SubIndex 002	2. PDO Mapping entry (object 0xF719 (PWM Outputs), entry 0x11 (Master gain))	UINT32	RO	0xF719:11, 16

Index 1800 PWM TxPDO-Par Inputs Ch. 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1800:0	PWM TxPDO-Par Inputs Ch. 1	PWM TxPDO-Par Inputs Ch. 1	UINT8	RO	0x06 (6_{dec})
1800:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A01 (6657_{dec})

Index 1801 PWM TxPDO-Par Ext. Inputs Ch. 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1801:0	PWM TxPDO-Par Ext. Inputs Ch. 1	PWM TxPDO-Par Ext. Inputs Ch. 1	UINT8	RO	0x06 (6_{dec})
1801:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A00 (6656_{dec})

Index 1802 PWM TxPDO-Par Inputs Ch. 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1802:0	PWM TxPDO-Par Inputs Ch. 2	PWM TxPDO-Par Inputs Ch. 2	UINT8	RO	0x06 (6_{dec})
1802:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A03 (6659_{dec})

Index 1803 PWM TxPDO-Par Ext. Inputs Ch. 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1803:0	PWM TxPDO-Par Ext. Inputs Ch. 2	PWM TxPDO-Par Ext. Inputs Ch. 2	UINT8	RO	0x06 (6_{dec})
1803:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A02 (6658_{dec})

Index 1804 PWM TxPDO-Par Inputs Ch. 3

Index (hex)	Name	Meaning	Data type	Flags	Default
1804:0	PWM TxPDO-Par Inputs Ch. 3	PWM TxPDO-Par Inputs Ch. 3	UINT8	RO	0x06 (6_{dec})
1804:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A05 (6661_{dec})

Index 1805 PWM TxPDO-Par Ext. Inputs Ch. 3

Index (hex)	Name	Meaning	Data type	Flags	Default
1805:0	PWM TxPDO-Par Ext. Inputs Ch. 3	PWM TxPDO-Par Ext. Inputs Ch. 3	UINT8	RO	0x06 (6_{dec})
1805:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A04 (6660_{dec})

Index 1806 PWM TxPDO-Par Inputs Ch. 4

Index (hex)	Name	Meaning	Data type	Flags	Default
1806:0	PWM TxPDO-Par Inputs Ch. 4	PWM TxPDO-Par Inputs Ch. 4	UINT8	RO	0x06 (6_{dec})
1806:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A07 (6663_{dec})

Index 1807 PWM TxPDO-Par Ext. Inputs Ch. 4

Index (hex)	Name	Meaning	Data type	Flags	Default
1807:0	PWM TxPDO-Par Ext. Inputs Ch. 4	PWM TxPDO-Par Ext. Inputs Ch. 4	UINT8	RO	0x06 (6_{dec})
1807:06	Exclude TxPDOs	Exclude TxPDOs	UINT16	RO	0x1A06 (6662_{dec})

Index 1A00 PWM TxPDO-Map Inputs Ch. 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	PWM TxPDO-Map Inputs Ch. 1	PDO Mapping TxPDO 1	UINT8	RO	0x05 (5_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x06 (Warning))	UINT32	RO	0x6000:06, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x07 (Error))	UINT32	RO	0x6000:07, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6000:10, 1

Index 1A01 PWM TxPDO-Map Ext. Inputs Ch. 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	PWM TxPDO-Map Ext. Inputs Ch. 1	PDO Mapping TxPDO 1	UINT8	RO	0x07 (7_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x06 (Warning))	UINT32	RO	0x6000:06, 1
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x07 (Error))	UINT32	RO	0x6000:07, 1
1A01:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6000:10, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (32 bits align)	UINT32	RO	0x0000:00, 32
1A01:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.1), entry 0x13 (Output duty cycle))	UINT32	RO	0x6000:13, 16

Index 1A02 PWM TxPDO-Map Inputs Ch. 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	PWM TxPDO-Map Inputs Ch. 2	PDO Mapping TxPDO 2	UINT8	RO	0x05 (5_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (PWM Inputs Ch.2), entry 0x06 (Warning))	UINT32	RO	0x6010:06, 1
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (PWM Inputs Ch.2), entry 0x07 (Error))	UINT32	RO	0x6010:07, 1
1A02:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A02:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (PWM Inputs Ch.2), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6010:10, 1

Index 1A03 PWM TxPDO-Map Ext. Inputs Ch. 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A03:0	PWM TxPDO-Map Ext. Inputs Ch. 2	PDO Mapping TxPDO 1	UINT8	RO	0x07 (7_{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A03:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.2), entry 0x06 (Warning))	UINT32	RO	0x6010:06, 1
1A03:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.2), entry 0x07 (Error))	UINT32	RO	0x6010:07, 1
1A03:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A03:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.2), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6010:10, 1
1A03:06	SubIndex 006	6. PDO Mapping entry (32 bits align)	UINT32	RO	0x0000:00, 32
1A03:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.2), entry 0x13 (Output duty cycle))	UINT32	RO	0x6010:13, 16

Index 1A04 PWM TxPDO-Map Inputs Ch. 3

Index (hex)	Name	Meaning	Data type	Flags	Default
1A04:0	PWM TxPDO-Map Inputs Ch. 3	PDO Mapping TxPDO 3	UINT8	RO	0x05 (5_{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A04:02	SubIndex 002	2. PDO Mapping entry (object 0x6020 (PWM Inputs Ch.3), entry 0x06 (Warning))	UINT32	RO	0x6020:06, 1
1A04:03	SubIndex 003	3. PDO Mapping entry (object 0x6020 (PWM Inputs Ch.3), entry 0x07 (Error))	UINT32	RO	0x6020:07, 1
1A04:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A04:05	SubIndex 005	5. PDO Mapping entry (object 0x6020 (PWM Inputs Ch.3), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6020:10, 1

Index 1A05 PWM TxPDO-Map Ext. Inputs Ch. 3

Index (hex)	Name	Meaning	Data type	Flags	Default
1A05:0	PWM TxPDO-Map Ext. Inputs Ch. 3	PDO Mapping TxPDO 1	UINT8	RO	0x07 (7_{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A05:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.3), entry 0x06 (Warning))	UINT32	RO	0x6020:06, 1
1A05:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.3), entry 0x07 (Error))	UINT32	RO	0x6020:07, 1
1A05:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A05:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.3), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6020:10, 1
1A05:06	SubIndex 006	6. PDO Mapping entry (32 bits align)	UINT32	RO	0x0000:00, 32
1A05:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.3), entry 0x13 (Output duty cycle))	UINT32	RO	0x6020:13, 16

Index 1A06 PWM TxPDO-Map Inputs Ch. 4

Index (hex)	Name	Meaning	Data type	Flags	Default
1A06:0	PWM TxPDO-Map Inputs Ch. 4	PDO Mapping TxPDO 4	UINT8	RO	0x05 (5_{dec})
1A06:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A06:02	SubIndex 002	2. PDO Mapping entry (object 0x6030 (PWM Inputs Ch.4), entry 0x06 (Warning))	UINT32	RO	0x6030:06, 1
1A06:03	SubIndex 003	3. PDO Mapping entry (object 0x6030 (PWM Inputs Ch.4), entry 0x07 (Error))	UINT32	RO	0x6030:07, 1
1A06:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A06:05	SubIndex 005	5. PDO Mapping entry (object 0x6030 (PWM Inputs Ch.4), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6030:10, 1

Index 1A07 PWM TxPDO-Map Ext. Inputs Ch. 4

Index (hex)	Name	Meaning	Data type	Flags	Default
1A07:0	PWM TxPDO-Map Ext. Inputs Ch. 4	PDO Mapping TxPDO 1	UINT8	RO	0x07 (7_{dec})
1A07:01	SubIndex 001	1. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1A07:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.4), entry 0x06 (Warning))	UINT32	RO	0x6030:06, 1
1A07:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.4), entry 0x07 (Error))	UINT32	RO	0x6030:07, 1
1A07:04	SubIndex 004	4. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A07:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.4), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6030:10, 1
1A07:06	SubIndex 006	6. PDO Mapping entry (32 bits align)	UINT32	RO	0x0000:00, 32
1A07:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (PWM Inputs Ch.4), entry 0x13 (Output duty cycle))	UINT32	RO	0x6030:13, 16

Index 1C00 Sync manager type

Index (hex)	Name	Meaning	Data type	Flags	Default
1C00:0	Sync manager type	Using the Sync Managers	UINT8	RO	0x04 (4_{dec})
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1_{dec})
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2_{dec})
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3_{dec})
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4_{dec})

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x05 (5_{dec})
1C12:01	Subindex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632_{dec})
1C12:02	Subindex 002	2. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1601 (5633_{dec})
1C12:03	Subindex 003	3. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1602 (5634_{dec})
1C12:04	Subindex 004	4. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1603 (5635_{dec})
1C12:05	Subindex 005	5. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1604 (5636_{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x04 (4_{dec})
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	Subindex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A02 (6658_{dec})
1C13:03	Subindex 003	3. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A04 (6600_{dec})
1C13:04	Subindex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A06 (6662_{dec})

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: cycle time of the local timer • Synchronous with SM 2 Event: cycle time of the master • DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: Free Run is supported • Bit 1 = 1: Synchron with SM 2 Event is supported • Bit 2-3 = 01: DC-Mode is supported • Bit 4-5 = 10: Output Shift with SYNC1 event (only DC mode) • Bit 14 = 1: dynamic times (measurement through writing of 1C32:08)	UINT16	RO	0x0003 (3_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x000186A0 (100000_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time	Minimum time between SYNC1 event and output of the outputs (in ns) 0, since EJ2564 does not support DC mode	UINT32	RO	0x00000000 (0_{dec})
1C32:08	Get Cycle Time	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:09, 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32$_{dec}$)
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 event (outputs available)	UINT16	RW	0x0022 (34$_{dec}$)
1C33:02	Cycle time	Cycle time (in ns): • Free Run: cycle time of the local timer • Synchronous with SM 2 Event: cycle time of the master DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000$_{dec}$)
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0$_{dec}$)
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: Free Run is supported • Bit 1: Synchron with SM 2 Event is supported (outputs available) • Bit 1: Synchron with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC-Mode is supported • Bit 4-5 = 01: Input Shift through local event (outputs available) • Bit 4-5 = 10: Input Shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 or 0x1C33:08)	UINT16	RO	0x0003 (3$_{dec}$)
1C33:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x000186A0 (100000$_{dec}$)
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0$_{dec}$)
1C33:07	Minimum delay time	Min. time between SYNC1 event and the reading of the inputs (in ns, DC mode only) 0, since EJ2564 does not support DC mode	UINT32	RO	0x00000000 (0$_{dec}$)
1C33:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03, 0x1C32:05 0x1C32:06, 0x1C32:09 0x1C33:03, 0x1C33:06, 0x1C33:09 are up-dated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0$_{dec}$)
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0$_{dec}$)
1C33:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0$_{dec}$)
1C33:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OP-ERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0$_{dec}$)
1C33:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0$_{dec}$)
1C33:20	Sync error	The synchronization was not correct in the last cycle (out-puts were output too late; DC mode only)	BOOLEAN	RO	0x00 (0$_{dec}$)

8 Appendix

8.1 Support and Service

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

Beckhoff's branch offices and representatives

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

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

You will also find further documentation for Beckhoff components there.

Beckhoff Support

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

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

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

Beckhoff Service

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

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

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

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

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

More Information:
www.beckhoff.com/EJ2564

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