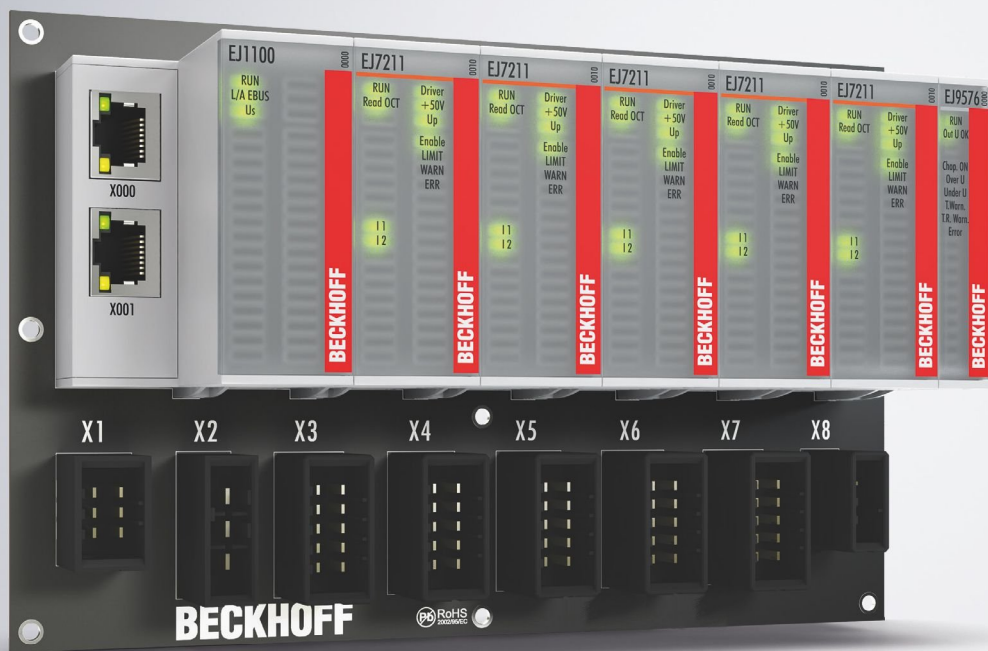


Documentation | EN

## EJ6002

2-channel serial interface RS232, RS422 or RS485





# Table of contents

|   |           |
|---|-----------|
| <b>1 Foreword</b> .....   | <b>5</b>  |
| 1.1 Notes on the documentation .....  | 5         |
| 1.2 Safety instructions .....   | 6         |
| 1.3 Intended use .....  | 7         |
| 1.4 Signal distribution board.....  | 7         |
| 1.5 Documentation issue status .....  | 7         |
| 1.6 Guide through documentation .....   | 8         |
| 1.7 Marking of EtherCAT plug-in modules .....   | 9         |
| 1.7.1 Beckhoff Identification Code (BIC).....   | 11        |
| 1.7.2 Electronic access to the BIC (eBIC).....  | 13        |
| 1.7.3 Certificates .....  | 15        |
| <b>2 System overview</b> .....  | <b>16</b> |
| <b>3 EJ6002 - Product description</b> .....   | <b>17</b> |
| 3.1 Introduction .....  | 17        |
| 3.2 Technical Data .....  | 18        |
| 3.3 Pinout RS232 .....  | 19        |
| 3.4 Pinout RS422 .....  | 20        |
| 3.5 Pinout RS485 .....  | 21        |
| 3.6 LEDs .....  | 22        |
| <b>4 Installation of EJ modules</b> .....   | <b>23</b> |
| 4.1 Power supply for the EtherCAT plug-in modules .....   | 23        |
| 4.2 Note on load voltage supply .....   | 24        |
| 4.3 EJxxxx - dimensions .....   | 25        |
| 4.4 Installation positions and minimum distances .....  | 26        |
| 4.4.1 Minimum distances for ensuring installability .....   | 26        |
| 4.4.2 Installation positions .....  | 27        |
| 4.5 Codings .....   | 29        |
| 4.5.1 Color coding .....  | 29        |
| 4.5.2 Mechanical position coding .....  | 30        |
| 4.6 Installation on the signal distribution board .....   | 31        |
| 4.7 Extension options .....   | 32        |
| 4.7.1 Using placeholder modules for unused slots .....  | 32        |
| 4.7.2 Linking with EtherCAT Terminals and EtherCAT Box modules via an Ethernet/EtherCAT connection..... | 33        |
| 4.8 IPC integration .....   | 34        |
| 4.9 Disassembly of the signal distribution board .....  | 36        |
| 4.10 Disposal .....   | 36        |
| <b>5 EtherCAT basics</b> .....  | <b>37</b> |
| <b>6 Commissioning</b> .....  | <b>38</b> |
| 6.1 Reference to documentation EL600x.....  | 38        |
| 6.2 EJ6002 - object description and parameterization .....  | 38        |
| 6.2.1 Restore object.....   | 39        |
| 6.2.2 Configuration data.....   | 39        |

|          |  |           |
|----------|--|-----------|
| 6.2.3    | Input data .....                       | 42        |
| 6.2.4    | Output data .....                      | 43        |
| 6.2.5    | Information and diagnostic data .....  | 44        |
| 6.2.6    | Standard objects (0x1000-0x1FFF) ..... | 44        |
| 6.2.7    | Command object (0xFB00) .....          | 56        |
| <b>7</b> | <b>Appendix .....</b>                  | <b>57</b> |
| 7.1      | Support and Service .....              | 57        |

# 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.

The qualified personnel is obliged to always use the currently valid documentation.

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.



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

### Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

### Third-party brands

Trademarks of third parties may be used in this documentation. You can find the trademark notices here: <https://www.beckhoff.com/trademarks>

## 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.

### Signal words

The signal words used in the documentation are classified below. In order to prevent injury and damage to persons and property, read and follow the safety and warning notices.

#### Personal injury warnings

**⚠ DANGER**

Hazard with high risk of death or serious injury.

**⚠ WARNING**

Hazard with medium risk of death or serious injury.

**⚠ CAUTION**

There is a low-risk hazard that could result in medium or minor injury.

#### Warning of damage to property or environment

**NOTICE**

The environment, equipment, or data may be damaged.

#### Information on handling the product



This information includes, for example:  
recommendations for action, assistance or further information on the product.

## 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

### NOTICE

#### 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   |
|---------|---|
| 1.6.0   | <ul style="list-style-type: none"> <li>• Update chapter <i>EJ6002 - Object description and parameterization</i></li> <li>• Update structure</li> </ul>  |
| 1.5     | <ul style="list-style-type: none"> <li>• Update chapter <i>Marking of EtherCAT plug-in modules</i></li> <li>• Update Technical Data</li> <li>• Update chapter <i>Power supply for the EtherCAT plug-in modules</i></li> <li>• Update chapter <i>EJ6002 - Object description and parameterization</i></li> </ul>   |
| 1.4     | <ul style="list-style-type: none"> <li>• Update chapter <i>Marking of EtherCAT plug-in modules</i></li> <li>• Update Technical Data</li> <li>• Chapter <i>Disposal</i> added</li> <li>• Update structure</li> </ul>   |
| 1.3     | <ul style="list-style-type: none"> <li>• Update chapter <i>Configuration data</i></li> </ul>  |
| 1.2     | <ul style="list-style-type: none"> <li>• New title page</li> <li>• Update Technical data</li> <li>• Update chapter <i>EJ6002 - Connection</i></li> <li>• Chapters <i>Basics communication</i>, <i>TwinCAT Quick Start</i>, <i>TwinCAT development environment</i> and <i>General Notes - EtherCAT Slave Application</i> replaced by references in the chapter <i>Guide through the documentation</i></li> <li>• Chapter <i>EJ6002 - Object description and parameterization</i> added</li> <li>• Update revision status</li> <li>• Structural update</li> </ul> |
| 1.1     | <ul style="list-style-type: none"> <li>• Note <i>Signal Distribution Board</i> added</li> <li>• Chapter <i>Version identification of EtherCAT devices</i> replaced by <i>Marking of EtherCAT plug-in modules</i></li> <li>• Update Technical data</li> <li>• Update chapter <i>Connection</i></li> </ul>  |
| 1.0     | <ul style="list-style-type: none"> <li>• First publication EJ6002</li> </ul>  |

## 1.6 Guide through documentation

### NOTICE



#### Further components of documentation

This documentation describes device-specific content. It is part of the modular documentation concept for Beckhoff I/O components. For the use and safe operation of the device / devices described in this documentation, additional cross-product descriptions are required, which can be found in the following table.

| Title  | Description  |
|--|--|
| <b>EtherCAT System Documentation</b> ( <a href="#">PDF</a> )   | <ul style="list-style-type: none"> <li>• System overview</li> <li>• EtherCAT basics</li> <li>• Cable redundancy</li> <li>• Hot Connect</li> <li>• EtherCAT devices configuration</li> </ul>  |
| <b>Design Guide EJ8xxx - Signal distribution board for standard EtherCAT plug-in modules</b> ( <a href="#">PDF</a> )                                 | <p>Notes on the design of a signal distribution board for standard EtherCAT plug-in modules.</p> <ul style="list-style-type: none"> <li>• Requirements for the signal distribution board</li> <li>• Backplane mounting guidelines</li> <li>• Module placement</li> <li>• Routing guidelines</li> </ul> |
| <b>Documentation of the corresponding ELxxxx EtherCAT Terminal</b><br>(s. <a href="#">note on documentation of ELxxxx</a> ) [ <a href="#">▶ 38</a> ] | <ul style="list-style-type: none"> <li>• Notes on the principle of operation and</li> <li>• descriptions for configuration and parameterization are transferable to the corresponding EtherCAT plug-in modules</li> </ul>  |
| <b>Infrastructure for EtherCAT/Ethernet</b> ( <a href="#">PDF</a> )  | Technical recommendations and notes for design, implementation and testing   |
| <b>Software Declarations I/O</b> ( <a href="#">PDF</a> )   | Open source software declarations for Beckhoff I/O components  |

The documentations can be viewed at and downloaded from the Beckhoff website ([www.beckhoff.com](http://www.beckhoff.com)) via:

- the “Documentation and Download” area of the respective product page,
- the [Download finder](#),
- the [Beckhoff Information System](#).

If you have any suggestions or proposals for our documentation, please send us an e-mail stating the documentation title and version number to: [documentation@beckhoff.com](mailto:documentation@beckhoff.com)



## 1.7 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)).

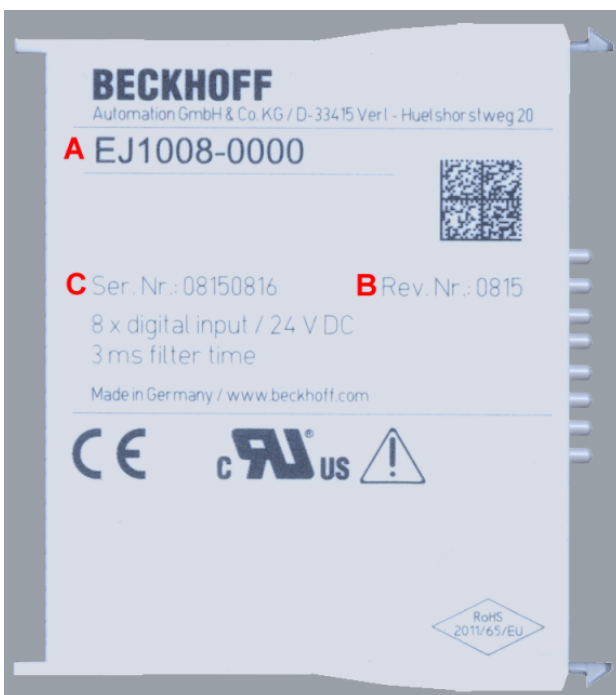


Fig. 1: Order identifier (A), Revision number (B) and serial number (C) using the example of EJ1008

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

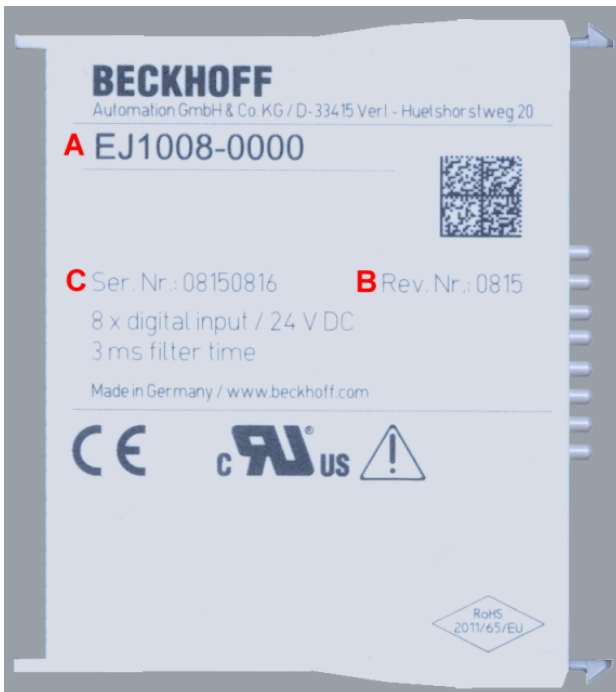


Fig. 2: Order identifier (A), revision number (B) and serial number (C) using the example of EJ1008

| 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 - firmware version: 08          |
| HH - hardware version                       | 16 - hardware version: 16          |

### 1.7.1 Beckhoff Identification Code (BIC)

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



Fig. 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              | <b>Beckhoff order number</b>  | 1P              | 8                                      | <b>1P</b> 072222            |
| 2        | Beckhoff Traceability Number (BTN) | <b>Unique serial number, see note below</b>                           | S               | 12                                     | <b>S</b> BTNk4p562d7        |
| 3        | Article description                | <b>Beckhoff article description, e.g. EL1008</b>                      | 1K              | 32                                     | <b>1K</b> EL1809            |
| 4        | Quantity                           | <b>Quantity in packaging unit, e.g. 1, 10, etc.</b>                   | Q               | 6                                      | <b>Q</b> 1                  |
| 5        | Batch number                       | Optional: Year and week of production                                 | 2P              | 14                                     | <b>2P</b> 401503180016      |
| 6        | ID/serial number                   | Optional: Present-day serial number system, e.g. with safety products | 51S             | 12                                     | <b>51S</b> 678294104        |
| 7        | Variant number                     | Optional: Product variant number on the basis of standard products    | 30P             | 12                                     | <b>30P</b> F971 ,<br>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 position 6. The data identifiers are marked in bold font for better display:

**1P**072222**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.

### NOTICE

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.

## 1.7.2 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.

The interface that the product can be electronically addressed by is crucial for the electronic readout.

### K-bus devices (IP20, IP67)

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

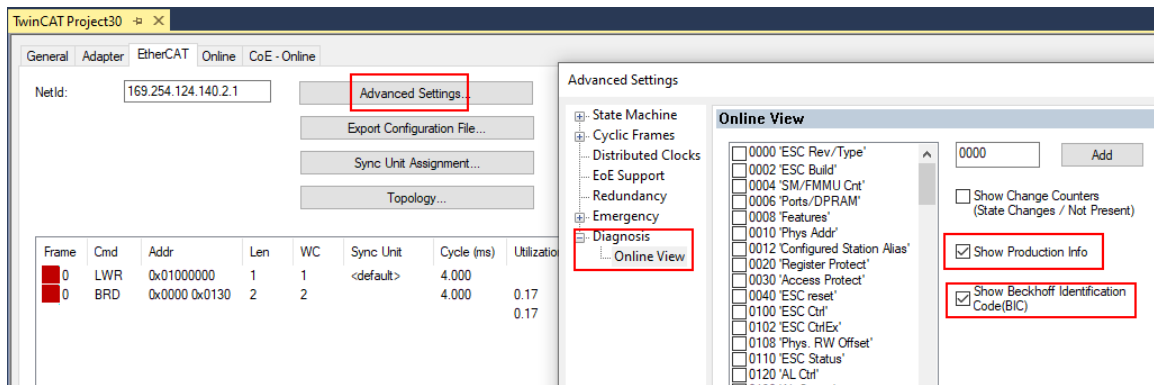
### EtherCAT devices (IP20, IP67)

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

Beckhoff also stores the eBIC in the ESI-EEPROM. The eBIC was introduced into Beckhoff IO production (terminals, box modules) in 2020; as of 2023, implementation is largely complete.

The user can electronically access the eBIC (if present) 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 "Show Beckhoff Identification Code (BIC)" checkbox under EtherCAT → Advanced Settings → Diagnostics:



- The BTN and its contents are then displayed:

| No | Addr | Name            | State | CRC | Fw | Hw | Production Data | ItemNo | BTN      | Description | Quantity | BatchNo | SerialNo |
|----|------|-----------------|-------|-----|----|----|-----------------|--------|----------|-------------|----------|---------|----------|
| 1  | 1001 | Term 1 (EK1100) | OP    | 0.0 | 0  | 0  | ---             |        |          |             |          |         |          |
| 2  | 1002 | Term 2 (EL1018) | OP    | 0.0 | 0  | 0  | 2020 KW36 Fr    | 072222 | k4p562d7 | EL1809      | 1        |         | 678294   |
| 3  | 1003 | Term 3 (EL3204) | OP    | 0.0 | 7  | 6  | 2012 KW24 Sa    |        |          |             |          |         |          |
| 4  | 1004 | Term 4 (EL2004) | OP    | 0.0 | 0  | 0  | ---             | 072223 | k4p562d7 | EL2004      | 1        |         | 678295   |
| 5  | 1005 | Term 5 (EL1008) | OP    | 0.0 | 0  | 0  | ---             |        |          |             |          |         |          |
| 6  | 1006 | Term 6 (EL2008) | OP    | 0.0 | 0  | 12 | 2014 KW14 Mo    |        |          |             |          |         |          |
| 7  | 1007 | Term 7 (EK1110) | OP    | 0   | 1  | 8  | 2012 KW25 Mo    |        |          |             |          |         |          |

- Note: As shown in the figure, the production data HW version, FW version, and production date, which have been programmed since 2012, can also be displayed with "Show production info".
- Access from the PLC: From TwinCAT 3.1. build 4024.24, the functions *FB\_EcReadBIC* and *FB\_EcReadBTN* for reading into the PLC are available in the *Tc2\_EtherCAT* library from v3.3.19.0.
- EtherCAT devices with a CoE directory may also have the object 0x10E2:01 to display their own eBIC, which can also be easily accessed by the PLC:

- 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    | 1P158442SBTN0008jckp1KELM3704 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 preferentially introduced into stock products in the course of necessary firmware revision.
- From TwinCAT 3.1. build 4024.24, the functions *FB\_EcCoEReadBIC* and *FB\_EcCoEReadBTN* for reading into the PLC are available in the *Tc2\_EtherCAT* library from v3.3.19.0
- The following auxiliary functions are available for processing the BIC/BTN data in the PLC in *Tc2\_Uilities* as of TwinCAT 3.1 build 4024.24
  - *F\_SplitBIC*: The function splits the Beckhoff Identification Code (BIC) *sBICValue* into its components using known identifiers and returns the recognized substrings in the *ST\_SplittedBIC* structure as a return value
  - *BIC\_TO\_BTN*: The function extracts the BTN from the BIC and returns it as a return value
- Note: If there is further electronic 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 written as an additional category in the ESI-EEPROM during device production. The structure of the ESI content is largely dictated by the ETG specifications, therefore the additional vendor-specific content is stored using a category in accordance with the ETG.2010. ID 03 tells all EtherCAT masters that they may not overwrite these data in the event of an update or restore the data after an ESI update.  
The structure follows the content of the BIC, see here. The EEPROM therefore requires approx. 50..200 bytes of memory.
- 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 which each have 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, and DeviceNet devices

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

### 1.7.3 Certificates

- The EtherCAT 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.

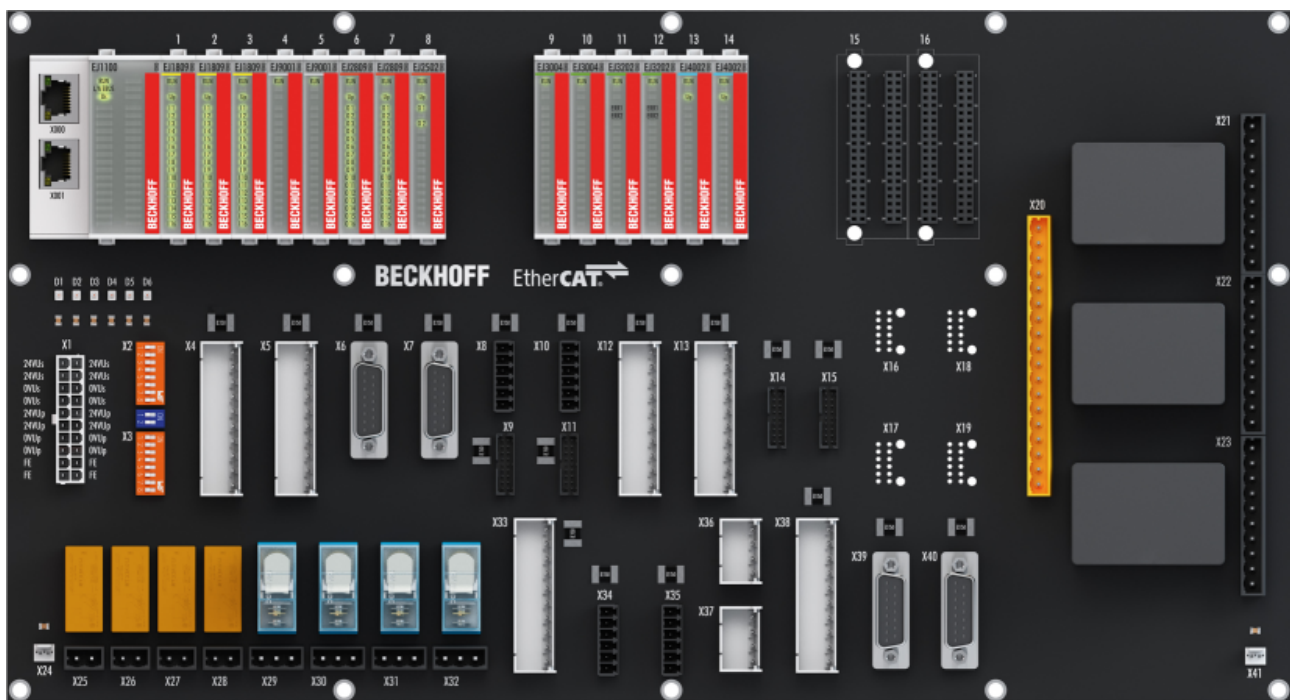


Fig. 6: EJ system sample

### 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 EJ6002 - Product description

### 3.1 Introduction

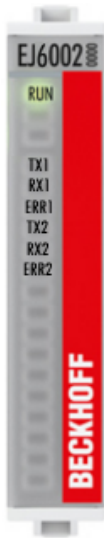


Fig. 7: EJ6002

#### **2-channel serial interface RS232, RS422 or RS485**

The serial interface EJ6002 allows the connection of up to two devices with RS232, RS422 or RS485 interfaces. The interfaces are electrically isolated from each other and from the EtherCAT.

Different operation modes (RS232, RS422 and RS485), baud rates and modes (e.g. termination, BIAS failsafe) can be configured via software.

The devices connected to the EJ6002 EtherCAT plug-in module communicate with the automation device via the coupler in an electrically isolated manner and thus guarantee a high level of interference immunity.

The active communication channel operates independently of the higher-level EtherCAT system in full duplex mode with 300 baud up to 256 kbaud.

In conjunction with the TwinCAT Virtual Serial COM Driver the EJ6002 can be used as a normal Windows COM interface.

## 3.2 Technical Data

| Technical data   | EJ6002   |
|--|--|
| Technology   | Serial interface   |
| Interfaces   | Individually selectable RS232, RS422 or RS485<br>Termination settings via software configuration |
| Data transfer channels                                 | 2  |
| Data transfer rate                                     | 300 ... 256000 Baud (freely configurable)  |
| Distributed Clocks                                     | -  |
| Data buffer  | 864 byte receive buffer, 128 byte transmit buffer per channel                                    |
| Electrical isolation                                   | 500 V (E-bus/field voltage)  |
| Current consumption via E-bus                          | 190 mA   |
| Permissible ambient temperature range during operation | -25°C ... +60°C  |
| Permissible ambient temperature range during storage   | -40°C ... +85°C  |
| Permissible relative air humidity                      | 95 %, no condensation  |
| Operating altitude                                     | max. 2,000 m   |
| Dimensions (W x H x D)                                 | approx. 12 mm x 66 mm x 55 mm  |
| Weight   | approx. 30 g   |
| Mounting   | on signal distribution board   |
| Pollution degree                                       | 2  |
| Mounting position                                      | <a href="#">Standard</a> [ <a href="#">▶ 27</a> ]  |
| Position of the coding pins<br><a href="#">▶ 30</a>    | 2 and 5  |
| Color coding   | grey   |
| Vibration/shock resistance                             | according 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 class                                       | EJ module: IP20<br>EJ system: dependent on the signal distribution board and housing             |
| Approvals/markings*                                    | CE, EAC, UKCA,<br>UL   |

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

### **i** 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 Pinout RS232

| EJ6002 RS232 |    |                   |                   |
|--------------|----|-------------------|-------------------|
| Pin#         |    | Signal            |                   |
| 1            | 2  | U <sub>EBUS</sub> | U <sub>EBUS</sub> |
| 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 | RTS_1             | RX_1              |
| 19           | 20 | TX_1              | CTS_1             |
| 21           | 22 | NC                | NC                |
| 23           | 24 | NC                | GND_Sensor1       |
| 25           | 26 | RTS_2             | RX_2              |
| 27           | 28 | TX_2              | CTS_2             |
| 29           | 30 | NC                | NC                |
| 31           | 32 | NC                | GND_Sensor2       |
| 33           | 34 | NC                | NC                |
| 35           | 36 | NC                | NC                |
| 37           | 38 | NC                | NC                |
| 39           | 40 | SGND              | SGND              |

| Signal            | Description                                 |
|-------------------|---|
| U <sub>EBUS</sub> | 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              |
| RTS_1             | Request to send signal (Channel 1)          |
| TX_1              | Transmit signal (Channel 1)                 |
| RTS_2             | Request to send signal (Channel 2)          |
| TX_2              | Transmit signal (Channel 2)                 |
| RX_1              | Receive signal (Channel 1)                  |
| CTS_1             | Clear to send signal (Channel 1)            |
| GND_Sensor1       | 0 V power supply (sensor1)                  |
| RX_2              | Receive signal (Channel 2)                  |
| CTS_2             | Clear to send signal (Channel 2)            |
| GND_Sensor2       | 0 V power supply (sensor2)                  |
| NC                | Do not connect                              |
| SGND              | Shield Ground                               |

**E-Bus contacts**

The power supply U<sub>EBUS</sub> is provided by the coupler and supplied from the supply voltage U<sub>S</sub> of the EtherCAT coupler.


**Signals**

**U<sub>P</sub>-Contacts**

The device has no U<sub>P</sub>-contacts.  
The power is supplied exclusively via U<sub>EBUS</sub>.

Fig. 8: EJ6002 - Pinout RS232

The PCB footprint can be downloaded from the Beckhoff [homepage](#).

| NOTICE  |   |
|---|---|
|  | <p><b>Damage may result!</b></p> <ul style="list-style-type: none"> <li>• The pins labeled "NC" must not be contacted.</li> <li>• Before installation and commissioning, please also read the chapters <a href="#">Installation of EJ modules [▶ 23]</a> and <a href="#">Commissioning [▶ 38]</a>!</li> </ul> |

### 3.4 Pinout RS422

| EJ6002 RS422 |    |            |             |
|--------------|----|------------|-------------|
| Pin#         |    | Signal     |             |
| 1            | 2  | $U_{EBUS}$ | $U_{EBUS}$  |
| 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 | TX_1+      | RX_1+       |
| 19           | 20 | TX_1-      | RX_1-       |
| 21           | 22 | NC         | NC          |
| 23           | 24 | NC         | GND_Sensor1 |
| 25           | 26 | TX_2+      | RX_2+       |
| 27           | 28 | TX_2-      | RX_2-       |
| 29           | 30 | NC         | NC          |
| 31           | 32 | NC         | GND_Sensor2 |
| 33           | 34 | NC         | NC          |
| 35           | 36 | NC         | NC          |
| 37           | 38 | NC         | NC          |
| 39           | 40 | SGND       | SGND        |

**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.

**Signals**

**$U_P$ -Contacts**

The device has no  $U_P$ -contacts.  
The power is supplied exclusively via  $U_{EBUS}$ .

| Signal      | Description                                     |
|-------------|---|
| $U_{EBUS}$  | E-Bus power supply 3.3 V                        |
| GND         | E-Bus GND signal. Don't connect with 0V $U_P$ ! |
| RXn+        | Positive E-Bus receive signal                   |
| RXn-        | Negative E-Bus receive signal                   |
| TXn+        | Positive E-Bus transmit signal                  |
| TXn-        | Negative E-Bus transmit signal                  |
| TX_1+       | Positive transmit signal (Channel 1)            |
| TX_1-       | Negative transmit signal (Channel 1)            |
| TX_2+       | Positive transmit signal (Channel 2)            |
| TX_2-       | Negative transmit signal (Channel 2)            |
| RX_1+       | Positive receive signal (Channel 1)             |
| RX_1-       | Negative receive signal (Channel 1)             |
| GND_Sensor1 | 0 V power supply (sensor1)                      |
| RX_2+       | Positive receive signal (Channel 2)             |
| RX_2-       | Negative receive signal (Channel 2)             |
| GND_Sensor2 | 0 V power supply (sensor2)                      |
| NC          | Do not connect                                  |
| SGND        | Shield Ground                                   |

Fig. 9: EJ6002 - Pinout RS422

The PCB footprint can be downloaded from the Beckhoff [homepage](#).

#### NOTICE



#### Damage may result!

- The pins labeled "NC" must not be contacted.
- Before installation and commissioning, please also read the chapters [Installation of EJ modules \[▶ 23\]](#) and [Commissioning \[▶ 38\]](#)!

### 3.5 Pinout RS485

| EJ6002 RS485 |    |                   |                   |
|--------------|----|-------------------|-------------------|
| Pin#         |    | Signal            |                   |
| 1            | 2  | U <sub>EBUS</sub> | U <sub>EBUS</sub> |
| 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 | DATA_1+           | NC                |
| 19           | 20 | DATA_1-           | NC                |
| 21           | 22 | NC                | NC                |
| 23           | 24 | NC                | GND Sensor1       |
| 25           | 26 | DATA_2+           | NC                |
| 27           | 28 | DATA_2-           | NC                |
| 29           | 30 | NC                | NC                |
| 31           | 32 | NC                | GND Sensor2       |
| 33           | 34 | NC                | NC                |
| 35           | 36 | NC                | NC                |
| 37           | 38 | NC                | NC                |
| 39           | 40 | SGND              | SGND              |

**E-Bus contacts**

The power supply U<sub>EBUS</sub> is provided by the coupler and supplied from the supply voltage U<sub>S</sub> of the EtherCAT coupler.

**Signals**

**U<sub>P</sub>-Contacts**


The device has no U<sub>P</sub>-contacts. The power is supplied exclusively via U<sub>EBUS</sub>.

| Signal            | Description  |
|-------------------|--|
| U <sub>EBUS</sub> | E-Bus power supply 3.3 V                                 |
| GND               | E-Bus GND signal. Don't connect with 0V U <sub>P</sub> ! |
| RXn+              | Positive E-Bus receive signal                            |
| RXn-              | Negative E-Bus receive signal                            |
| TXn+              | Positive E-Bus transmit signal                           |
| TXn-              | Negative E-Bus transmit signal                           |
| DATA_1+           | Positive DATA signal (Channel 1)                         |
| DATA_1-           | Negative DATA signal (Channel 1)                         |
| DATA_2+           | Positive DATA signal (Channel 2)                         |
| DATA_2-           | Negative DATA signal (Channel 2)                         |
| GND Sensor1       | 0 V power supply (sensor1)                               |
| GND Sensor2       | 0 V power supply (sensor2)                               |
| NC                | Do not connect   |
| SGND              | Shield Ground  |

Fig. 10: EJ6002 - Pinout RS485

The PCB footprint can be downloaded from the Beckhoff homepage.

**NOTICE**



**Damage may result!**

- The pins labeled "NC" must not be contacted.
- Before installation and commissioning, please also read the chapters [Installation of EJ modules](#) [▶ 23] and [Commissioning](#) [▶ 38]!

### 3.6 LEDs

| LED No. | EJ6002 |
|---------|--------|
| A       | RUN    |
| B       |        |
| C       |        |
|         |        |
| 1       | TX1    |
| 2       | RX1    |
| 3       | ERR 1  |
| 4       | TX2    |
| 5       | RX2    |
| 6       | ERR 2  |
| 7       |        |
| 8       |        |
| 9       |        |
| 10      |        |
| 11      |        |
| 12      |        |
| 13      |        |
| 14      |        |
| 15      |        |
| 16      |        |

Fig. 11: EJ6002 - LEDs

| LED          | Color | Display      | State            | Description  |
|--------------|-------|--------------|------------------|--|
| RUN          | green | off          | Init             | State of the <u>EtherCAT State Machine</u> <b>INIT</b> = initialization of the plug-in module  |
|              |       | flashing     | Pre-Operational  | State of the EtherCAT State Machine: <b>PREOP</b> = function for mailbox communication and different default settings set                                      |
|              |       | single flash | Safe-Operational | State of the EtherCAT State Machine: <b>SAFEOP</b> = verification of the <u>Sync-Manager</u> channels and the distributed clocks. Outputs remain in safe state |
|              |       | on           | Operational      | State of the EtherCAT State Machine: <b>OP</b> = normal operating state; mailbox and process data communication are possible                                   |
|              |       | flickering   | Bootstrap        | State of the EtherCAT State Machine: <b>BOOTSTRAP</b> = function for <u>firmware updates</u> of the plug-in module   |
| TX1, TX2     | green | on           | -                | Data is transmitted via the respective signal line (channel 1, channel 2)  |
|              |       | off          | -                | No activity on the transmission line   |
| RX1, RX2     | green | on           | -                | Data are received via the respective signal line (channel 1, channel 2)  |
|              |       | off          | -                | No activity on the receive line  |
| ERR1<br>ERR2 | red   | off          | -                | No error   |
|              |       | on           | -                | An error has occurred.   |

## 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 supply provides safe electrical isolation and limitation of the voltage without a connection to the protective conductor, a PELV 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:

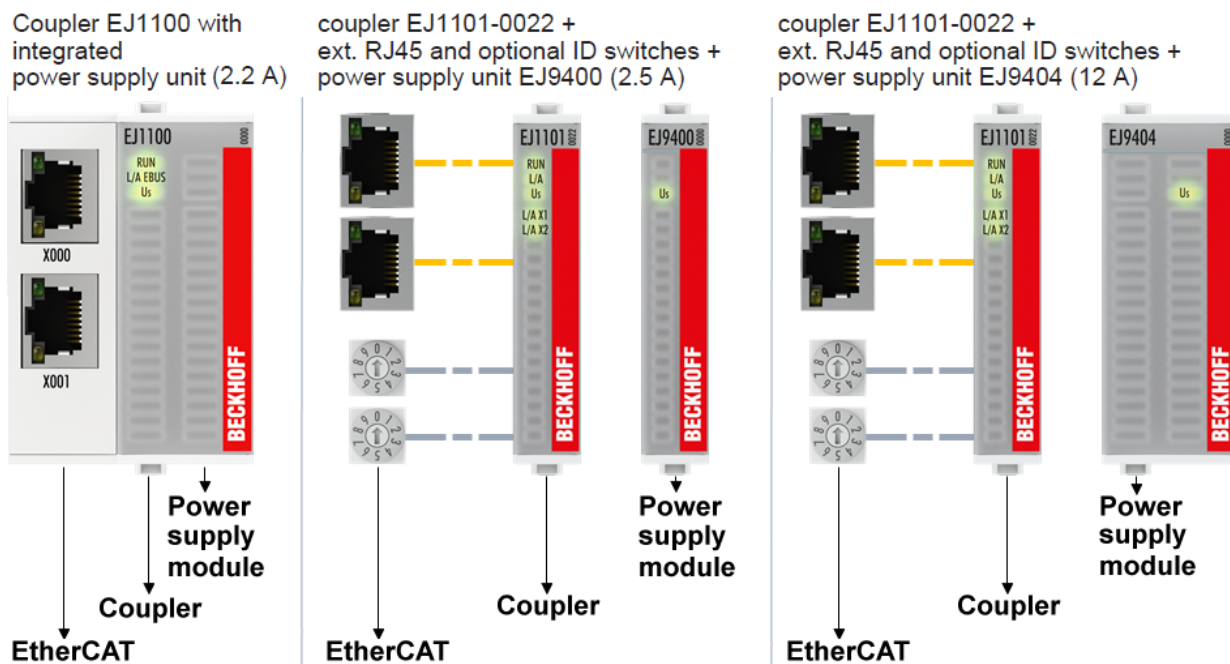


Fig. 12: E-bus power supply with EJ1100 or EJ1101-0022 + EJ940x

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  $U_s$  of 24 V<sub>DC</sub> (-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  $U_p$  of 24 V<sub>DC</sub> 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<sub>DC</sub>).



Fig. 13: PCB with Embedded PC, EK1110-0043 and EJxxxx, rear view EK1110-0043

## 4.2 Note on load voltage supply

### ⚠ WARNING

#### Load voltage supply

Some devices permit an additional load voltage, e.g. 48 V DC, to be connected for the operation of a motor. In order to avoid stray currents on the protective conductor during operation, EN 60204-1:2018 provides for the possibility that the negative pole of the load voltage does not necessarily have to be connected to the protective conductor system (SELV).

Therefore, the load voltage supply should be designed as an SELV supply.



### 4.3 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_2xjr45_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. 14: EJxxxx - Dimensions

The technical drawings can be downloaded from the Beckhoff [homepage](#). The drawings are named as described in the drawing below.

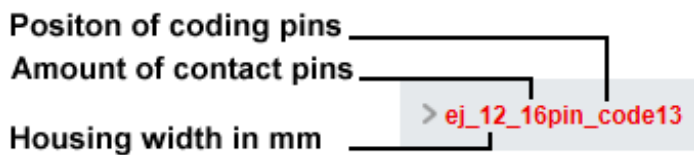


Fig. 15: Naming of the technical drawings

## 4.4 Installation positions and minimum distances

### 4.4.1 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.

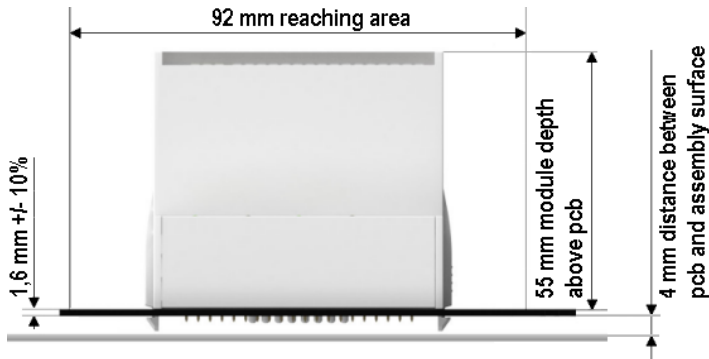


Fig. 16: Mounting distances EJ module - PCB

#### **i** 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](#) [▶ 27](#)) 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.

### 4.4.2 Installation positions

**NOTICE**

**Constraints regarding installation position and operating temperature range**

Please refer to the [technical data \[►\\_18\]](#) 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.

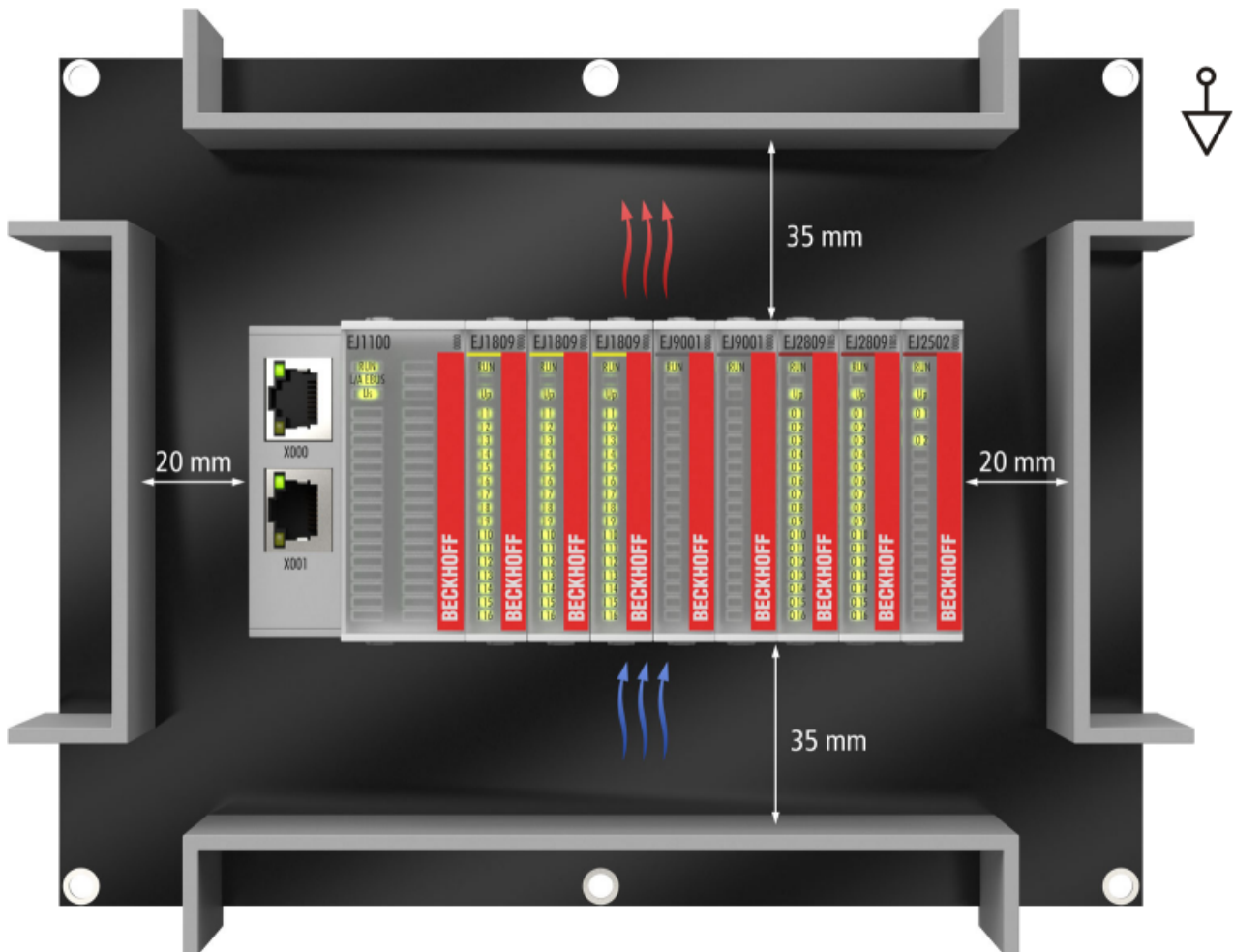


Fig. 17: Recommended distances for standard installation position

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

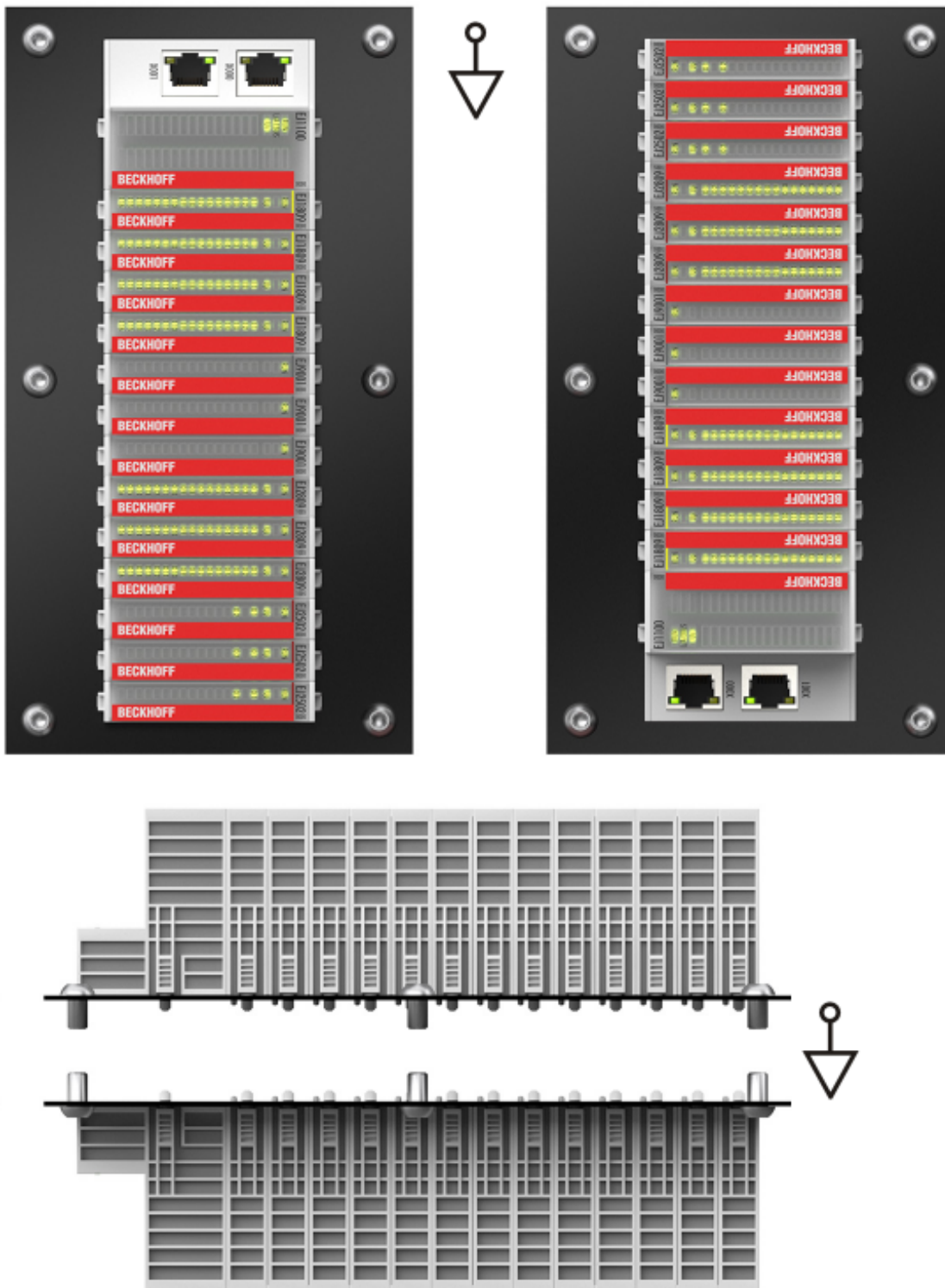


Fig. 18: Other installation positions

## 4.5 Codings

### 4.5.1 Color coding

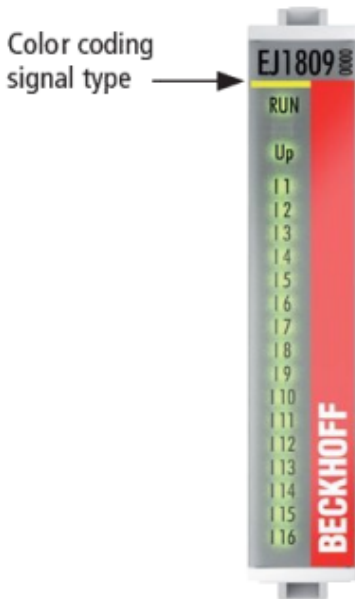


Fig. 19: EJ modules color code; sample: EJ1809

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            |

### 4.5.2 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.

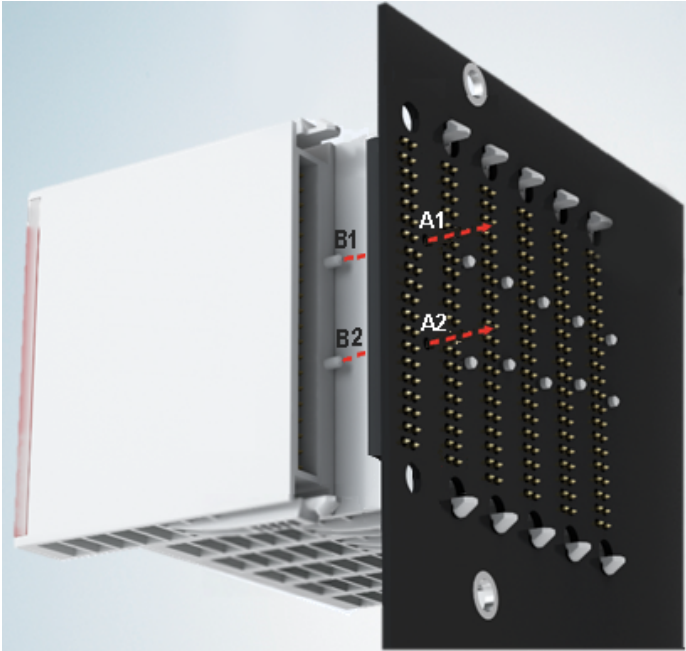


Fig. 20: Mechanical position coding with coding pins (B1 and B2) and coding holes (A1 and A2)

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.

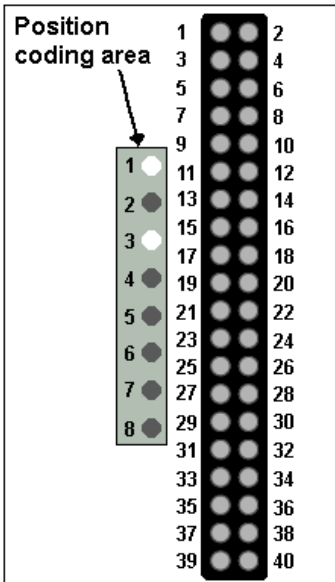


Fig. 21: Pin coding; sample: digital input modules

## 4.6 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

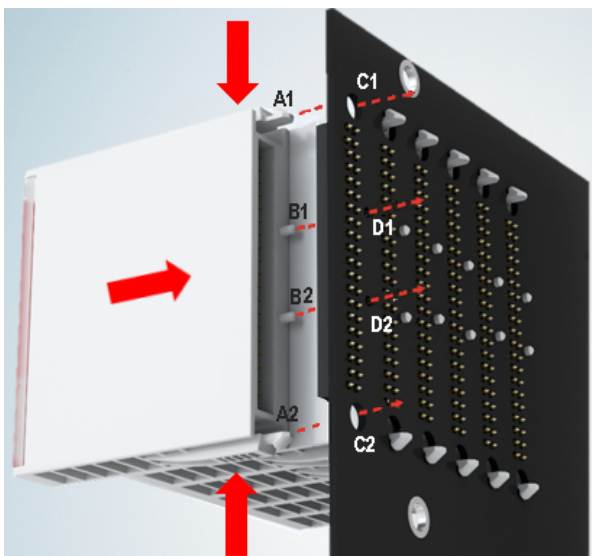
#### Risk of injury through electric shock and damage to the device!

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

### NOTICE

#### Risk of damage to components through electrostatic discharge!

Observe the regulations for ESD protection.



A1 / A2: Latching lugs top / bottom

B1 / B2: Coding pins

C1 / C2: Mounting holes

D1 / D2: Coding holes

Installation of EJ modules

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.

### NOTICE

#### Ensure safe latching of the modules on the signal distribution board

- 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.

## 4.7 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

### 4.7.1 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.

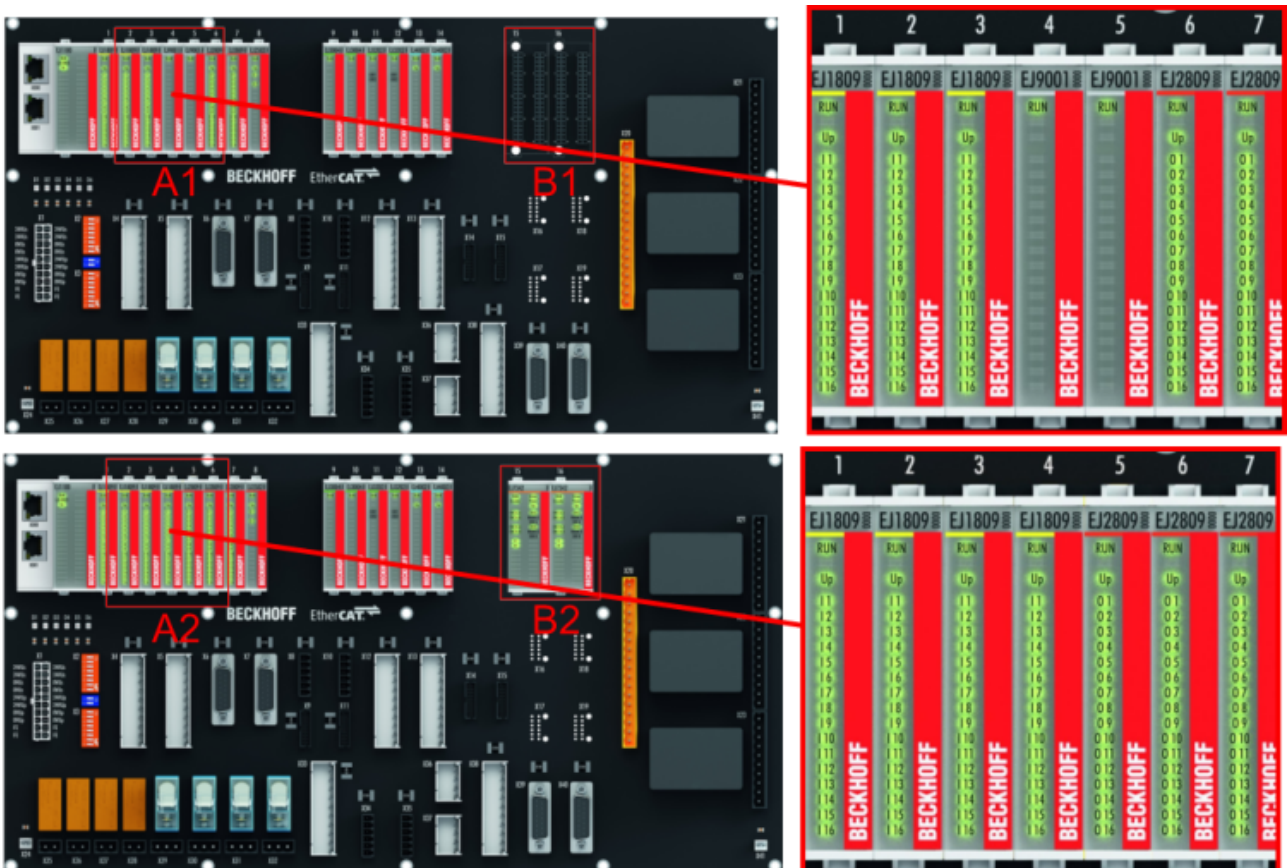


Fig. 22: Sample: Exchanging placeholder modules and assigning reserve slots

#### ● E-bus supply

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



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

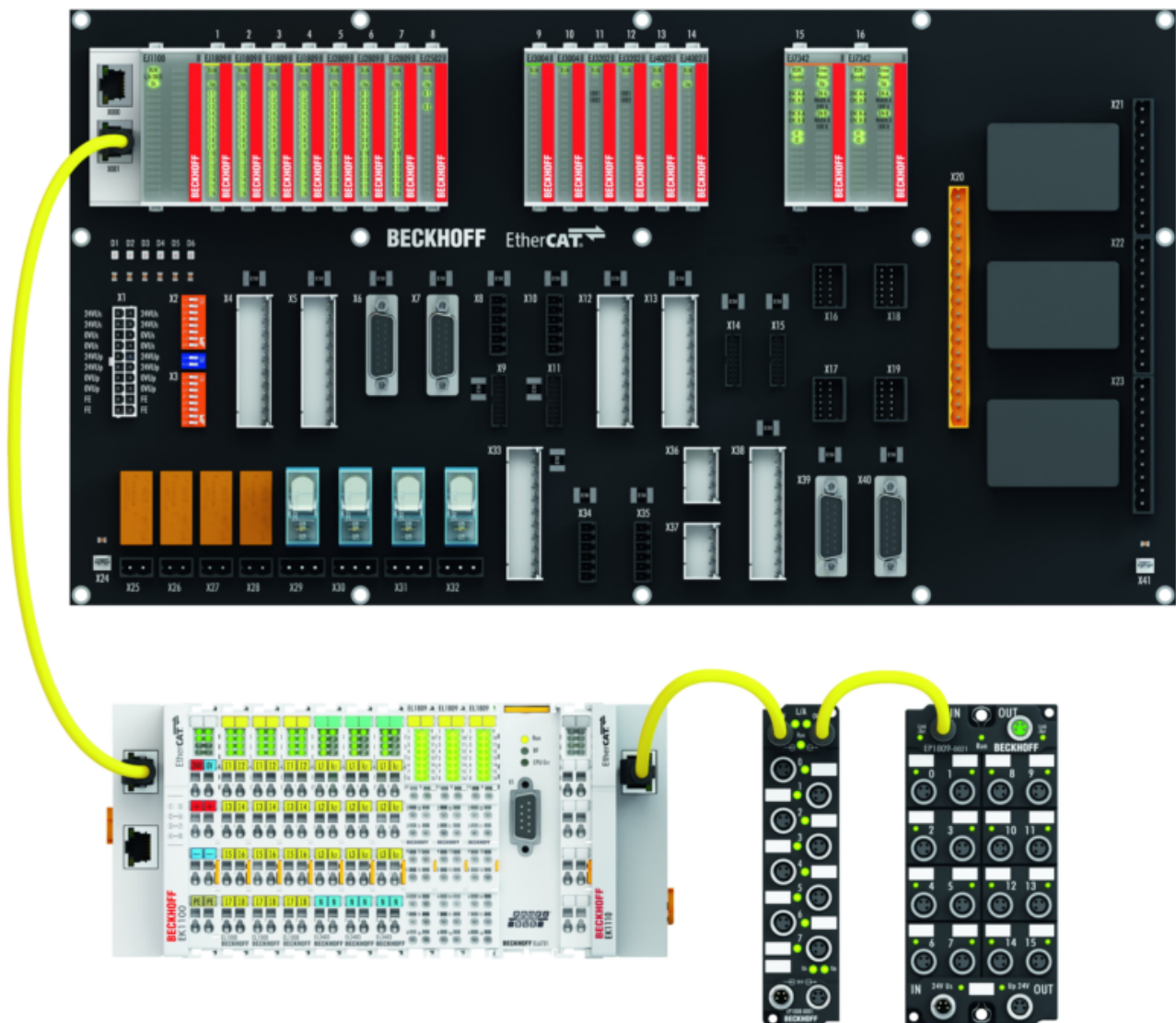


Fig. 23: Example of extension via an Ethernet/EtherCAT connection

## 4.8 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 (ELxxx) 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. 24: 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).

**NOTICE**



**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. 25: Example for the connection of a C6015 IPC to an EJ system

## 4.9 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.

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

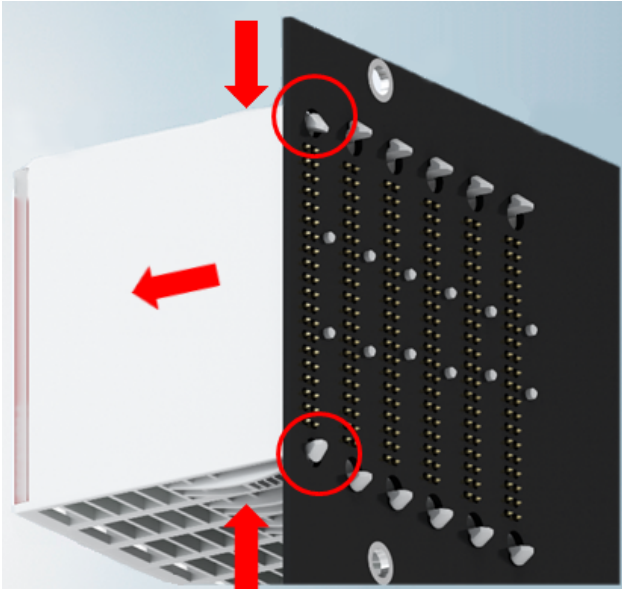


Fig. 26: Disassembly of EJ modules

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.

## 4.10 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 Commissioning

### 6.1 Reference to documentation EL600x

Detailed documentation on the commissioning of the EJ6002 module is being prepared.

#### NOTICE



#### Damage to devices or loss of data

The descriptions and notes on the commissioning of the EL600x EtherCAT Terminals are transferable to the EJ6002 EtherCAT plug-in module.

Before commissioning, read the detailed description of the process data, operation modes and parameterization in the [EL600x](#) documentation.

### 6.2 EJ6002 - 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.

#### NOTICE



#### 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 object index 0x1011
  - [Configuration data \[▶ 39\]](#) index 0x80n0
- Profile-specific objects:
  - [Input data \[▶ 42\]](#) index 0x60n0, 0x60n1
  - [Output data \[▶ 43\]](#) 0x70n0, 0x70n1
  - [Information and diagnostic \[▶ 44\]](#) data index 0xA0n0, 0xF000, 0xF008, 0xF010
- [Standard objects \[▶ 44\]](#)

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

## 6.2.1 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 <sub>dec</sub> )       |
| 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 <sub>dec</sub> ) |

## 6.2.2 Configuration data

### Index 80n0 COM Settings Ch.1 (n = 0), Ch.2 (n = 1)

| Index (hex) | Name                              | Meaning  | Data type | Flags | Default                   |
|-------------|-----------------------------------|--|-----------|-------|---------------------------|
| 80n0:0      | COM Settings Ch.(n+1)             | Max. Subindex (hex)  | UINT8     | RO    | 0x1D (29 <sub>dec</sub> ) |
| 80n0:01     | Enable RTS/CTS                    | FALSE:<br>RTS/CTS not enabled<br><br>TRUE:<br>enabled  | BOOLEAN   | RW    | 0x01 (1 <sub>dec</sub> )  |
| 80n0:02     | Enable XON/XOFF supported tx data | FALSE:<br>XON/XOFF is not supported for send data<br><br>TRUE:<br>XON/XOFF is supported for send data  | BOOLEAN   | RW    | 0x00 (0 <sub>dec</sub> )  |
| 80n0:03     | Enable XON/XOFF supported rx data | FALSE:<br>XON/XOFF is not supported for input data<br><br>TRUE:<br>XON/XOFF is supported for input data  | BOOLEAN   | RW    | 0x00 (0 <sub>dec</sub> )  |
| 80n0:04     | Enable send FIFO data continuous  | FALSE:<br>No continuous sending of data from the FIFO<br><br>TRUE:<br>Continuous sending of data from the FIFO enabled: <ul style="list-style-type: none"> <li>The send buffer is filled (up to 128 bytes) by the controller.</li> <li>The filled buffer content is sent with a rising edge of the bit "Send continuous" (0x70n0:04).</li> <li>The module acknowledges the data transfer to the controller by setting the bit "Init accepted" (0x60n0:03). "Init accepted" is reset with "Send continuous".</li> </ul> | BOOLEAN   | RW    | 0x00 (0 <sub>dec</sub> )  |
| 80n0:05     | Enable transfer rate optimization | FALSE:<br>Transfer rate optimization switched off<br><br>TRUE:<br>Transfer rate optimization switched on:<br>The content of the input buffer is automatically transferred into the process image if <ul style="list-style-type: none"> <li>no further byte was received for approx. 16 bit times (the time it would have taken to receive 2 bytes) after receiving data.</li> <li>the process image is filled</li> </ul>   | BOOLEAN   | RW    | 0x01 (1 <sub>dec</sub> )  |
| 80n0:06     | Enable half duplex                | FALSE:<br>Full-duplex mode<br><br>TRUE:<br>Half-duplex mode  | BOOLEAN   | RW    | 0x00 (0 <sub>dec</sub> )  |

| Index (hex) | Name                        | Meaning  | Data type | Flags | Default                           |
|-------------|-----------------------------|--|-----------|-------|-----------------------------------|
| 80n0:11     | Baud rate                   | Configurable baud rates:<br>1 <sub>dec</sub> : 300 baud<br>2 <sub>dec</sub> : 600 baud<br>3 <sub>dec</sub> : 1200 baud<br>4 <sub>dec</sub> : 2400 baud<br>5 <sub>dec</sub> : 4800 baud<br>6 <sub>dec</sub> : 9600 baud (default)<br>7 <sub>dec</sub> : 19.2 kbaud<br>8 <sub>dec</sub> : 38.4 kbaud<br>9 <sub>dec</sub> : 57.6 kbaud<br>10 <sub>dec</sub> : 115.2 kbaud<br>11 <sub>dec</sub> : Use explicit baud rate (Subindex 0x80n0:1B)  | BIT4      | RW    | 0x06 (6 <sub>dec</sub> )          |
| 80n0:15     | Data frame                  | Available coding<br>1 <sub>dec</sub> : 7E1<br>2 <sub>dec</sub> : 701<br>3 <sub>dec</sub> : 8N1 (default)<br>4 <sub>dec</sub> : 8E1<br>5 <sub>dec</sub> : 801<br>8 <sub>dec</sub> : 7N2<br>9 <sub>dec</sub> : 7E2<br>10 <sub>dec</sub> : 702<br>11 <sub>dec</sub> : 8N2<br>12 <sub>dec</sub> : 8E2<br>13 <sub>dec</sub> : 802   | BIT4      | RW    | 0x03 (3 <sub>dec</sub> )          |
| 80n0:1A     | Rx buffer full notification | The value specifies the number of data in the receive FIFO from which the bit in index 0x60n0:04 "buffer full" is set.   | UINT16    | RW    | 0x0360 (864 <sub>dec</sub> )      |
| 80n0:1B     | Explicit baud rate          | The desired baud rate can be entered here as a number<br>All baud rates between 300 and 256000 baud are supported.   | UINT32    | RW    | 0x00000384 (9600 <sub>dec</sub> ) |
| 80n0:1D     | Mode of operation           | Selection of the communication standard:<br>16 <sub>dec</sub> RS232 (default)<br>Rx, Tx, RTS, CTS are controlled automatically.<br>17 <sub>dec</sub> RS232, manual operation<br>The input of Rx and CTS can be read from the ext. status (0x60n1 [▶ 42]).<br>The outputs Tx and RTS can be written by the ext. controlword (0x70n1 [▶ 43]).<br>Sending and receiving data is not handled by the firmware.<br>18 <sub>dec</sub> RS232, manual flow control<br>Sending and receiving data is controlled by the firmware.<br>The input state of CTS can be read from the ext. status (0x60n1 [▶ 42]).<br>The state of the RTS output can be set in the ext. controlword (0x70n1 [▶ 43]).<br>32 <sub>dec</sub> RS422 (Busdevice)<br>RS422 is typically used as point to point connection between two devices. In this mode, this module can listen to one of the devices.<br>Transmitting data is possible but not recommended. This could interfere an ongoing communication.<br>Termination and BIAS/Failsafe resistors should be handled by the two main devices. So they are disabled in this mode.<br>33 <sub>dec</sub> RS422, Terminated RX<br>Point to point connection between two devices.<br>The termination resistor on the receiver is enabled.<br>34 <sub>dec</sub> RS422, BIAS/Failsafe RX<br>Point to point connection between two devices.<br>The BIAS/Failsafe resistors on the receiver are enabled. | UINT32    | RW    | 0x00000010 (16 <sub>dec</sub> )   |



| Index (hex) | Name              | Meaning   | Data type | Flags | Default                            |
|-------------|-------------------|---|-----------|-------|------------------------------------|
| 80n0:1D     | Mode of operation | 35 <sub>dec</sub><br><i>RS422, Terminated and BIAS/Failsafe RX (Enddevice)</i><br>Point to point connection between two devices. The termination resistor and the BIAS/Failsafe resistors on the receiver are enabled. This should be the default mode for a RS422 configuration. | UINT32    | RW    | 0x00000010<br>(16 <sub>dec</sub> ) |
|             |                   | 64 <sub>dec</sub><br><i>RS485 (Busdevice)</i><br>This configuration will not enable the termination or the BIAS/Failsafe resistors. It is designed for devices which are connected in between of the two terminating devices.   |           |       |                                    |
|             |                   | 65 <sub>dec</sub><br><i>RS485, Terminated</i><br>RS485 configuration with enabled termination resistor  |           |       |                                    |
|             |                   | 66 <sub>dec</sub><br><i>RS485, BIAS/Failsafe</i><br>RS485 configuration with enabled termination resistor   |           |       |                                    |
|             |                   | 67 <sub>dec</sub><br><i>RS485, Terminated and BIAS/Failsafe (Enddevice)</i><br>RS485 configuration with enabled termination resistor and enabled BIAS/Failsafe resistors. This should be the default configuration for both endpoints of the RS485 network.                       |           |       |                                    |

## 6.2.3 Input data

### Index 60n0 COM Inputs Ch. 1 (n = 0), Ch. 2 (n = 1)

| Index (hex) | Name                 | Meaning  | Data type | Flags | Default                      |
|-------------|----------------------|--|-----------|-------|------------------------------|
| 60n0:0      | COM Inputs Ch. (n+1) | Length of this object  | UINT8     | RO    | 0x26<br>(38 <sub>dec</sub> ) |
| 60n0:01     | Transmit accepted    | The module acknowledges the receipt of data with a change of state of this bit.<br>Only then are new data transferred from the controller to the module.   | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:02     | Receive request      | By changing the state of this bit, the module notifies the controller that the number of bytes indicated in index 0x60n0:09 "Input length" are located in the DataIn bytes.<br>The controller must acknowledge receipt of the data with a change of state of the "Receive accepted" bit (index 0x70n0:02).<br>Only then is new data transferred from the module to the controller. | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:03     | Init Accepted        | FALSE:<br>The module is once again ready for serial data exchange.<br>TRUE:<br>The initialization was executed by the module.  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:04     | Buffer full          | The receive FIFO is full.<br><b>Notice</b> All incoming data will be lost from this point on!  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:05     | Parity error         | A parity error has occurred.   | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:06     | Framing error        | A framing error has occurred.  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:07     | Overrun error        | An overrun error has occurred.   | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:09     | Input length         | Number of input bytes ready for transfer from the module to the controller.  | UINT8     | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 60n0:11     | Data In 0            | Input byte 0   | UINT8     | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| ...         | ...                  | ...  | ...       | ...   | ...                          |
| 60n0:26     | Data In 21           | Input byte 21  | UINT8     | RO    | 0x00<br>(0 <sub>dec</sub> )  |

### Index 60n1 Com Ext. Inputs Ch. 1 (n = 0), Ch.2 (n = 1)

| Index (hex) | Name                     | Meaning   | Data type | Flags | Default                     |
|-------------|--------------------------|---|-----------|-------|-----------------------------|
| 60n1:0      | Com Ext. Inputs Ch.(n+1) | Length of this object   | UINT8     | RO    | 0x02<br>(2 <sub>dec</sub> ) |
| 60n1:01     | Rx                       | In operation mode<br>"RS232, manual operation" (CoE 0x80n0:1D [▶ 39] = 17 <sub>dec</sub> ) or<br>"RS232, manual flow control" (CoE 0x80n0:1D [▶ 39] = 18 <sub>dec</sub> ) the level of Rx input is displayed here.  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> ) |
| 60n1:02     | CTS                      | In operation mode<br>"RS232, manual operation" (CoE 0x80n0:1D [▶ 39] = 17 <sub>dec</sub> ) or<br>"RS232, manual flow control" (CoE 0x80n0:1D [▶ 39] = 18 <sub>dec</sub> ) the level of CTS input is displayed here. | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> ) |

## 6.2.4 Output data

### Index 0x70n0 COM Outputs Ch. 1 (n = 0), Ch. 2 (n = 1)

| Index (hex) | Name                      | Meaning   | Data type | Flags | Default                      |
|-------------|---------------------------|---|-----------|-------|------------------------------|
| 70n0:0      | COM Outputs Ch. 1 + Ch. 2 | Max. SubIndex (hex)   | UINT8     | RO    | 0x26<br>(38 <sub>dec</sub> ) |
| 70n0:01     | Transmit request          | By changing the state of this bit, the controller informs the module that the DataOut bytes contain the number of bytes displayed in index 0x70n0:099 "Output length".<br>The module acknowledges receipt of the data by changing the state of the index 0x60n0:01 "TransmitAccepted [▶ 42]" bit.<br>Only now new data can be transferred from the controller to the module.  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 70n0:02     | Receive accepted          | The controller acknowledges receipt of data by changing the state of this bit.<br>Only then new data can be transferred from the terminal to the controller.  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 70n0:03     | Init request              | 0:<br>The controller once again requests the module to prepare for serial data exchange.<br>1:<br>The controller requests the module for initialization.<br>The transmit and receive functions will be blocked<br>The FIFO pointer will be reset and the interface will be initialized with the values of the responsible Settings object. The execution of the initialization will be acknowledged by the module with the "Init accepted [▶ 42]" bit in index 0x60n0:03. | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 70n0:04     | Send continuous           | Continuous sending of data from the FIFO.<br>The send buffer is filled (up to 128 bytes) by the controller. The filled buffer contents will be sent on the rising edge of the bit.<br>If the data has been transmitted, the module informs the controller by setting the "Init accepted [▶ 42]" bit in index 0x60n0:03. "Init accepted" is cleared with "Send continuous".  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 70n0:09     | Output length             | Number of output bytes available for transfer from the controller to the module.  | UINT8     | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| 70n0:11     | Data Out 0                | Output byte 0   | UINT8     | RO    | 0x00<br>(0 <sub>dec</sub> )  |
| ...         | ...                       | ....  | ...       | ...   | ...                          |
| 70n0:26     | Data Out 21               | Output byte 21  | UINT8     | RO    | 0x00<br>(0 <sub>dec</sub> )  |

### Index 0x70n1 COM Ext. Outputs Ch. 1 (n = 0), Ch. 2 (n = 1)

| Index (hex) | Name                           | Meaning  | Data type | Flags | Default                     |
|-------------|--------------------------------|--|-----------|-------|-----------------------------|
| 70n1:0      | COM Ext. Outputs Ch. 1 + Ch. 2 | Max. SubIndex (hex)  | UINT8     | RO    | 0x02<br>(2 <sub>dec</sub> ) |
| 70n1:01     | Tx                             | Transmit signal<br>In operation mode<br>"RS232, manual operation" (CoE 0x80n0:1D [▶ 39] = 17 <sub>dec</sub> ) the level of Tx output can be set manually here.<br>Sending and receiving data is not handled by the firmware in this operation mode.  | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> ) |
| 70n1:02     | RTS                            | Request to send signal<br>In operation mode<br>"RS232, manual operation" (CoE 0x80n0:1D [▶ 39] = 17 <sub>dec</sub> ) or "RS232, manual flow control" (CoE 0x80n0:1D [▶ 39] = 18 <sub>dec</sub> ) the level of RTS output can be set manually here.<br>Sending and receiving data is not handled by the firmware in "RS232, manual operation" mode. | BOOLEAN   | RO    | 0x00<br>(0 <sub>dec</sub> ) |

## 6.2.5 Information and diagnostic data

### Index 0xA0n0 COM Diag data Ch. 1 (n = 0), Ch. 2 (n = 1)

| Index (hex) | Name                         | Meaning   | Data type | Flags | Default                    |
|-------------|------------------------------|---|-----------|-------|----------------------------|
| A0n0:0      | COM Diag data Ch. 1 + Ch. 2  | Max. SubIndex (hex)   | UINT8     | RO    | 0x21 (33 <sub>dec</sub> )  |
| A0n0:01     | Buffer overflow              | A buffer overflow has occurred.   | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )   |
| A0n0:02     | Parity error                 | A parity error has occurred.  | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )   |
| A0n0:03     | Framing error                | A framing error has occurred  | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )   |
| A0n0:04     | Overrun error                | An overrun error has occurred.  | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )   |
| A0n0:05     | Buffer full                  | The reception FIFO is full.<br><b>Notice</b> All incoming data will be lost from this point on! | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )   |
| A0n0:06     | Power supply overload        | Overload of the internal power supply   | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )   |
| A0n0:11     | Data bytes in send buffer    | Number of data bytes in the send FIFO   | UINT16    | RO    | 0x0000 (0 <sub>dec</sub> ) |
| A0n0:21     | Data bytes in receive buffer | Number of data bytes in the receive FIFO  | UINT16    | RO    | 0x0000 (0 <sub>dec</sub> ) |

### 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 <sub>dec</sub> )    |
| F000:01     | Module index distance     | Index distance of the objects of the individual channels | UINT16    | RO    | 0x0010 (16 <sub>dec</sub> ) |
| F000:02     | Maximum number of modules | Number of channels                                       | UINT16    | RO    | 0x0002 (2 <sub>dec</sub> )  |

### Index F008 Code word

| Index (hex) | Name      | Meaning  | Data type | Flags | Default                        |
|-------------|-----------|----------|-----------|-------|--------------------------------|
| F008:0      | Code word | reserved | UINT32    | RW    | 0x00000000 (0 <sub>dec</sub> ) |

### Index F010 Module list

| Index (hex) | Name         | Meaning               | Data type | Flags | Default                          |
|-------------|--------------|-----------------------|-----------|-------|----------------------------------|
| F010:0      | Module list  | Length of this object | UINT8     | RW    | 0x02 (2 <sub>dec</sub> )         |
| F010:01     | SubIndex 001 | Revision              | UINT32    | RW    | 0x00000258 (600 <sub>dec</sub> ) |
| F010:02     | SubIndex 002 | -                     | UINT32    | RW    | 0x00000258 (600 <sub>dec</sub> ) |

### Index F081 Download revision

| Index (hex) | Name         | Meaning                    | Data type | Flags | Default                          |
|-------------|--------------|----------------------------|-----------|-------|----------------------------------|
| F010:0      | Module list  | Max. Subindex              | UINT8     | RW    | 0x01 (1 <sub>dec</sub> )         |
| F010:01     | SubIndex 001 | Configured module revision | UINT32    | RW    | 0x00000258 (600 <sub>dec</sub> ) |

## 6.2.6 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: <ul style="list-style-type: none"> <li>The Lo-Word contains the CoE profile used (5001).</li> <li>The Hi-Word contains the module profile according to the modular device profile.</li> </ul> | UINT32    | RO    | 0x02581389 (39326601 <sub>dec</sub> ) |

**Index 1008 Device name**

| Index (hex) | Name        | Meaning                           | Data type | Flags | Default |
|-------------|-------------|-----------------------------------|-----------|-------|---------|
| 1008:0      | Device name | Device name of the EtherCAT slave | STRING    | RO    | EJ6002  |

**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 1018 Identity**

| Index (hex) | Name          | Meaning   | Data type | Flags | Default                                 |
|-------------|---------------|---|-----------|-------|---|
| 1018:0      | Identity      | Information for identifying the slave   | UINT8     | RO    | 0x04 (4 <sub>dec</sub> )                |
| 1018:01     | Vendor ID     | Vendor ID of the EtherCAT slave   | UINT32    | RO    | 0x00000002 (2 <sub>dec</sub> )          |
| 1018:02     | Product code  | Product code of the EtherCAT slave  | UINT32    | RO    | 0x177222852 (393357394 <sub>dec</sub> ) |
| 1018:03     | Revision      | Revision number of the EtherCAT slave: <ul style="list-style-type: none"> <li>• The Low Word (bit 0-15) indicates the special terminal number.</li> <li>• The High Word (bit 16-31) refers to the device description.</li> </ul>  | UINT32    | RO    | 0x00000000 (0 <sub>dec</sub> )          |
| 1018:04     | Serial number | Serial number of the EtherCAT slave: <ul style="list-style-type: none"> <li>• Low Word                             <ul style="list-style-type: none"> <li>◦ The Low Byte (bit 0-7) of the Low Word contains the year of production.</li> <li>◦ The High Byte (bit 8-15) of the Low Word contains the week of production.</li> </ul> </li> <li>• The High Word (bit 16-31) is 0</li> </ul> | UINT32    | RO    | 0x00000000 (0 <sub>dec</sub> )          |

**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 <sub>dec</sub> )       |
| 10F0:01     | Checksum                  | Checksum across all backup entries of the EtherCAT slave          | UINT32    | RO    | 0x00000000 (0 <sub>dec</sub> ) |

## Index 1600 COM RxPDO-Map Outputs Ch.1

| Index (hex) | Name                       | Meaning  | Data type | Flags | Default                   |
|-------------|----------------------------|--|-----------|-------|---------------------------|
| 1600:0      | COM RxPDO-Map Outputs Ch.1 | PDO Mapping RxPDO 1  | UINT8     | RO    | 0x1C (28 <sub>dec</sub> ) |
| 1600:01     | SubIndex 001               | 1. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1), entry 0x01 (Transmit request)) | UINT32    | RO    | 0x7000:01, 1              |
| 1600:02     | SubIndex 002               | 2. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1), entry 0x02 (Receive accepted)) | UINT32    | RO    | 0x7000:02, 1              |
| 1600:03     | SubIndex 003               | 3. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1), entry 0x03 (Init request))     | UINT32    | RO    | 0x7000:03, 1              |
| 1600:04     | SubIndex 004               | 4. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1), entry 0x04 (Send continuous))  | UINT32    | RO    | 0x7000:04, 1              |
| 1600:05     | SubIndex 005               | 5. PDO Mapping entry (4 bits align)  | UINT32    | RO    | 0x0000:00, 4              |
| 1600:06     | SubIndex 006               | 6. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1), entry 0x09 (Output length))    | UINT32    | RO    | 0x7000:09, 8              |
| 1600:07     | SubIndex 007               | 7. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x11 (Data Out 0))        | UINT32    | RO    | 0x7000:11, 8              |
| 1600:08     | SubIndex 008               | 8. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x12 (Data Out 1))        | UINT32    | RO    | 0x7000:12, 8              |
| 1600:09     | SubIndex 009               | 9. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x13 (Data Out 2))        | UINT32    | RO    | 0x7000:13, 8              |
| 1600:0A     | SubIndex 010               | 10. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x14 (Data Out 3))       | UINT32    | RO    | 0x7000:14, 8              |
| 1600:0B     | SubIndex 011               | 11. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x15 (Data Out 4))       | UINT32    | RO    | 0x7000:15, 8              |
| 1600:0C     | SubIndex 012               | 12. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x16 (Data Out 5))       | UINT32    | RO    | 0x7000:16, 8              |
| 1600:0D     | SubIndex 013               | 13. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x17 (Data Out 6))       | UINT32    | RO    | 0x7000:17, 8              |
| 1600:0E     | SubIndex 014               | 14. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x18 (Data Out 7))       | UINT32    | RO    | 0x7000:18, 8              |
| 1600:0F     | SubIndex 015               | 15. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x19 (Data Out 8))       | UINT32    | RO    | 0x7000:19, 8              |
| 1600:10     | SubIndex 016               | 16. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x1A (Data Out 9))       | UINT32    | RO    | 0x7000:1A, 8              |
| 1600:11     | SubIndex 017               | 17. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x1B (Data Out 10))      | UINT32    | RO    | 0x7000:1B, 8              |
| 1600:12     | SubIndex 018               | 18. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x1C (Data Out 11))      | UINT32    | RO    | 0x7000:1C, 8              |
| 1600:13     | SubIndex 019               | 19. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x1D (Data Out 12))      | UINT32    | RO    | 0x7000:1D, 8              |
| 1600:14     | SubIndex 020               | 20. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x1E (Data Out 13))      | UINT32    | RO    | 0x7000:1E, 8              |
| 1600:15     | SubIndex 021               | 21. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x1F (Data Out 14))      | UINT32    | RO    | 0x7000:1F, 8              |
| 1600:16     | SubIndex 022               | 22. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x20 (Data Out 15))      | UINT32    | RO    | 0x7000:20, 8              |
| 1600:17     | SubIndex 023               | 23. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x21 (Data Out 16))      | UINT32    | RO    | 0x7000:21, 8              |
| 1600:18     | SubIndex 024               | 24. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x22 (Data Out 17))      | UINT32    | RO    | 0x7000:22, 8              |
| 1600:19     | SubIndex 025               | 25. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x23 (Data Out 18))      | UINT32    | RO    | 0x7000:23, 8              |
| 1600:1A     | SubIndex 026               | 26. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x24 (Data Out 19))      | UINT32    | RO    | 0x7000:24, 8              |
| 1600:1B     | SubIndex 027               | 27. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x25 (Data Out 20))      | UINT32    | RO    | 0x7000:25, 8              |
| 1600:1C     | SubIndex 028               | 28. PDO Mapping entry (object 0x7000 (COM Outputs Ch.1) entry 0x26 (Data Out 21))      | UINT32    | RO    | 0x7000:26, 8              |

**Index 1601 COM RxPDO-Map Ext. Outputs Ch.1**

| Index (hex) | Name                            | Meaning  | Data type | Flags | Default                   |
|-------------|---------------------------------|--|-----------|-------|---------------------------|
| 1601:0      | COM RxPDO-Map Ext. Outputs Ch.1 | PDO Mapping RxPDO 2  | UINT8     | RO    | 0x03 (03 <sub>dec</sub> ) |
| 1601:01     | SubIndex 001                    | 1. PDO Mapping entry (object 0x7001 (COM Ext. Outputs Ch.1), entry 0x01 (Tx))  | UINT32    | RO    | 0x7001:01, 1              |
| 1601:02     | SubIndex 002                    | 2. PDO Mapping entry (object 0x7001 (COM Ext. Outputs Ch.1), entry 0x02 (RTS)) | UINT32    | RO    | 0x7001:02, 1              |
| 1601:03     | SubIndex 003                    | 3. PDO Mapping entry (14 bits align)   | UINT32    | RO    | 0x0000:00, 14             |

## Index 1602 COM RxPDO-Map Outputs Ch.2

| Index (hex) | Name                       | Meaning  | Data type | Flags | Default                   |
|-------------|----------------------------|--|-----------|-------|---------------------------|
| 1602:0      | COM RxPDO-Map Outputs Ch.2 | PDO Mapping RxPDO 3  | UINT8     | RO    | 0x1C (28 <sub>dec</sub> ) |
| 1602:01     | SubIndex 001               | 1. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2), entry 0x01 (Transmit request)) | UINT32    | RO    | 0x7010:01, 1              |
| 1602:02     | SubIndex 002               | 2. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2), entry 0x02 (Receive accepted)) | UINT32    | RO    | 0x7010:02, 1              |
| 1602:03     | SubIndex 003               | 3. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2), entry 0x03 (Init request))     | UINT32    | RO    | 0x7010:03, 1              |
| 1602:04     | SubIndex 004               | 4. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2), entry 0x04 (Send continuous))  | UINT32    | RO    | 0x7010:04, 1              |
| 1602:05     | SubIndex 005               | 5. PDO Mapping entry (4 bits align)  | UINT32    | RO    | 0x0000:00, 4              |
| 1602:06     | SubIndex 006               | 6. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2), entry 0x09 (Output length))    | UINT32    | RO    | 0x7010:09, 8              |
| 1602:07     | SubIndex 007               | 7. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x11 (Data Out 0))        | UINT32    | RO    | 0x7010:11, 8              |
| 1602:08     | SubIndex 008               | 8. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x12 (Data Out 1))        | UINT32    | RO    | 0x7010:12, 8              |
| 1602:09     | SubIndex 009               | 9. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x13 (Data Out 2))        | UINT32    | RO    | 0x7010:13, 8              |
| 1602:0A     | SubIndex 010               | 10. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x14 (Data Out 3))       | UINT32    | RO    | 0x7010:14, 8              |
| 1602:0B     | SubIndex 011               | 11. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x15 (Data Out 4))       | UINT32    | RO    | 0x7010:15, 8              |
| 1602:0C     | SubIndex 012               | 12. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x16 (Data Out 5))       | UINT32    | RO    | 0x7010:16, 8              |
| 1602:0D     | SubIndex 013               | 13. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x17 (Data Out 6))       | UINT32    | RO    | 0x7010:17, 8              |
| 1602:0E     | SubIndex 014               | 14. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x18 (Data Out 7))       | UINT32    | RO    | 0x7010:18, 8              |
| 1602:0F     | SubIndex 015               | 15. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x19 (Data Out 8))       | UINT32    | RO    | 0x7010:19, 8              |
| 1602:10     | SubIndex 016               | 16. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x1A (Data Out 9))       | UINT32    | RO    | 0x7010:1A, 8              |
| 1602:11     | SubIndex 017               | 17. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x1B (Data Out 10))      | UINT32    | RO    | 0x7010:1B, 8              |
| 1602:12     | SubIndex 018               | 18. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x1C (Data Out 11))      | UINT32    | RO    | 0x7010:1C, 8              |
| 1602:13     | SubIndex 019               | 19. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x1D (Data Out 12))      | UINT32    | RO    | 0x7010:1D, 8              |
| 1602:14     | SubIndex 020               | 20. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x1E (Data Out 13))      | UINT32    | RO    | 0x7010:1E, 8              |
| 1602:15     | SubIndex 021               | 21. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x1F (Data Out 14))      | UINT32    | RO    | 0x7010:1F, 8              |
| 1602:16     | SubIndex 022               | 22. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x20 (Data Out 15))      | UINT32    | RO    | 0x7010:20, 8              |
| 1602:17     | SubIndex 023               | 23. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x21 (Data Out 16))      | UINT32    | RO    | 0x7010:21, 8              |
| 1602:18     | SubIndex 024               | 24. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x22 (Data Out 17))      | UINT32    | RO    | 0x7010:22, 8              |
| 1602:19     | SubIndex 025               | 25. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x23 (Data Out 18))      | UINT32    | RO    | 0x7010:23, 8              |
| 1602:1A     | SubIndex 026               | 26. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x24 (Data Out 19))      | UINT32    | RO    | 0x7010:24, 8              |
| 1602:1B     | SubIndex 027               | 27. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x25 (Data Out 20))      | UINT32    | RO    | 0x7010:25, 8              |
| 1602:1C     | SubIndex 028               | 28. PDO Mapping entry (object 0x7010 (COM Outputs Ch.2) entry 0x26 (Data Out 21))      | UINT32    | RO    | 0x7010:26, 8              |



**Index 1603 COM RxPDO-Map Ext. Outputs Ch.2**

| Index (hex) | Name                            | Meaning  | Data type | Flags | Default                   |
|-------------|---------------------------------|--|-----------|-------|---------------------------|
| 1603:0      | COM RxPDO-Map Ext. Outputs Ch.2 | PDO Mapping RxPDO 4  | UINT8     | RO    | 0x03 (03 <sub>dec</sub> ) |
| 1603:01     | SubIndex 001                    | 1. PDO Mapping entry (object 0x7011 (COM Ext. Outputs Ch.2), entry 0x01 (Tx))  | UINT32    | RO    | 0x7011:01, 1              |
| 1603:02     | SubIndex 002                    | 2. PDO Mapping entry (object 0x7011 (COM Ext. Outputs Ch.2), entry 0x02 (RTS)) | UINT32    | RO    | 0x7011:02, 1              |
| 1603:03     | SubIndex 003                    | 3. PDO Mapping entry (14 bits align)   | UINT32    | RO    | 0x0000:00, 14             |

## Index 1A00 COM TxPDO-Map Inputs Ch.1

| Index (hex) | Name                      | Meaning  | Data type | Flags | Default                   |
|-------------|---------------------------|--|-----------|-------|---------------------------|
| 1A00:0      | COM TxPDO-Map Inputs Ch.1 | PDO Mapping TxPDO 1  | UINT8     | RO    | 0x1F (31 <sub>dec</sub> ) |
| 1A00:01     | SubIndex 001              | 1. PDO Mapping entry (object 0x6000 (COM Inputs Ch.1), entry 0x01 (Transmit accepted)) | UINT32    | RO    | 06000:01, 1               |
| 1A00:02     | SubIndex 002              | 2. PDO Mapping entry (object 0x6000 (COM Inputs Ch.1), entry 0x02 (Receive request))   | UINT32    | RO    | 06000:02, 1               |
| 1A00:03     | SubIndex 003              | 3. PDO Mapping entry (object 0x6000 (COM Inputs Ch.1), entry 0x03 (Init accepted))     | UINT32    | RO    | 06000:03, 1               |
| 1A00:04     | SubIndex 004              | 4. PDO Mapping entry (object 0x6000 (COM Inputs Ch.1), entry 0x04 (Buffer full))       | UINT32    | RO    | 06000:04, 1               |
| 1A00:05     | SubIndex 005              | 5. PDO Mapping entry (object 0x6000 (COM Inputs Ch.1), entry 0x05 (Parity error))      | UINT32    | RO    | 06000:05, 1               |
| 1A00:06     | SubIndex 006              | 6. PDO Mapping entry (object 0x6000 (COM Inputs Ch.1), entry 0x06 (Framing error))     | UINT32    | RO    | 06000:06, 1               |
| 1A00:07     | SubIndex 007              | 7. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x07 (Overrun error))   | UINT32    | RO    | 06000:07, 1               |
| 1A00:08     | SubIndex 008              | 8. PDO Mapping entry (1 bits align)  | UINT32    | RO    | 0x0000:00, 1              |
| 1A00:09     | SubIndex 009              | 9. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x09 (Input length))    | UINT32    | RO    | 06000:09, 8               |
| 1A00:0A     | SubIndex 010              | 10. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x11 (Data In 0))      | UINT32    | RO    | 06000:11, 8               |
| 1A00:0B     | SubIndex 011              | 11. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x12 (Data In 1))      | UINT32    | RO    | 06000:12, 8               |
| 1A00:0C     | SubIndex 012              | 12. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x13 (Data In 2))      | UINT32    | RO    | 06000:13, 8               |
| 1A00:0D     | SubIndex 013              | 13. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x14 (Data In 3))      | UINT32    | RO    | 06000:14, 8               |
| 1A00:0E     | SubIndex 014              | 14. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x15 (Data In 4))      | UINT32    | RO    | 06000:15, 8               |
| 1A00:0F     | SubIndex 015              | 15. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x16 (Data In 5))      | UINT32    | RO    | 06000:16, 8               |
| 1A00:10     | SubIndex 016              | 16. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x17 (Data In 6))      | UINT32    | RO    | 06000:17, 8               |
| 1A00:11     | SubIndex 017              | 17. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x18 (Data In 7))      | UINT32    | RO    | 06000:18, 8               |
| 1A00:12     | SubIndex 018              | 18. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x19 (Data In 8))      | UINT32    | RO    | 06000:19, 8               |
| 1A00:13     | SubIndex 019              | 19. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x1A (Data In 9))      | UINT32    | RO    | 06000:1A, 8               |
| 1A00:14     | SubIndex 020              | 20. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x1B (Data In 10))     | UINT32    | RO    | 06000:1B, 8               |
| 1A00:15     | SubIndex 021              | 21. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x1C (Data In 11))     | UINT32    | RO    | 06000:1C, 8               |
| 1A00:16     | SubIndex 022              | 22. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x1C (Data In 12))     | UINT32    | RO    | 06000:1D, 8               |
| 1A00:17     | SubIndex 023              | 23. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x1E (Data In 13))     | UINT32    | RO    | 06000:1E, 8               |
| 1A00:18     | SubIndex 024              | 24. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x1F (Data In 14))     | UINT32    | RO    | 06000:1F, 8               |
| 1A00:19     | SubIndex 025              | 25. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x20 (Data In 15))     | UINT32    | RO    | 06000:20, 8               |
| 1A00:1A     | SubIndex 026              | 26. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x21 (Data In 16))     | UINT32    | RO    | 06000:21, 8               |
| 1A00:1B     | SubIndex 027              | 27. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x22 (Data In 17))     | UINT32    | RO    | 06000:22, 8               |
| 1A00:1C     | SubIndex 028              | 28. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x23 (Data In 18))     | UINT32    | RO    | 06000:23, 8               |
| 1A00:1D     | SubIndex 029              | 29. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x24 (Data In 19))     | UINT32    | RO    | 06000:24, 8               |
| 1A00:1E     | SubIndex 030              | 30. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x25 (Data In 20))     | UINT32    | RO    | 06000:25, 8               |
| 1A00:1F     | SubIndex 031              | 31. PDO Mapping entry (object 0x0x6000 (COM Inputs Ch.1), entry 0x26 (Data In 21))     | UINT32    | RO    | 06000:26, 8               |

**Index 1A01 COM TxPDO-Map Ext. Inputs Ch.1**

| Index (hex) | Name                           | Meaning   | Data type | Flags | Default                  |
|-------------|--------------------------------|---|-----------|-------|--------------------------|
| 1A01:0      | COM TxPDO-Map Ext. Inputs Ch.1 | PDO Mapping TxPDO 2   | UINT8     | RO    | 0x03 (3 <sub>dec</sub> ) |
| 1A01:01     | SubIndex 001                   | 1. PDO Mapping entry (object 0x6001 (COM Ext. Inputs Ch.1), entry 0x01 (Rx))  | UINT32    | RO    | 0x6001:01, 1             |
| 1A01:02     | SubIndex 002                   | 2. PDO Mapping entry (object 0x6001 (COM Ext. Inputs Ch.1), entry 0x02 (CTS)) | UINT32    | RO    | 0x6001:02, 1             |
| 1A01:03     | SubIndex 003                   | 3. PDO Mapping entry (14 bits align)  | UINT32    | RO    | 0x0000:00, 14            |

## Index 1A02 COM TxPDO-Map Inputs Ch.2

| Index (hex) | Name                      | Meaning  | Data type | Flags | Default                   |
|-------------|---------------------------|--|-----------|-------|---------------------------|
| 1A02:0      | COM TxPDO-Map Inputs Ch.2 | PDO Mapping TxPDO 3  | UINT8     | RO    | 0x1F (31 <sub>dec</sub> ) |
| 1A02:01     | SubIndex 001              | 1. PDO Mapping entry (object 0x6010 (COM Inputs Ch.2), entry 0x01 (Transmit accepted)) | UINT32    | RO    | 06010:01, 1               |
| 1A02:02     | SubIndex 002              | 2. PDO Mapping entry (object 0x6010 (COM Inputs Ch.2), entry 0x02 (Receive request))   | UINT32    | RO    | 06010:02, 1               |
| 1A02:03     | SubIndex 003              | 3. PDO Mapping entry (object 0x6010 (COM Inputs Ch.2), entry 0x03 (Init accepted))     | UINT32    | RO    | 06010:03, 1               |
| 1A02:04     | SubIndex 004              | 4. PDO Mapping entry (object 0x6010 (COM Inputs Ch.2), entry 0x04 (Buffer full))       | UINT32    | RO    | 06010:04, 1               |
| 1A02:05     | SubIndex 005              | 5. PDO Mapping entry (object 0x6010 (COM Inputs Ch.2), entry 0x05 (Parity error))      | UINT32    | RO    | 06010:05, 1               |
| 1A02:06     | SubIndex 006              | 6. PDO Mapping entry (object 0x6010 (COM Inputs Ch.2), entry 0x06 (Framing error))     | UINT32    | RO    | 06010:06, 1               |
| 1A02:07     | SubIndex 007              | 7. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x07 (Overrun error))   | UINT32    | RO    | 06010:07, 1               |
| 1A02:08     | SubIndex 008              | 8. PDO Mapping entry (1 bits align)  | UINT32    | RO    | 0x0000:00, 1              |
| 1A02:09     | SubIndex 009              | 9. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x09 (Input length))    | UINT32    | RO    | 06010:09, 8               |
| 1A02:0A     | SubIndex 010              | 10. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x11 (Data In 0))      | UINT32    | RO    | 06010:11, 8               |
| 1A02:0B     | SubIndex 011              | 11. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x12 (Data In 1))      | UINT32    | RO    | 06010:12, 8               |
| 1A02:0C     | SubIndex 012              | 12. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x13 (Data In 2))      | UINT32    | RO    | 06010:13, 8               |
| 1A02:0D     | SubIndex 013              | 13. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x14 (Data In 3))      | UINT32    | RO    | 06010:14, 8               |
| 1A02:0E     | SubIndex 014              | 14. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x15 (Data In 4))      | UINT32    | RO    | 06010:15, 8               |
| 1A02:0F     | SubIndex 015              | 15. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x16 (Data In 5))      | UINT32    | RO    | 06010:16, 8               |
| 1A02:10     | SubIndex 016              | 16. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x17 (Data In 6))      | UINT32    | RO    | 06010:17, 8               |
| 1A02:11     | SubIndex 017              | 17. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x18 (Data In 7))      | UINT32    | RO    | 06010:18, 8               |
| 1A02:12     | SubIndex 018              | 18. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x19 (Data In 8))      | UINT32    | RO    | 06010:19, 8               |
| 1A02:13     | SubIndex 019              | 19. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x1A (Data In 9))      | UINT32    | RO    | 06010:1A, 8               |
| 1A02:14     | SubIndex 020              | 20. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x1B (Data In 10))     | UINT32    | RO    | 06010:1B, 8               |
| 1A02:15     | SubIndex 021              | 21. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x1C (Data In 11))     | UINT32    | RO    | 06010:1C, 8               |
| 1A02:16     | SubIndex 022              | 22. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x1D (Data In 12))     | UINT32    | RO    | 06010:1D, 8               |
| 1A02:17     | SubIndex 023              | 23. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x1E (Data In 13))     | UINT32    | RO    | 06010:1E, 8               |
| 1A02:18     | SubIndex 024              | 24. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x1F (Data In 14))     | UINT32    | RO    | 06010:1F, 8               |
| 1A02:19     | SubIndex 025              | 25. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x20 (Data In 15))     | UINT32    | RO    | 06010:20, 8               |
| 1A02:1A     | SubIndex 026              | 26. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x21 (Data In 16))     | UINT32    | RO    | 06010:21, 8               |
| 1A02:1B     | SubIndex 027              | 27. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x22 (Data In 17))     | UINT32    | RO    | 06010:22, 8               |
| 1A02:1C     | SubIndex 028              | 28. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x23 (Data In 18))     | UINT32    | RO    | 06010:23, 8               |
| 1A02:1D     | SubIndex 029              | 29. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x24 (Data In 19))     | UINT32    | RO    | 06010:24, 8               |
| 1A02:1E     | SubIndex 030              | 30. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x25 (Data In 20))     | UINT32    | RO    | 06010:25, 8               |
| 1A02:1F     | SubIndex 031              | 31. PDO Mapping entry (object 0x0x6010 (COM Inputs Ch.2), entry 0x26 (Data In 21))     | UINT32    | RO    | 06010:26, 8               |

**Index 1A03 COM TxPDO-Map Ext. Inputs Ch.2**

| Index (hex) | Name                           | Meaning   | Data type | Flags | Default                  |
|-------------|--------------------------------|---|-----------|-------|--------------------------|
| 1A03:0      | COM TxPDO-Map Ext. Inputs Ch.2 | PDO Mapping TxPDO 2   | UINT8     | RO    | 0x03 (3 <sub>dec</sub> ) |
| 1A03:01     | SubIndex 001                   | 1. PDO Mapping entry (object 0x6011 (COM Ext. Inputs Ch.2), entry 0x01 (Rx))  | UINT32    | RO    | 0x6011:01, 1             |
| 1A03:02     | SubIndex 002                   | 2. PDO Mapping entry (object 0x6011 (COM Ext. Inputs Ch.2), entry 0x02 (CTS)) | UINT32    | RO    | 0x6011:02, 1             |
| 1A03:03     | SubIndex 003                   | 3. PDO Mapping entry (14 bits align)  | UINT32    | RO    | 0x0000:00, 14            |

**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 <sub>dec</sub> ) |
| 1C00:01     | SubIndex 001      | Sync-Manager Type Channel 1: Mailbox Write                | UINT8     | RO    | 0x01 (1 <sub>dec</sub> ) |
| 1C00:02     | SubIndex 002      | Sync-Manager Type Channel 2: Mailbox Read                 | UINT8     | RO    | 0x02 (2 <sub>dec</sub> ) |
| 1C00:03     | SubIndex 003      | Sync-Manager Type Channel 3: Process Data Write (Outputs) | UINT8     | RO    | 0x03 (3 <sub>dec</sub> ) |
| 1C00:04     | SubIndex 004      | Sync-Manager Type Channel 4: Process Data Read (Inputs)   | UINT8     | RO    | 0x04 (4 <sub>dec</sub> ) |

**Index 1C12 RxPDO assign**

| Index (hex) | Name         | Meaning            | Data type | Flags | Default                  |
|-------------|--------------|--------------------|-----------|-------|--------------------------|
| 1C12:0      | RxPDO assign | PDO Assign Outputs | UINT8     | RW    | 0x00 (0 <sub>dec</sub> ) |

**Index 1C13 TxPDO assign**

| Index (hex) | Name         | Meaning           | Data type | Flags | Default                  |
|-------------|--------------|-------------------|-----------|-------|--------------------------|
| 1C13:0      | TxPDO assign | PDO Assign Inputs | UINT8     | RO    | 0x00 (0 <sub>dec</sub> ) |

## Index 1C32 SM output parameter

| Index (hex) | Name                    | Meaning  | Data type | Flags | Default                             |
|-------------|-------------------------|--|-----------|-------|-------------------------------------|
| 1C32:0      | SM output parameter     | Synchronization parameters for the inputs  | UINT8     | RO    | 0x20 (32 <sub>dec</sub> )           |
| 1C32:01     | Sync mode               | Current synchronization mode: <ul style="list-style-type: none"> <li>0: Free Run</li> <li>1: Synchron with SM 3 Event (no outputs available)</li> <li>2: DC - Synchron with SYNC0 Event</li> <li>3: DC - Synchron with SYNC1 Event</li> <li>34: Synchron with SM 2 Event (outputs available)</li> </ul>  | UINT16    | RW    | 0x0000 (0 <sub>dec</sub> )          |
| 1C32:02     | Cycle time              | Cycle time (in ns): <ul style="list-style-type: none"> <li>Free Run: Cycle time of the local timer</li> <li>Synchron with SM 2 Event: Master cycle time</li> <li>DC mode: SYNC0/SYNC1 Cycle Time</li> </ul>  | UINT32    | RW    | 0x0003D090 (250000 <sub>dec</sub> ) |
| 1C32:03     | Shift time              | Time between SYNC0 event and reading of the inputs (in ns, only DC mode)   | UINT32    | RO    | 0x00000384 (900 <sub>dec</sub> )    |
| 1C32:04     | Sync modes supported    | Supported synchronization modes: <ul style="list-style-type: none"> <li>Bit 0: free run is supported</li> <li>Bit 1: Synchron with SM 2 Event is supported (outputs available)</li> <li>Bit 1: Synchron with SM 3 Event is supported (no outputs available)</li> <li>Bit 2-3 = 01: DC mode is supported</li> <li>Bit 4-5 = 01: Input Shift through local event (outputs available)</li> <li>Bit 4-5 = 10: Input Shift with SYNC1 Event (no outputs available)</li> <li>Bit 14 = 1: dynamic times (measurement through writing of 0x1C33:08)</li> </ul> | UINT16    | RO    | 0x0001 (1 <sub>dec</sub> )          |
| 1C32:05     | Minimum cycle time      | Minimum cycle time (in ns)   | UINT32    | RO    | 0x00002710 (10000 <sub>dec</sub> )  |
| 1C32:06     | 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 <sub>dec</sub> )      |
| 1C32:07     | Minimum delay time      | Min. time between SYNC1 event and the reading of the inputs (in ns, DC mode only)  | UINT32    | RO    | 0x00000384 (900 <sub>dec</sub> )    |
| 1C32:08     | Command                 | With this entry the real required process data provision time can be measured. <ul style="list-style-type: none"> <li>0: Measurement of the local cycle time is stopped</li> <li>1: Measurement of the local cycle time is started</li> </ul> The entries 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 <sub>dec</sub> )          |
| 1C32:09     | Maximum Delay time      | Time between SYNC1 event and reading of the inputs (in ns, only DC mode)   | UINT32    | RO    | 0x00000384 (900 <sub>dec</sub> )    |
| 1C32:0B     | SM event missed counter | Number of missed SM events in OPERATIONAL (DC mode only)   | UINT16    | RO    | 0x0000 (0 <sub>dec</sub> )          |
| 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 <sub>dec</sub> )          |
| 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 <sub>dec</sub> )          |
| 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 <sub>dec</sub> )            |

**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 <sub>dec</sub> )           |
| 1C33:01     | Sync mode               | Current synchronization mode: <ul style="list-style-type: none"> <li>• 0: Free Run</li> <li>• 1: Synchron with SM 3 Event (no outputs available)</li> <li>• 2: DC - Synchron with SYNC0 Event</li> <li>• 3: DC - Synchron with SYNC1 Event</li> <li>• 34: Synchron with SM 2 Event (outputs available)</li> </ul>  | UINT16    | RW    | 0x0022 (34 <sub>dec</sub> )         |
| 1C33:02     | Cycle time              | Cycle time (in ns): <ul style="list-style-type: none"> <li>• Free Run: Cycle time of the local timer</li> <li>• Synchron with SM 2 Event: Master cycle time</li> <li>• DC mode: SYNC0/SYNC1 Cycle Time</li> </ul>  | UINT32    | RW    | 0x0003D090 (250000 <sub>dec</sub> ) |
| 1C33:03     | Shift time              | Time between SYNC0 event and reading of the inputs (in ns, only DC mode)   | UINT32    | RO    | 0x00000384 (900 <sub>dec</sub> )    |
| 1C33:04     | Sync modes supported    | Supported synchronization modes: <ul style="list-style-type: none"> <li>• Bit 0: free run is supported</li> <li>• Bit 1: Synchron with SM 2 Event is supported (outputs available)</li> <li>• Bit 1: Synchron with SM 3 Event is supported (no outputs available)</li> <li>• Bit 2-3 = 01: DC mode is supported</li> <li>• Bit 4-5 = 01: Input Shift through local event (outputs available)</li> <li>• Bit 4-5 = 10: Input Shift with SYNC1 Event (no outputs available)</li> <li>• Bit 14 = 1: dynamic times (measurement through writing of 0x1C33:08)</li> </ul> | UINT16    | RO    | 0x0001 (1 <sub>dec</sub> )          |
| 1C33:05     | Minimum cycle time      | Minimum cycle time (in ns)   | UINT32    | RO    | 0x00002710 (10000 <sub>dec</sub> )  |
| 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 <sub>dec</sub> )      |
| 1C33:07     | Minimum delay time      | Min. time between SYNC1 event and the reading of the inputs (in ns, DC mode only)  | UINT32    | RO    | 0x00000384 (900 <sub>dec</sub> )    |
| 1C33:08     | Command                 | With this entry the real required process data provision time can be measured. <ul style="list-style-type: none"> <li>• 0: Measurement of the local cycle time is stopped</li> <li>• 1: Measurement of the local cycle time is started</li> </ul> The entries 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 <sub>dec</sub> )          |
| 1C33:09     | Maximum Delay time      | Time between SYNC1 event and reading of the inputs (in ns, only DC mode)   | UINT32    | RO    | 0x00000384 (900 <sub>dec</sub> )    |
| 1C33:0B     | SM event missed counter | Number of missed SM events in OPERATIONAL (DC mode only)   | UINT16    | RO    | 0x0000 (0 <sub>dec</sub> )          |
| 1C33:0C     | 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 <sub>dec</sub> )          |
| 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 <sub>dec</sub> )          |
| 1C33:20     | Sync error              | The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)   | BOOLEAN   | RO    | 0x00 (0 <sub>dec</sub> )            |

## 6.2.7 Command object (0xFB00)

### Index FB00 CMD command

| Index (hex) | Name        | Meaning  | Data type             | Flags | Default                   |
|-------------|-------------|--|-----------------------|-------|---------------------------|
| FB00:0      | CMD command | Max. Subindex (hex)  | UINT8                 | RO    | 0x03 (03 <sub>dec</sub> ) |
| FB00:01     | Request     | The Object command can be used to trigger various actions. No commands are available at this time.   | OCTET-STRING [16 bit] | RW    |                           |
| FB00:02     | Status      | Status of the command currently being executed<br>0: Command executed without error and without response<br>1: Command executed without error and with response<br>2: Command executed with error and without response<br>3: Command executed with error and with response<br>255: Command is being executed | UINT8                 | RO    | 0x00 (0 <sub>dec</sub> )  |
| FB00:03     | Response    | Optional return value of the command<br>Response: up to 48 bit   | OCTET-STRING [48 bit] | RO    |                           |



## 7 Appendix

### 7.1 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: [www.beckhoff.com](http://www.beckhoff.com)

You will also find further documentation for Beckhoff components there.

#### Support

The Beckhoff 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  
e-mail: [support@beckhoff.com](mailto:support@beckhoff.com)  
web: [www.beckhoff.com/support](http://www.beckhoff.com/support)

#### 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  
e-mail: [service@beckhoff.com](mailto:service@beckhoff.com)  
web: [www.beckhoff.com/service](http://www.beckhoff.com/service)

#### Headquarters Germany

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Germany

Phone: +49 5246 963 0  
e-mail: [info@beckhoff.com](mailto:info@beckhoff.com)  
web: [www.beckhoff.com](http://www.beckhoff.com)

## **Trademark statements**

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.

## **Third-party trademark statements**

DeviceNet and EtherNet/IP are trademarks of ODVA, Inc.

Microsoft, Microsoft Azure, Microsoft Edge, PowerShell, Visual Studio, Windows and Xbox are trademarks of the Microsoft group of companies.

More Information:  
[www.beckhoff.com/EJ6002](http://www.beckhoff.com/EJ6002)

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Germany  
Phone: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)

