TECHDOCS

Manual

WAGO-I/O-SYSTEM 750

750-354/000-001

EtherCAT® Fieldbus Coupler, ID Switch

100 Mbit/s; digital and analog signals

Version 1.2.1

The following applies for EtherCAT® and TwinCAT®: EtherCAT® and TwinCAT® are registered trademarks of and licensed by Beckhoff Automation GmbH, Germany.

The following applies for CANopen®: CANopen® is a registered community trademark of CAN in Automation e. V.

WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

Phone: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-Mail: info@wago.com

Web: www.wago.com

Technical Support

Phone: +49 (0) 571/8 87 – 4 45 55
Fax: +49 (0) 571/8 87 – 84 45 55

E-Mail: support@wago.com

Every conceivable measure has been taken to ensure the accuracy and completeness of this documentation. However, as errors can never be fully excluded, we always appreciate any information or suggestions for improving the documentation.

E-Mail: documentation@wago.com

We wish to point out that the software and hardware terms as well as the trademarks of companies used and/or mentioned in the present manual are generally protected by trademark or patent.

WAGO is a registered trademark of WAGO Verwaltungsgesellschaft mbH.

Table of Contents

1 Notes about this Documentation ... 8
1.1 Validity of this Documentation ... 8
1.2 Copyright ... 8
1.3 Symbols ... 9
1.4 Number Notation ... 11
1.5 Font Conventions ... 11
2 Important Notes ... 12
2.1 Legal Bases ... 12
2.1.1 Subject to Changes ... 12
2.1.2 Personnel Qualifications ... 12
2.1.3 Use of the 750 Series in Compliance with Underlying Provisions ... 12
2.1.4 Technical Condition of Specified Devices ... 13
2.1.4.1 Disposal ... 13
2.1.4.1.1 Electrical and Electronic Equipment ... 13
2.1.4.1.2 Packaging ... 14
2.2 Safety Advice (Precautions) ... 15
2.3 Special Use Conditions for ETHERNET Devices ... 17
3 System Description ... 18
3.1 Manufacturing Number ... 19
3.2 Update ... 20
3.3 Storage, Assembly and Transport ... 21
3.4 Assembly Guidelines/Standards ... 22
3.5 Power Supply ... 23
3.5.1 Overcurrent Protection ... 23
3.5.2 Isolation ... 23
3.5.3 System Supply ... 24
3.5.3.1 Connection ... 24
3.5.3.2 Dimensioning ... 25
3.5.4 Field Supply ... 28
3.5.4.1 Connection ... 28
3.5.4.2 Fusing via Power Supply Module ... 30
3.5.4.3 Fusing External ... 32
3.5.5 Supply Example ... 34
3.5.6 Power Supply Unit ... 36
3.6 Grounding ... 37
3.6.1 Grounding the DIN Rail ... 37
3.6.1.1 Framework Assembly ... 37
3.6.1.2 Insulated Assembly ... 37
3.6.2 Grounding Function ... 38
3.7 Shielding ... 39
3.7.1 General ... 39
3.7.2 Bus Cables ... 39
3.7.3 Shielded Signal Lines ... 40
3.7.4 WAGO Shield Connecting System ... 40

4 Device Description 41
4.1 View 42
4.2 Connectors 44
4.2.1 Device Supply 44
4.2.2 Fieldbus Connection 44
4.3 Display Elements 46
4.4 Operating Elements 47
4.4.1 Service Interface 47
4.4.2 ID Selection Switch 48
4.5 Technical Data 49
4.5.1 Device Data 49
4.5.2 Safe Electrical Isolation 49
4.5.3 Supply 49
4.5.4 Communication 50
4.5.5 Accessories 50
4.5.6 Connection Type 50
4.5.7 Climatic Environmental Conditions 51
4.5.8 Mechanical Strength 51
4.6 Approvals 52
4.7 Standards and Guidelines 53

5 Mounting 54
5.1 Installation Position 54
5.2 Overall Configuration 54
5.3 Mounting onto Carrier Rail 56
5.3.1 Carrier Rail Properties 56
5.3.2 WAGO DIN Rails 57
5.4 Spacing 57
5.5 Mounting Sequence 58
5.6 Inserting and Removing Devices 59
5.6.1 Inserting the Fieldbus Coupler/Controller 60
5.6.2 Removing the Fieldbus Coupler/Controller 60
5.6.3 Inserting the I/O Module 61
5.6.4 Removing the I/O Module 62

6 Connect Devices 63
6.1 Data Contacts/Local Bus 63
6.2 Power Contacts/Field Supply 64
6.3 Connecting a Conductor to the CAGE CLAMP® 65

7 Function Description 66
7.1 Starting Up the Fieldbus Coupler 66
7.2 Start Behavior 67
7.3 Process Data Architecture 68
7.3.1 General Mechanisms for EtherCAT® to Map Process Data 69
7.3.1.1 Object 0x6000 72
7.3.1.2 Object 0x7000 73
7.3.1.3 Object 0xF100 75
7.3.1.4 Object 0xF200 75
7.3.1.5 RxPDO and TxPDO Objects 76
7.3.2 Alternative PDO Index Assignment 77

7.3.2.1 Example According to the Modular Device Profile 78
7.3.2.2 Example According to Linear PDO Indexing 87
7.4 Object Dictionary 96
7.4.1 Structure "Module Ident" 96
7.4.2 Object 0x10F3 (Diagnostic History) 97
7.4.3 Objects 0x1C32 and 0x1C33 102
7.4.4 Objects 0x2000 … 0x2009 103
7.4.5 Objects 0x2010 and 0x2011 105
7.4.7 Object 0x2100 107
7.4.8 Object 0x2134 107
7.4.9 Objects 0x9000 ... 0x93F0 108
7.4.10 Objects 0xF030 and 0xF050 108
7.5 Data Exchange 109
7.5.1 Arrangement of the Process Data in the PAA/PAE 110
7.5.2 Determining the Process Data Arrangement Using the EtherCAT® Master 111
7.5.3 Controlling the Process Images 113
7.5.3.1 Fieldbus Coupler Control Word 113
7.5.3.2 Fieldbus Coupler Status Word 114
7.5.3.3 Behaviour of the Fieldbus Coupler During a Cycle Time Overrun 114
7.6 Hot Connect 116

8 Commissioning 118

8.1 Connecting Client PC and Fieldbus Nodes 118
8.2 Load XML-File in the Master 118
8.3 Testing the Function of the EtherCAT® Fieldbus Node 119
8.4 Testing I/O Modules using WAGO-I/O-*CHECK* 120

9 Diagnostics 121

9.1 Fieldbus Coupler Diagnostics 121
9.1.1 LED Signaling 121
9.1.2 Evaluating Fieldbus Status 121
9.1.3 Evaluating Node Status – I/O LED (Blink Code Table) 123
9.2 I/O Module Diagnostics 130
9.2.1 Diagnostic Control Word 130
9.2.2 Diagnostic Status Word 131
9.3 Behavior of the Fieldbus Coupler during Interruption of Operations 133
9.3.1 Loss of Power 133
9.3.2 Loss of Fieldbus 133
9.3.3 Local Bus Error 133
9.4 Behavior during Other Operating Errors 135
9.4.1 Configuration Error in a Sync Manager 135
9.4.2 Interruption of the Cyclical Process Data Exchange 138
9.4.3 Errors During CoE Accesses 139

10 Fieldbus Communication 140

10.1 EtherCAT® 140
10.1.1 General 140
10.1.2 Network Structure 141
10.1.2.1 Transmission Medium 141

10.1.2.2 Network Topology 141
10.1.2.3 Couplers 142
10.1.3 Network Communication 142
10.1.3.1 Communication Principle 142
10.1.3.2 Addressing 142
10.1.3.3 Configuration 143
10.1.3.4 EtherCAT® State Machine (ESM) 144
10.1.3.5 Synchronization Using Distributed Clocks 147
10.1.3.6 Performance 148
10.1.3.7 Diagnostics 148
10.1.4 EtherCAT® Interfaces 149
10.1.4.1 Slave Information Interface (SII) 149
10.1.4.2 CoE Interface (CAN Application Layer Over EtherCAT®) 149

11 I/O Modules 153

11.1 Overview 153
11.2 Process Data Architecture for EtherCAT® 154
11.2.1 Digital Input Modules 155
11.2.1.1 1 Channel Digital Input Module with Diagnostics 155
11.2.1.2 2 Channel Digital Input Modules 155
11.2.1.3 2 Channel Digital Input Module with Diagnostics 155
11.2.1.4 2 Channel Digital Input Module with Diagnostics and Output Process Data 156
11.2.1.5 4 Channel Digital Input Modules 157
11.2.1.6 8 Channel Digital Input Modules 157
11.2.1.7 16 Channel Digital Input Modules 157
11.2.2 Digital Output Modules 158
11.2.2.1 1 Channel Digital Output Module with Input Process Data 158
11.2.2.2 2 Channel Digital Output Modules 158
11.2.2.3 2 Channel Digital Input Modules with Diagnostics and Input Process Data 159
11.2.2.4 4 Channel Digital Output Modules 160
11.2.2.5 4 Channel Digital Output Modules with Diagnostics and Input Process Data 160
11.2.2.6 8 Channel Digital Output Module 161
11.2.2.7 8 Channel Digital Output Modules with Diagnostics and Input Process Data 161
11.2.2.8 16 Channel Digital Output Modules 162
11.2.2.9 8 Channel Digital Input/Output Modules 163
11.2.3 Analog Input Modules 163
11.2.3.1 1 Channel Analog Input Modules 163
11.2.3.2 2 Channel Analog Input Modules 164
11.2.3.3 2 Channel Analog Input Modules HART 164
11.2.3.4 4 Channel Analog Input Modules 165
11.2.3.5 8 Channel Analog Input Modules 166
11.2.3.6 3-Phase Power Measurement Module 167
11.2.4 Analog Output Modules 169
11.2.4.1 2 Channel Analog Output Modules 169
11.2.4.2 4 Channel Analog Output Modules 169
11.2.4.3 8 Channel Analog Output Modules 170

11.2.5 Specialty Modules ... 170
11.2.5.1 Counter Modules ... 171
11.2.5.2 Pulse Width Modules ... 173
11.2.5.3 Serial Interface Modules with Alternative Data Format ... 173
11.2.5.4 Serial Interface Modules with Standard Data Format ... 174
11.2.5.5 Serial Interface Modules ... 174
11.2.5.6 Data Exchange Module ... 175
11.2.5.7 SSI Transmitter Interface Modules ... 175
11.2.5.8 Incremental Encoder Interface Modules ... 176
11.2.5.9 DC-Drive Controller ... 178
11.2.5.10 Stepper Controller ... 179
11.2.5.11 RTC Module ... 180
11.2.5.12 DALI/DSI Master Module ... 180
11.2.5.13 DALI Multi-Master Module ... 180
11.2.5.14 EnOcean Radio Receiver ... 183
11.2.5.15 MP Bus Master Module ... 183
11.2.5.16 *Bluetooth*® RF-Transceiver ... 184
11.2.5.17 Vibration Velocity/Bearing Condition Monitoring VIB I/O ... 185
11.2.5.18 AS-interface Master Module ... 185
11.2.6 System Modules ... 187
11.2.6.1 System Modules with Diagnostics ... 187
11.2.6.2 Filter Module ... 187
11.2.6.3 Binary Space Module ... 188

12 Use in Hazardous Environments ... 189

12.1 Marking Configuration Examples ... 190
12.1.1 Marking for Europe According to ATEX and IECEx ... 190
12.1.2 Marking for the United States of America (NEC) and Canada (CEC) ... 194
12.2 Installation Regulations ... 197
12.2.1 Special Notes Regarding Explosion Protection ... 197
12.2.2 Special Notes Regarding ANSI/ISA Ex ... 199

13 Appendix ... 200

13.1 EtherCAT® Object Dictionary ... 200
13.2 AL Status Codes ... 208
13.3 SDO Abort Codes ... 209
13.4 EMCY Codes ... 210

List of Figures ... 211

List of Tables ... 213

1 Notes about this Documentation

Note

Always retain this documentation!
This documentation is part of the product. Therefore, retain the documentation during the entire service life of the product. Pass on the documentation to any subsequent user. In addition, ensure that any supplement to this documentation is included, if necessary.

1.1 Validity of this Documentation

This documentation is only applicable to the “EtherCAT® Fieldbus Coupler, ID Switch” (750-354/000-001).

The product “EtherCAT® Fieldbus Coupler, ID Switch” (750-354/000-001) shall only be installed and operated according to the instructions in this manual and the system description for the WAGO-I/O-SYSTEM 750.

NOTICE

Consider power layout of the WAGO-I/O-SYSTEM 750!
In addition to these operating instructions, you will also need the system description for the WAGO-I/O-SYSTEM 750, which can be downloaded at www.wago.com. There, you can obtain important information including information on electrical isolation, system power and supply specifications.

1.2 Copyright

1.3 Symbols

DANGER

Personal Injury!
Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.

DANGER

Personal Injury Caused by Electric Current!
Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.

WARNING

Personal Injury!
Indicates a moderate-risk, potentially hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION

Personal Injury!
Indicates a low-risk, potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

NOTICE

Damage to Property!
Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.

NOTICE

Damage to Property Caused by Electrostatic Discharge (ESD)!
Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.

Note

Important Note!
Indicates a potential malfunction which, if not avoided, however, will not result in damage to property.

Information

Additional Information:
Refers to additional information which is not an integral part of this documentation (e.g., the Internet).

1.4 Number Notation

Table 1: Number Notation

Number Code	Example	Note
Decimal	100	Normal notation
Hexadecimal	0x64	C notation
Binary	'100' '0110.0100'	In quotation marks, nibble separated with dots (.)

1.5 Font Conventions

Table 2: Font Conventions

Font Type	Indicates
italic	Names of paths and data files are marked in italic-type. e.g.: *C:\Program Files\WAGO Software*
Menu	Menu items are marked in bold letters. e.g.: **Save**
>	A greater-than sign between two names means the selection of a menu item from a menu. e.g.: **File** > **New**
Input	Designation of input or optional fields are marked in bold letters, e.g.: **Start of measurement range**
"Value"	Input or selective values are marked in inverted commas. e.g.: Enter the value "4 mA" under **Start of measurement range**.
[Button]	Pushbuttons in dialog boxes are marked with bold letters in square brackets. e.g.: **[Input]**
[Key]	Keys are marked with bold letters in square brackets. e.g.: **[F5]**

2 Important Notes

This section includes an overall summary of the most important safety requirements and notes that are mentioned in each individual section. To protect your health and prevent damage to devices as well, it is imperative to read and carefully follow the safety guidelines.

2.1 Legal Bases

2.1.1 Subject to Changes

WAGO Kontakttechnik GmbH & Co. KG reserves the right to provide for any alterations or modifications. WAGO Kontakttechnik GmbH & Co. KG owns all rights arising from the granting of patents or from the legal protection of utility patents. Third-party products are always mentioned without any reference to patent rights. Thus, the existence of such rights cannot be excluded.

2.1.2 Personnel Qualifications

All sequences implemented on WAGO-I/O-SYSTEM 750 devices may only be carried out by electrical specialists with sufficient knowledge in automation. The specialists must be familiar with the current norms and guidelines for the devices and automated environments.

All changes to the coupler or controller should always be carried out by qualified personnel with sufficient skills in PLC programming.

2.1.3 Use of the 750 Series in Compliance with Underlying Provisions

Fieldbus couplers, controllers and I/O modules found in the modular WAGO-I/O-SYSTEM 750 receive digital and analog signals from sensors and transmit them to actuators or higher-level control systems. Using controllers, the signals can also be (pre-) processed.

The devices have been developed for use in an environment that meets the IP20 protection class criteria. Protection against finger injury and solid impurities up to 12.5 mm diameter is assured; protection against water damage is not ensured. Unless otherwise specified, operation of the devices in wet and dusty environments is prohibited.

Operating the WAGO-I/O-SYSTEM 750 devices in home applications without further measures is only permitted if they meet the emission limits (emissions of interference) according to EN 61000-6-3. You will find the relevant information in the section “Device Description” > “Standards and Guidelines” in the manual for the used fieldbus coupler or controller.

Appropriate housing (per 2014/34/EU) is required when operating the WAGO-I/O-SYSTEM 750 in hazardous environments. Please note that a prototype test certificate must be obtained that confirms the correct installation of the system in a housing or switch cabinet.

The implementation of safety functions such as EMERGENCY STOP or safety door monitoring must only be performed by the F-I/O modules within the modular WAGO-I/O-SYSTEM 750. Only these safe F-I/O modules ensure functional safety in accordance with the latest international standards. WAGO's interference-free output modules can be controlled by the safety function.

2.1.4 Technical Condition of Specified Devices

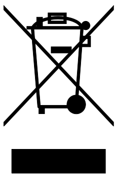
The devices to be supplied ex works are equipped with hardware and software configurations, which meet the individual application requirements. These modules contain no parts that can be serviced or repaired by the user. The following actions will result in the exclusion of liability on the part of WAGO Kontakttechnik GmbH & Co. KG:

- Repairs,
- Changes to the hardware or software that are not described in the operating instructions,
- Improper use of the components.

Further details are given in the contractual agreements. Please send your request for modified and new hardware or software configurations directly to WAGO Kontakttechnik GmbH & Co. KG.

2.1.4.1 Disposal

2.1.4.1.1 Electrical and Electronic Equipment



Electrical and electronic equipment may not be disposed of with household waste. This also applies to products without this symbol.

Electrical and electronic equipment contain materials and substances that can be harmful to the environment and health. Electrical and electronic equipment must be disposed of properly after use.
WEEE 2012/19/EU applies throughout Europe. Directives and laws may vary nationally.

Environmentally friendly disposal benefits health and protects the environment from harmful substances in electrical and electronic equipment.

- Observe national and local regulations for the disposal of electrical and electronic equipment.
- Clear any data stored on the electrical and electronic equipment.
- Remove any added battery or memory card in the electrical and electronic equipment.
- Have the electrical and electronic equipment sent to your local collection point.

Improper disposal of electrical and electronic equipment can be harmful to the environment and human health.

2.1.4.1.2 Packaging

Packaging contains materials that can be reused.
PPWD 94/62/EU and 2004/12/EU packaging guidelines apply throughout Europe. Directives and laws may vary nationally.

Environmentally friendly disposal of the packaging protects the environment and allows sustainable and efficient use of resources.

- Observe national and local regulations for the disposal of packaging.
- Dispose of packaging of all types that allows a high level of recovery, reuse and recycling.

Improper disposal of packaging can be harmful to the environment and wastes valuable resources.

2.2 Safety Advice (Precautions)

For installing and operating purposes of the relevant device to your system the following safety precautions shall be observed:

DANGER

Do not work on devices while energized!
All power sources to the device shall be switched off prior to performing any installation, repair or maintenance work.

DANGER

Install the device only in appropriate housings, cabinets or in electrical operation rooms!
The WAGO-I/O-SYSTEM 750 and its components are an open system. As such, install the system and its components exclusively in appropriate housings, cabinets or in electrical operation rooms. Allow access to such equipment and fixtures to authorized, qualified staff only by means of specific keys or tools.

DANGER

Ensure a standard connection!
To minimize any hazardous situations resulting in personal injury or to avoid failures in your system, the data and power supply lines shall be installed according to standards, with careful attention given to ensuring the correct terminal assignment. Always adhere to the EMC directives applicable to your application.

NOTICE

Replace defective or damaged devices!
Replace defective or damaged device/module (e.g., in the event of deformed contacts).

NOTICE

Protect the components against materials having seeping and insulating properties!
The components are not resistant to materials having seeping and insulating properties such as: aerosols, silicones and triglycerides (found in some hand creams). If you cannot exclude that such materials will appear in the component environment, then install the components in an enclosure being resistant to the above-mentioned materials. Clean tools and materials are imperative for handling devices/modules.

NOTICE

Clean only with permitted materials!
Clean housing and soiled contacts with propanol.

NOTICE

Do not use any contact spray!
Do not use any contact spray. The spray may impair contact area functionality in connection with contamination.

NOTICE

Do not reverse the polarity of connection lines!
Avoid reverse polarity of data and power supply lines, as this may damage the devices involved.

NOTICE

Avoid electrostatic discharge!
The devices are equipped with electronic components that may be destroyed by electrostatic discharge when touched. Please observe the safety precautions against electrostatic discharge per DIN EN 61340-5-1/-3. When handling the devices, please ensure that environmental factors (personnel, work space and packaging) are properly grounded.

2.3 Special Use Conditions for ETHERNET Devices

If not otherwise specified, ETHERNET devices are intended for use on local networks. Please note the following when using ETHERNET devices in your system:

- Do not connect control components and control networks to an open network such as the Internet or an office network. WAGO recommends putting control components and control networks behind a firewall.
- Limit physical and electronic access to all automation components to authorized personnel only.
- Change the default passwords before first use! This will reduce the risk of unauthorized access to your system.
- Regularly change the passwords used! This will reduce the risk of unauthorized access to your system.
- If remote access to control components and control networks is required, use a Virtual Private Network (VPN).
- Regularly perform threat analyses. You can check whether the measures taken meet your security requirements.
- Use “defense-in-depth” mechanisms in your system's security configuration to restrict the access to and control of individual products and networks.

3 System Description

The WAGO-I/O-SYSTEM 750 is a modular, fieldbus-independent input/output system (I/O system). The configuration described here consists of a fieldbus coupler/controller (1) and the modular I/O modules (2) for any signal shapes that form the fieldbus node together. The end module (3) completes the node and is required for correct operation of the fieldbus node.

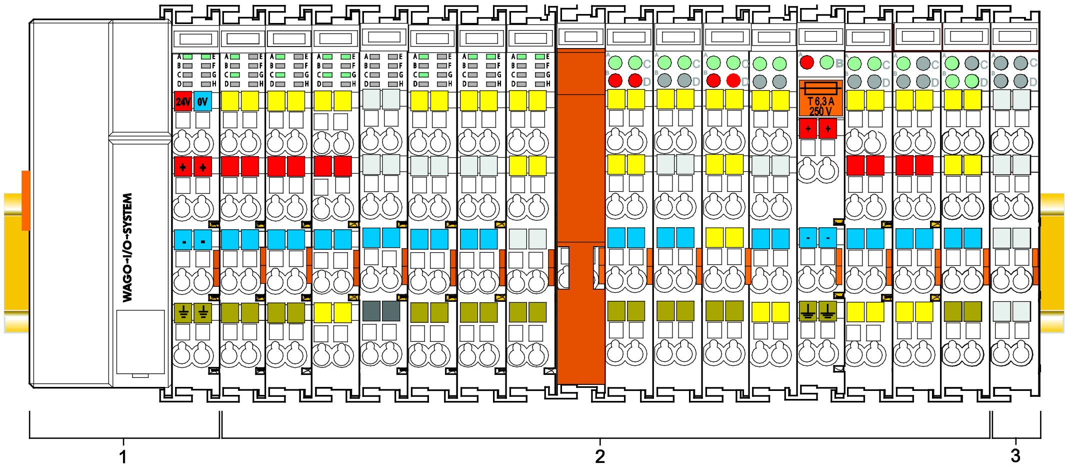


Figure 1: Fieldbus Node (Example)

Fieldbus couplers/controllers are available for different fieldbus systems.

The ECO coupler contains the fieldbus interface, electronics and a power supply for the system. The fieldbus interface forms the physical interface to the relevant fieldbus. The electronics process the data of the bus modules and make it available for the fieldbus communication.

I/O modules for diverse digital and analog I/O signals as well as special functions can be connected to the fieldbus coupler/controller. The communication between the fieldbus coupler/controller and the I/O modules is carried out via a local bus.

The components of the WAGO-I/O-SYSTEM 750 have clear termination points, light emitting diodes for status display, plug-in mini WSB tags and group marker cards for labeling.

The 1, 2 or 3 wire technology supplemented by a ground wire connection allows for direct sensor or actuator wiring.

3.1 Manufacturing Number

The serial number indicates the delivery status directly after production. This number is part of the labeling on the side of each component.

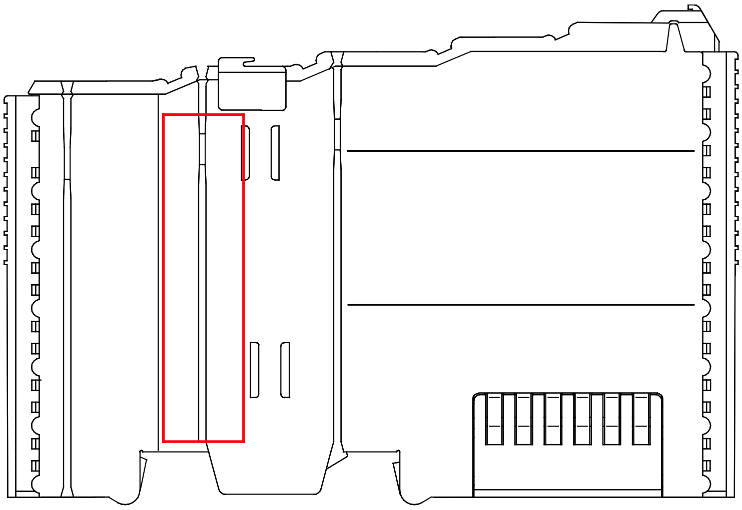


Figure 2: Marking Area for Serial Numbers

There are two serial numbers in two rows in the side marking. They are left of the release tab. The first 10 positions in the longer row of the serial numbers contain version and date identifications.

Example structure of the rows: 0114010101…

01	**14**	**01**	**01**	**01**	**(additional positions)**
WW	**YY**	**FW --**	**HW**	**FL**	**-**
Calendar week	Year	Firmware version	Hardware version	Firmware loader version	Internal information

The row order can vary depending on the production year, only the longer row is relevant. The back part of this and the shorter row contain internal administration information from the manufacturer.

In addition, the serial number is printed on the front of the fieldbus coupler/controller on the cover cap of the service interface, so that it can also be read when installed.

3.2 Update

For products that can be updated, the side inscription has a prepared matrix in which the current update data can be entered in columns.

Up to 2015, the matrix has rows to enter the “NO” work order number (or “BA” to CW 13/2004), “DS” update date, “SW” software index (optional), “HW” hardware index and “FWL” firmware loader index (optional).

NO			
DS			
SW			
HW			
FWL			

Figure 3: Update Matrix up to 2015

From 2016, the matrix has rows to enter the “FA” production or work order number and to enter the “PD” production date and “AZ” item number.

FA	XXXXXXXXXX	
PD	WWJJ	
AZ	FWHWFL	

Figure 4: Update Matrix from 2016

Table 3: Legend for Figure “Update Matrix from 2016”

	Description
FA	Production order number, 10-digit
PD	KW = calendar week YY = year
AZ	FW = firmware index HW = hardware index FL = firmware loader index

For factory updates to a head station, the current production or work order number is also printed on the cover cap of the service interface.

The original manufacturing information on the product housing remains unchanged.

3.3 Storage, Assembly and Transport

Whenever possible, the components are to be stored in their original packaging. Likewise, the original packaging provides optimal protection during transport.

When assembling or repacking the components, the contacts must not be soiled or damaged. The components must be stored and transported in appropriate containers/packaging. Thereby, the ESD information is to be regarded.

3.4 Assembly Guidelines/Standards

- DIN 60204 Electrical equipment of machines
- DIN EN 50178 Electronic equipment for use in power installations (replacement for VDE 0160)
- EN 60439 Low-voltage switchgear and controlgear assemblies

3.5 Power Supply

3.5.1 Overcurrent Protection

The system and field voltage of the WAGO-I/O-SYSTEMs 750 is supplied on the head stations and bus supply modules.
For components that work with extra low voltage, only SELV/PELV voltage sources should be used.

A single voltage source supplying multiple components must be designed according to the component with the strictest electrical safety requirements.
For components which are only allowed to be supplied by SELV voltage sources, these requirements are listed in the technical data.

Most components in the WAGO-I/O-SYSTEM 750 have no internal overcurrent protection. Therefore, appropriate overcurrent production must always be implemented externally for the power supply to these components, e.g. via fuses. The maximum permissible current is listed in the technical data of the components used.

3.5.2 Isolation

Within the fieldbus node, there are three electrically isolated potentials:

- Electrically isolated fieldbus interface via transformer
- Electronics of the fieldbus couplers/controllers and the I/O modules (local bus)
- All I/O modules have an electrical isolation between the electronics (local bus, logic) and the field electronics. Some digital and analog input modules have each channel electrically isolated, please see catalog.

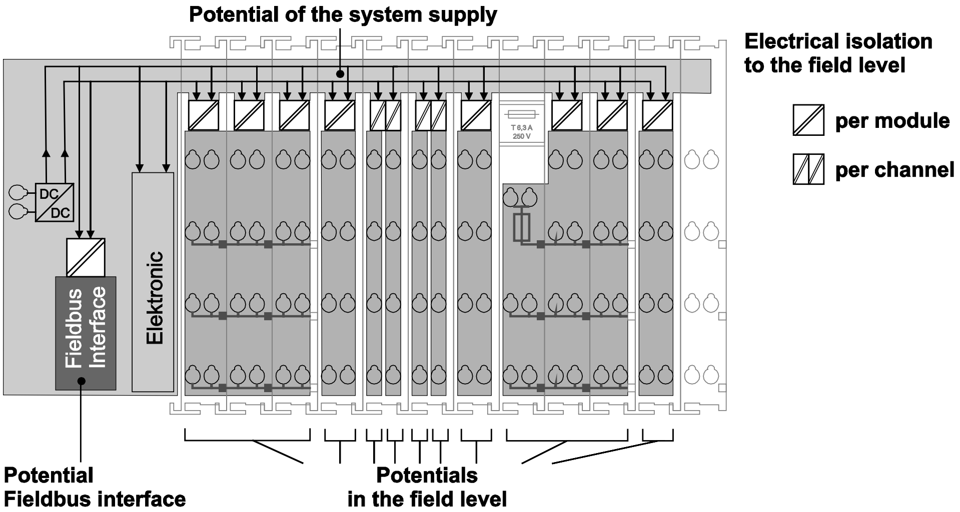


Figure 5: Isolation (Example)

3.5.3 System Supply

3.5.3.1 Connection

The WAGO-I/O-SYSTEM 750 requires a 24 V direct current system supply.
The power supply is provided via the fieldbus coupler/controller and, if necessary, in addition via internal system supply modules 750-613. The power supply is reverse voltage protected.

NOTICE

Do not use an incorrect voltage/frequency!
The use of an incorrect supply voltage or frequency can cause severe damage to the components.

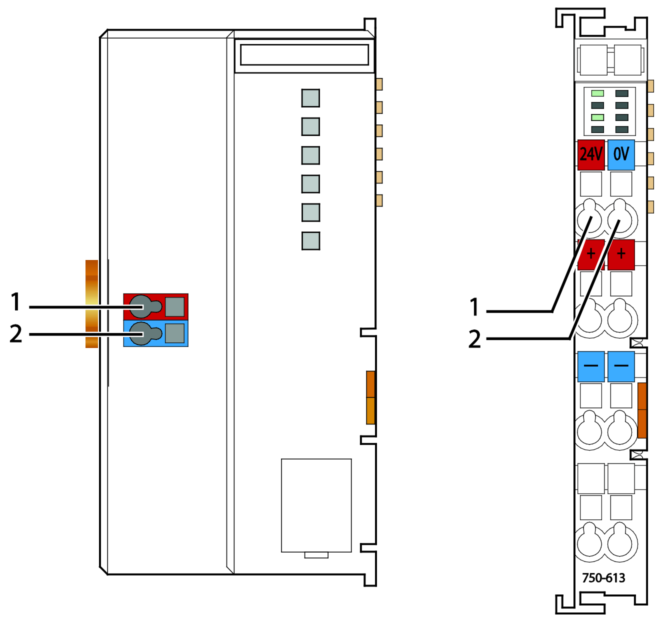


Figure 6: System Supply

Table 4: Legend for Figure "System Supply"

Pos.	Description
1	System supply 24 VDC (−15% / +20%)
2	System supply 0 V

The fed-in 24 VDC supplies all internal system components, e.g. fieldbus coupler/controller electronics, fieldbus interface and I/O modules via the local bus (5 VDC system voltage). The 5 VDC system voltage is galvanically connected to the 24 VDC supply voltage.

NOTICE

System supply only with appropriate fuse protection!
Without overcurrent protection, the electronics can be damaged.
If you implement the overcurrent protection for the system supply with a fuse, a fuse, max. 2 A, slow-acting, should be used.

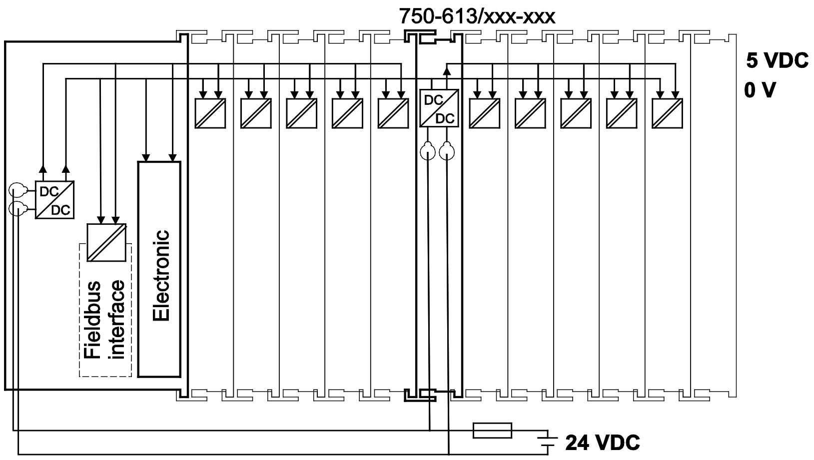


Figure 7: System Voltage (Example)

Note

Only reset the system simultaneously for all supply modules!
Reset the system by switching the system supply simultaneously at all supply modules (fieldbus coupler/controller and potential supply module with bus power supply) off and on again.

3.5.3.2 Dimensioning

Note

Recommendation
A stable power supply cannot always be assumed. Therefore, you should use regulated power supplies to ensure the quality of the supply voltage.

The supply capacity of the fieldbus coupler/controller or the internal system supply module can be taken from the technical data of the components.

Table 5: Alignment

Internal current consumption*)	Current consumption via system voltage (5 V for electronics of I/O modules and fieldbus coupler/controller).
Total current for I/O modules*)	Available current for the I/O modules. Provided by the bus power supply unit. See fieldbus coupler/controller and internal system supply module

*) See current catalog, manuals, Internet

Example:

Calculating the current consumption on an example coupler

Internal current consumption	300 mA at 5 V
Residual current for bus modules	700 mA at 5 V
Sum $I_{(5\ V)\ total}$	**1000 mA at 5 V**

The internal current consumption is indicated in the technical data for each bus terminal. In order to determine the total requirement, add together the values of all I/O modules in the node.

Note

Please note the aggregate current for I/O modules. It may be necessary to supply potential!
When the sum of the internal current consumption for the I/O modules exceeds their aggregate current, you must use a supply module with bus power supply. Install it before the position where the permissible aggregate current would be exceeded.

Example:

Calculating the total current on the example coupler

A node with an example coupler consists e. g. of the following I/O modules:
20 relay modules (750-517) and 10 digital input modules (750-405).

Internal current consumption	10 * 90 mA =	900 mA
	20 * 2 mA =	40 mA
Sum		**940 mA**

In this example, the example coupler can provide 700 mA for the I/O modules. This value is given in the associated data sheet ("Total current for I/O modules"). Consequently, an internal system supply module (750-613), e. g. in the middle of the node, should be added.

Note

Recommendation
Utilize the **smartDESIGNER** feature WAGO ProServe® software to configure fieldbus node assembly. You can test the configuration via the integrated plausibility check.

The maximum input current of the 24 V system supply is 500 mA. The exact electrical consumption ($I_{(V)}$) can be determined with the following formulas:

Fieldbus coupler or controller

$I_{(5\ V)\ total}$ = Sum of all the internal current consumption of the connected I/O modules + internal current consumption of the fieldbus coupler/controller

Internal system supply module

$I_{(5\ V)\ total}$ = Sum of all the internal current consumption of the connected I/O modules at internal system supply module

$$\text{Input current } I_{(24\ V)} = \frac{5\ V}{24\ V} \times \frac{I_{(5\ V)\ total}}{\eta}$$

η = Efficiency of the power supply at nominal load 24 V

Note

Activate all outputs when testing the current consumption!
If the electrical consumption of a power supply point for the 24 V system supply exceeds 500 mA, then the cause may be an improperly dimensioned node or a defect.
During the test, you must activate all outputs.

3.5.4 Field Supply

3.5.4.1 Connection

Sensors and actuators can be directly connected to the relevant channel of the I/O module in 1, 2, 3 or 4 conductor connection technology. The I/O module supplies power to the sensors and actuators. The input and output drivers of some I/O modules require the field side supply voltage.

The power supply modules provide field side power (24 VDC). In this case it is a passive power supply without protection equipment. Power supply modules are available for different potentials, e.g. 24 VDC, 230 VAC or others.

Power supply modules with or without fuse holder and diagnostic capability are available for the power supply of other field potentials (24 VDC, 0 … 230 VAC/DC, 120 VAC, 230 VAC). The power supply modules can also be used to set up various potential groups. The connections are connected in pairs to a power contact.

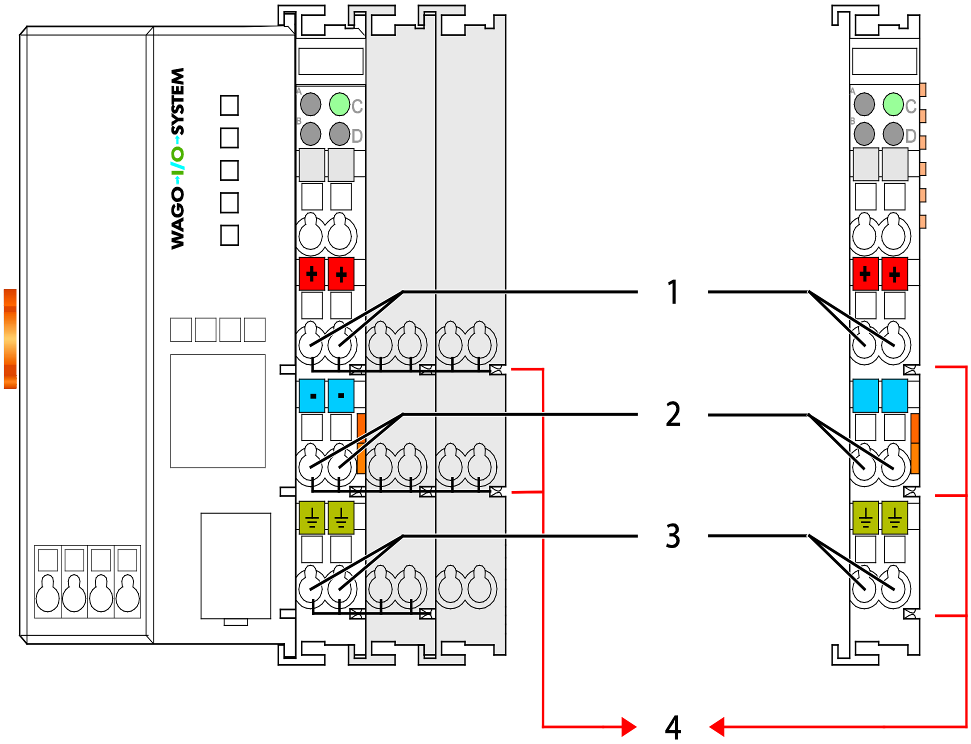


Figure 8: Field Supply (Sensor/Actuator)

Table 6: Legend for Figure "Field Supply (Sensor/Actuator) for ECO Fieldbus Coupler"

Field supply	
1	24 V (-15 % / +20 %)
2	0 V
3	Optional ground potential (functional earth)
Power jumper contacts	
4	Potential distribution to adjacent I/O modules

Note

In exceptional instances, I/O modules can be directly connected to the field supply!
The 24 V field supply can be connected also directly to a bus module, if the connection points are not needed for the peripheral device supply. In this case, the connection points need the connection to the power jumper contacts.

The field-side power supply is automatically derived from the power jumper contacts when snapping an I/O module.

The current load of the power contacts must not exceed 10 A on a continual basis.

By inserting an additional power supply module, the field supply via the power contacts is disrupted. From there a new power supply occurs which may also contain a new voltage potential.

Note

Re-establish the ground connection when the connection to the power jumper contacts is disrupted!
Some I/O modules have no or very few power contacts (depending on the I/O function). Due to this, the passing through of the relevant potential is disrupted. If you require a field supply via power jumper contacts for subsequent I/O modules, then you have to use a power supply module.
Note the data sheets of the I/O modules.

Note

Use a spacer module when setting up a node with different potentials!
In the case of a node setup with different potentials, e.g. the alteration from 24 VDC to 230 VAC, you should use a spacer module. The optical separation of the potentials acts as a warning to heed caution in the case of wiring and maintenance works. Thus, you can prevent the results of wiring errors.

3.5.4.2 Fusing via Power Supply Module

Internal fusing of the field supply is possible for various field voltages via an appropriate power supply module.

Table 7: Power Supply Modules

Order No.	Field Voltage
750-601	24 VDC, Supply/Fuse
750-609	230 VAC, Supply/Fuse
750-615	120 VAC, Supply/Fuse
750-617	24 VAC, Supply/Fuse
750-610	24 VDC, Supply/Fuse/Diagnosis
750-611	230 VAC, Supply/Fuse/Diagnosis
750-606	Supply Module 24 VDC, 1.0 A, Ex i
750-625/000-001	Supply Module 24 VDC, 1.0 A, Ex i (without diagnostics)

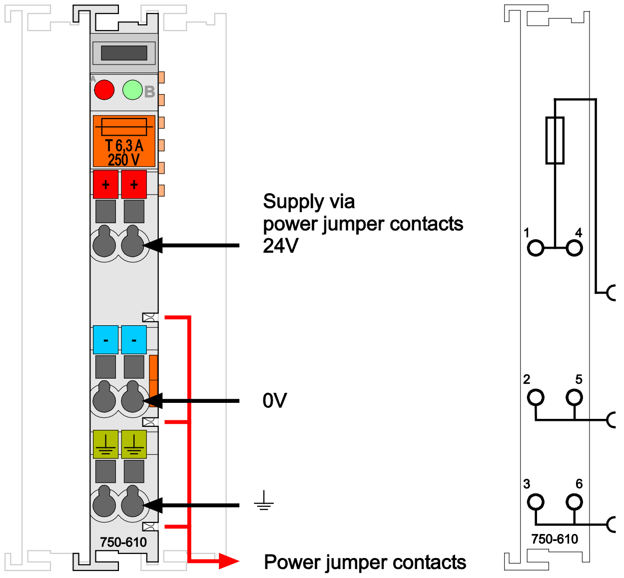


Figure 9: Supply Module with Fuse Carrier (Example 750-610)

NOTICE

Observe the maximum power dissipation and, if required, UL requirements!
In the case of power supply modules with fuse holders, you must only use fuses with a maximum dissipation of 1.6 W (IEC 127).
For UL approved systems only use UL approved fuses.

In order to insert or change a fuse, or to switch off the voltage in succeeding I/O modules, the fuse holder may be pulled out. In order to do this, use a screwdriver for example, to reach into one of the slits (one on both sides) and pull out the holder.

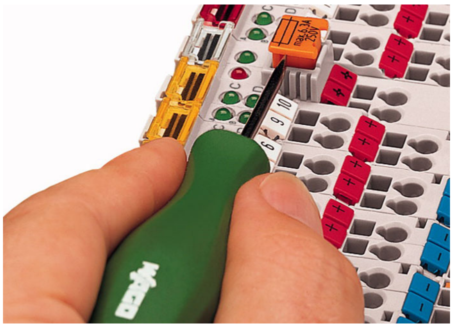


Figure 10: Removing the Fuse Carrier

Lifting the cover to the side opens the fuse carrier.

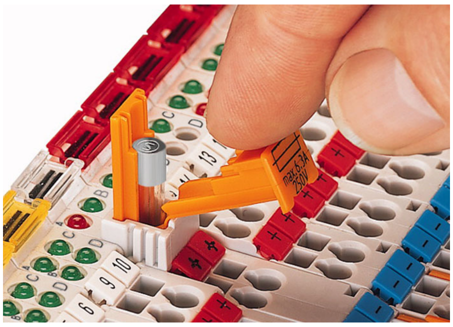


Figure 11: Opening the Fuse Carrier

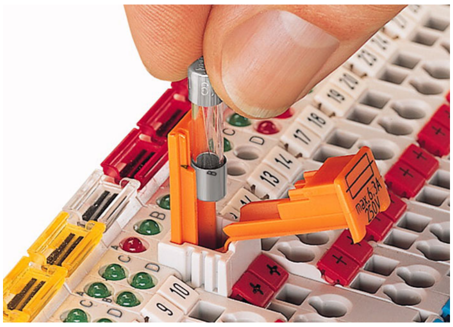


Figure 12: Changing the Fuse

After changing the fuse, the fuse carrier is pushed back into its original position.

3.5.4.3 Fusing External

NOTICE

Field supply only with appropriate fuse protection!
Without overcurrent protection, the electronics can be damaged.
If you alternatively implement the overcurrent protection for the field supply with an external fuse, an F 10 A fuse should be used.

For the external fusing, the fuse modules of the WAGO series 282, 2006, 281 and 2002 are suitable for this purpose.

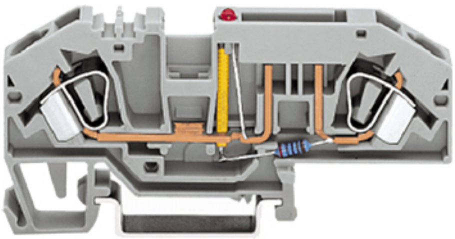


Figure 13: Fuse Modules for Automotive Fuses, Series 282

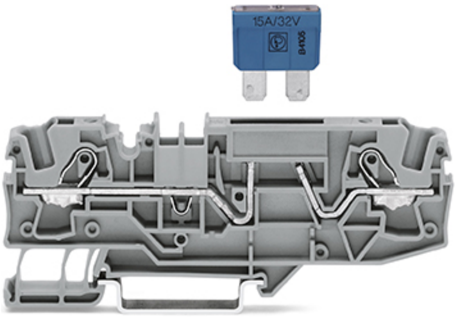


Figure 14: Fuse Modules for Automotive Fuses, Series 2006

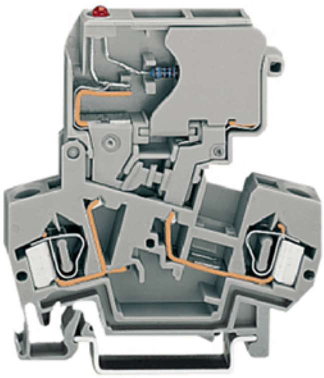


Figure 15: Fuse Modules with Pivotable Fuse Carrier, Series 281

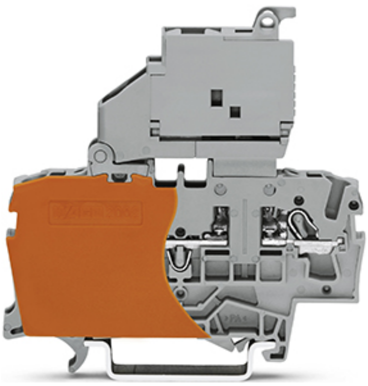


Figure 16: Fuse Modules with Pivotable Fuse Carrier, Series 2002

3.5.5 Supply Example

Note

The system supply and the field supply shall be separated!
You should separate the system supply and the field supply in order to ensure bus operation in the event of a short-circuit on the actuator side.

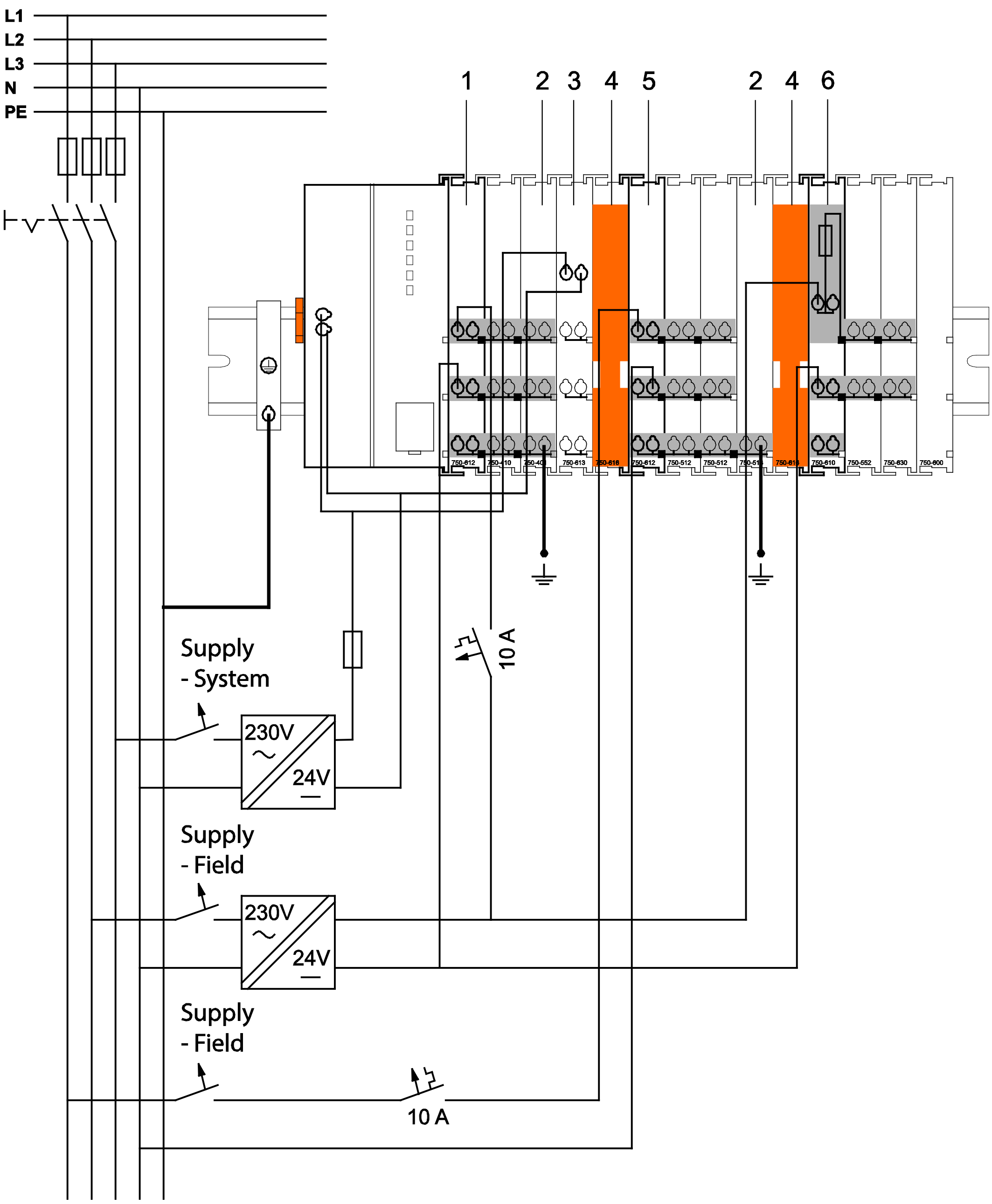


Figure 17: Supply Example

Table 8: Legend for Figure “Supply Example for Fieldbus Coupler/Controller”

Pos.	Description
1	Power Supply on fieldbus coupler/controller via external Supply Module
2	Power Supply with optional ground
3	Internal System Supply Module
4	Separation Module recommended
5	Supply Module passive
6	Supply Module with fuse carrier/diagnostics

3.5.6 Power Supply Unit

The WAGO-I/O-SYSTEM 750 requires a 24 VDC voltage (system supply).

Note

Recommendation
A stable power supply cannot always be assumed everywhere. Therefore, you should use regulated power supplies to ensure the quality of the supply voltage.

For brief voltage dips, a buffer (200 µF per 1 A load current) must be provided.

Note

Buffer for system power supply!
The system power supply must be buffered to bridge power outages. As the power demand depends on the respective node configuration, buffering is not implemented internally.
To achieve power outages of 1 ms to 10 ms according to IEC61131-2, determine the buffering appropriate for your node configuration and structure it as an external circuit.

The power demand must be determined individually depending on the entry point of the field supply. All loads through field devices and I/O modules must be taken into account. The field supply also impacts the I/O modules because the input and output drivers of some I/O modules require the voltage of the field supply.

Note

System and field supply must be isolated!
The system supply and field supply must be isolated to ensure bus operation in the event of short circuits on the actuator side.

Information

Power supply units are available in the eShop.
You can find suitable power supply units, e. g. from the EPSITRON series, in the eShop on www.wago.com.

3.6 Grounding

3.6.1 Grounding the DIN Rail

3.6.1.1 Framework Assembly

When setting up the framework, the carrier rail must be screwed together with the electrically conducting cabinet or housing frame. The framework or the housing must be grounded. The electrical connection is established via the screw. Thus, the carrier rail is grounded.

DANGER

Ensure sufficient grounding is provided!
You must take care to ensure the flawless electrical connection between the carrier rail and the frame or housing in order to guarantee sufficient grounding.

3.6.1.2 Insulated Assembly

Insulated assembly has been achieved when there is constructively no direct ohmic contact between the cabinet frame or machine parts and the carrier rail. Here, the earth ground must be set up via an electrical conductor in accordance with valid national safety regulations.

Note

Recommendation
The optimal setup is a metallic assembly plate with grounding connection which is electrically conductive linked to the carrier rail.

The separate grounding of the carrier rail can be easily set up with the aid of the WAGO ground wire terminals.

Table 9: WAGO Ground Wire Terminals

Order No.	Description
283-609	1-conductor ground (earth) terminal block make an automatic contact to the carrier rail; conductor cross section: 0.2 mm² … 16 mm^2 **Note**: Also order the end and intermediate plate (283-320).

3.6.2 Grounding Function

The grounding function increases the resistance against electro-magnetic interferences. Some components in the I/O system have a carrier rail contact that dissipates electro-magnetic interferences to the carrier rail.

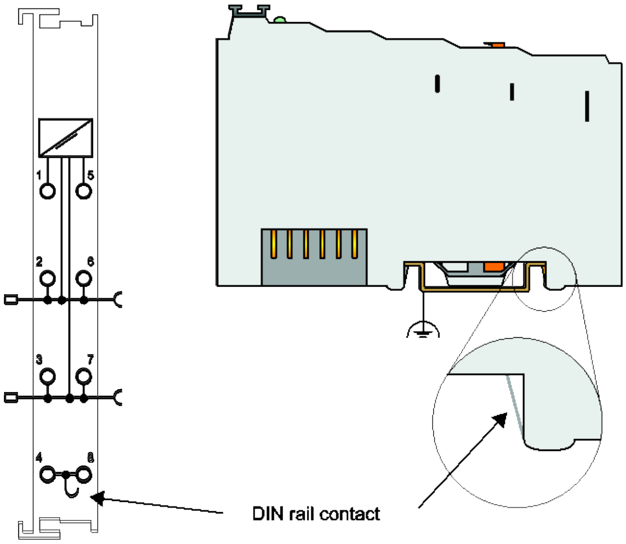


Figure 18: Carrier Rail Contact (Example)

DANGER

Ensure sufficient grounding is provided!
You must take care to ensure the direct electrical connection between the carrier rail contact and the carrier rail.
The carrier rail must be grounded.
For information on carrier rail properties, see section “Mounting” > … > “Carrier Rail Properties”.

The bottom CAGE CLAMP® connectors of the supply modules enable optional connection of a field-side functional ground. This potential is made available to the I/O module arranged on the right through the spring-loaded contact of the three power contacts. Some I/O modules are equipped with a knife-edge contact that taps this potential. This forms a potential group with regard to functional ground with the I/O module arranged on the left.

3.7 Shielding

3.7.1 General

Use of shielded cables reduces electromagnetic interference and thus increases signal quality. Measurement errors, data transmission errors and interference due to excessive voltage can be prevented.

Note

Connect the cable shield to the ground potential!
Integrated shielding is mandatory to meet the technical specifications in regards to measuring accuracy. Connect the cable shield and ground potential at the inlet to the cabinet or housing. This allows induced interference to dissipate and to be kept away from devices in the cabinet or housing.

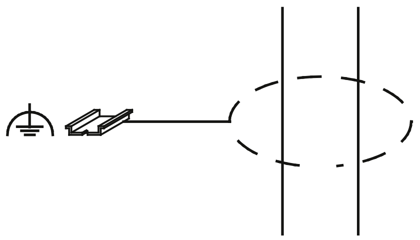


Figure 19: Cable Shield at Ground Potential

Note

Improve shielding performance by placing the shield over a large area!
Higher shielding performance is achieved via low-impedance connection between shield and ground. For this purpose, connect the shield over a large surface area, e.g., WAGO shield connecting system. This is especially recommended for large-scale systems where equalizing current or high impulse-type currents caused by atmospheric discharge may occur.

Note

Keep data and signal lines away from sources of interference!
Route data and signal lines separately from all high voltage cables and other sources of high electromagnetic emission (e.g., frequency converter or drives).

3.7.2 Bus Cables

The shielding of bus lines is described in the respective configuration guidelines and standards of the bus system.

3.7.3 Shielded Signal Lines

Note

Use shielded signal lines!
Always use shielded signal lines for analog signals and I/O modules which are equipped with shield clamps. Only then you can ensure that the accuracy and interference immunity specified for the respective I/O module can be achieved even in the presence of interference acting on the signal cable.

On some WAGO devices you can directly clamp the shield. For all other devices use the WAGO shield connecting system.

3.7.4 WAGO Shield Connecting System

The series 790 WAGO shield connecting system consists of shield clamping saddles, busbars and various mounting carriers. These components can be used to achieve many different configurations.

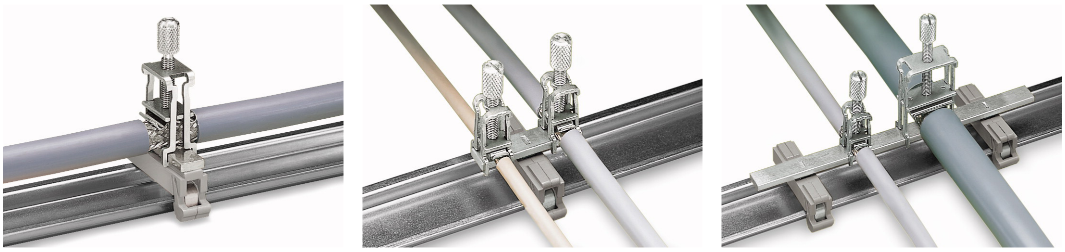


Figure 20: Examples of the WAGO Shield Connecting System

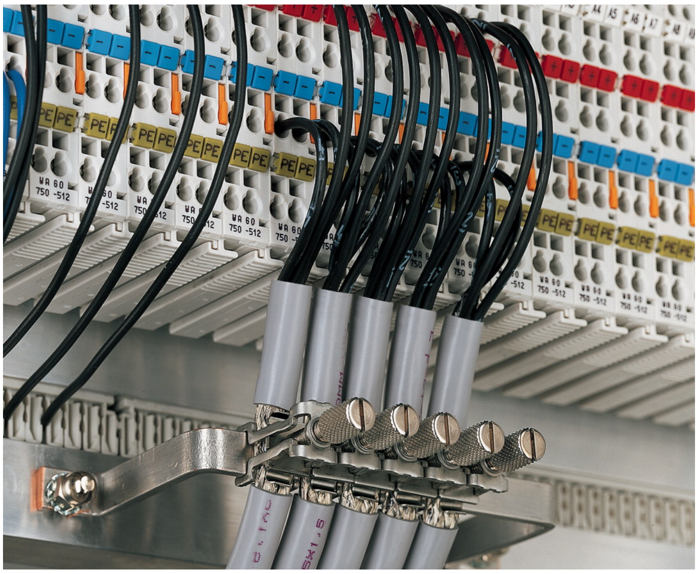


Figure 21: Application of the WAGO Shield Connecting System

4 Device Description

The Fieldbus coupler 750-354/000-001 connects the WAGO-I/O-SYSTEM with the Fieldbus system EtherCAT®.

This coupler can be used for applications in machine and plant construction as well as in the process industry and building technology.

Equipped with two RJ-45 ports, one In port and one Out port, the Fieldbus coupler enables easy and cost-effective cabling such as linear bus topology.

In the Fieldbus Coupler, all input signals from the sensors are combined. After connecting the Fieldbus Coupler, the Fieldbus Coupler determines which I/O modules are on the node and creates a local process image from these. Analog and specialty module data is sent via words and/or bytes; digital data is grouped bit-by-bit.

The local process image is divided into two data zones containing the data received and the data to be sent.

The data of the analog modules is mapped first into the process image. The modules are mapped in the order of their physical position after the Coupler.

The bits of the digital modules are combined into words and then mapped after the analog ones in the process image. If the number of digital I/Os is greater than 16 bits, the Fieldbus Coupler automatically begins a new word.

All sensor input signals are grouped in the coupler (slave) and transferred to the higher-order controller (master) via the fieldbus. Process data linking is performed in the higher-order controller. The higher-order controller puts out the resulting data to the actuators via the bus and the node.

4.1 View

The view below shows the different parts of the device:

- The fieldbus connection (X1, X2) is within the lower range on the left side.
- Over the fieldbus connection is a power supply unit (X3) for the system supply.
- LEDs for bus communication, error messages and diagnostics are within the upper range on the right side.
- Down right the service interface is to be found behind the flap.

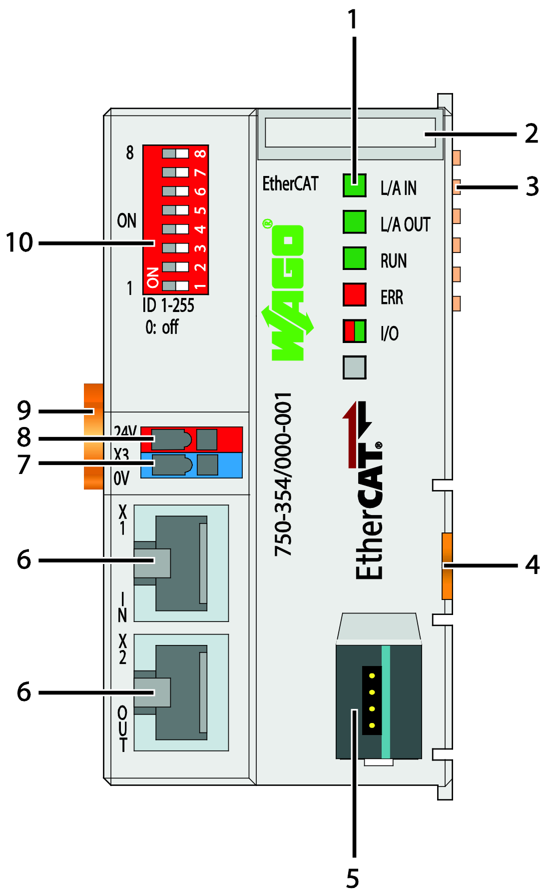


Figure 22: View EtherCAT® fieldbus coupler

Table 10: Legend for Figure “View EtherCAT® Fieldbus Coupler”

Pos.	Desig-nation	Meaning	Details see Section
1	L/A IN, L/A OUT, RUN, ERR, I/O	Status LEDs Fieldbus	„Device Description“ > „Display Elements“
2	---	Marking possibility on four miniature WSB markers	---
3	---	Data contacts	"Connect Devices" > "Data Contacts/Local Bus"
4	---	Unlocking lug	"Mounting" > "Inserting and Removing Devices"
5	---	Service interface (open flap)	"Device Description" > "Operating Elements"
6	X1 IN, X2 OUT	Fieldbus connection RJ-45	"Device Description" > "Connectors"
7	-	CAGE CLAMP® Connections System Supply 0 VDC	"System Description" >"Voltage Supply"
8	+	CAGE CLAMP® Connections System Supply 24 VDC	"System Description" >"Voltage Supply"
9	---	Locking Disc	„Mounting“ > „Plugging and Removal of the Device“
10	---	ID Selection Switch	"Device Description" > "Operating Elements"

4.2 Connectors

4.2.1 Device Supply

The device is powered via terminal blocks with CAGE CLAMP® connections.

The device supply generates the necessary voltage to power the electronics of the device and the internal electronics of the connected I/O modules.

The fieldbus interface is galvanically separated to the electrical potential of the device.

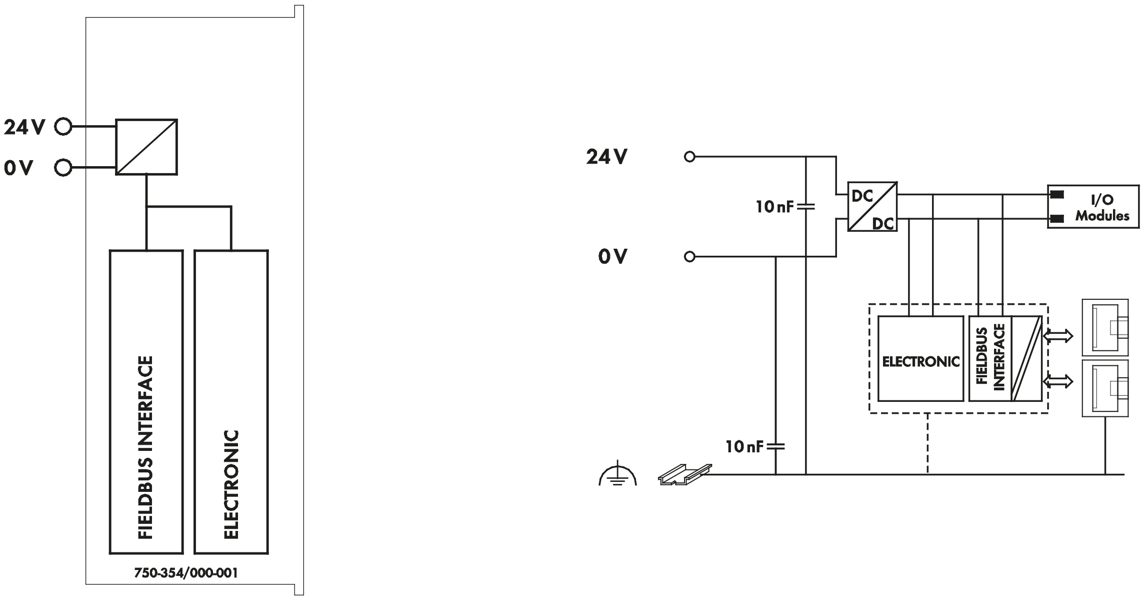


Figure 23: Device Supply

4.2.2 Fieldbus Connection

Connection to the fieldbus is by two RJ-45 connectors. The EtherCAT® coupler supports a transmission speed of 100 Mbit/s and full-duplex operations.

The RJ-45 socket on the fieldbus couplers are wired per the 100BaseTX standard. The colors of the wires should correspond to the assignment T568B according to TIA/EIA-568-B.

A fully-shielded twisted pair cable (SF/FTP, S/FTP, S/UTP) is exclusively recommended as the connecting cable. The maximum length of the transmission cable is 100 m, if transmission of Class D is achieved in accordance with EN 50173.

The socket is arranged physically lower, allowing the coupler to fit in an 80 mm high enclosure after plug connection.

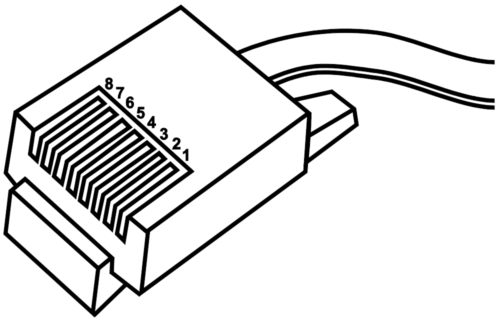


Figure 24: RJ-45 Connector

Table 11: RJ-45 Connector and RJ-45 Connector Configuration

Contact	Signal	
1	TD +	Transmit +
2	TD −	Transmit −
3	RD +	Receive +
4		free
5		free
6	RD −	Receive −
7		free
8		free

NOTICE

Do not use in telecommunication circuits!
Only use devices equipped with ETHERNET or RJ-45 connectors in LANs.
Never connect these devices with telecommunication networks.

4.3 Display Elements

The operating condition of the fieldbus coupler or the node is displayed with the help of illuminated indicators in the form of light-emitting diodes (LEDs).
The LED information is routed to the top of the case by light guides. In some cases, the LEDs are multi-colored (red, green or orange).

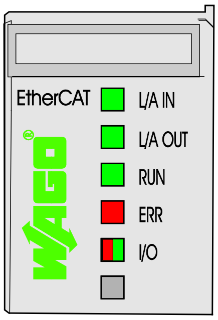


Figure 25: Display Elements

For the diagnostics of the different ranges fieldbus and node, the LED's can be divided into groups:

Table 12: Display Elements Fieldbus Status

LED	Color	Meaning
L/A IN	green	indicates a network connection and an activity at port X1 (IN)
L/A OUT	green	indicates a network connection and an activity at port X2 (OUT)
RUN	green	indicates the specific EtherCAT® status (Application Layer Status)
ERR	red	indicates a fieldbus error

Table 13: Display Elements Node Status

LED	Color	Meaning
I/O	red/green/ orange	Indicates the operation of the node and signals via a blink code faults encountered.

Information

More information about the LED Signaling
Read the detailed description for the evaluation of the displayed LED state in the section "Diagnostics" > … > "LED Signaling".

4.4 Operating Elements

4.4.1 Service Interface

The service interface is located behind the flap.

It is used for the communication with the WAGO-I/O-*CHECK* and for downloading the firmware updates.

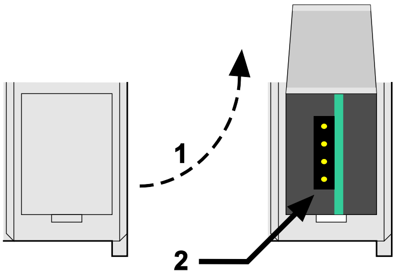


Figure 26: Service Interface (Closed and Opened Flap)

Table 14: Legend for Figure "Service Interface (Closed and Opened Flap)"

Number	Description
1	Open closed
2	View Service Interface

NOTICE

Device must be de-energized!
To prevent damage to the device, unplug and plug in the communication cable only when the device is de-energized!

The connection to the 4-pin header under the cover flap can be realized via the communication cables with the item numbers750-920 and 750-923 or via the WAGO radio adapter with the item number 750-921.

Alternatively, the connection can be made using either a USB cable (WAGO USB Service Cable 750-923) or a *Bluetooth*® dongle (WAGO Radio Adapter 750-921).

Information

Additional Information about the USB cable and *Bluetooth*® Dongle!
More information about the WAGO USB Service Cable 750-923 and the WAGO Radio Adapter 750-921 is available on the WAGO Internet site at: www.wago.com.

4.4.2 ID Selection Switch

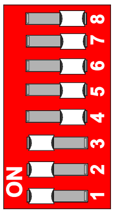


Figure 27: ID Selection Switch (for Example Setting "7")

The ID selection switch is used to set the lower eight bits (bit 0 to bit 7) for the "Explicit Device ID" (EDI). The "Explicit Device ID" is used to uniquely identify an EtherCAT® slave. The "Explicit Device ID" set on the device may only occur once in the existing EtherCAT® network.
The "Explicit Device ID" must be used when using "Hot Connect" applications and as protection against interchanging devices in the network. See Section "Functional Description" > "Hot Connect".
"Explicit Device ID" value 0 is invalid.

The "Explicit Device ID" is coded bit by bit and begins at bit 0 (LSB) with ID selection switch 1 and ends with bit 7 (MSB) at ID selection switch 8.
The upper eight bits (bit 8 to bit 15) of the "Explicit Device ID" can be configured via the SDO object 0x2134 and then stored in the device. Together, the two values form the fixed "Explicit Device ID" value of the WAGO EtherCAT® slave.

Table 15: EDI Address Structure ("Explicit Device ID")

Bit 15 ...8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Objekt 0x2134	DIP S8	DIP S7	DIP S6	DIP S5	DIP S4	DIP S3	DIP S2	DIP S1

Information

More information about object 0x2134!
Further information about Object 0x2134 is given in the Section "Function Description" → "Object Dictionary" → "Object 0x2134".

4.5 Technical Data

4.5.1 Device Data

Table 16: Technical Data – Device Data

Width	50 mm
Height	65 mm (from upper-edge of DIN 35 rail)
Length	97 mm
Weight	100 g
Material	Polycarbonate, Polyamide 6.6
Installation	On DIN 35 rail with interlock
Modular interconnection	Double feather key dovetail
Mounting position	Any orientation

4.5.2 Safe Electrical Isolation

Table 17: Technical Data – Safe Electrical Isolation

Air and creepage distance	Acc. to IEC 60664-1

4.5.3 Supply

Table 18: Technical Data – Supply

Nominal voltage supply	24 VDC
Voltage Supply	24 VDC (−25 % ... +30 %)
Input currenttyp. with nominal load	250 mA at 24 V
Efficiency of the power supply	85 %
Internal current consumption	300 mA at 5 V
Total current for I/O modules	700 mA at 5 V

4.5.4 Communication

Table 19: Technical Data – Communication

Number of Fieldbus nodes per Master	Limited by ETHERNET® specification
Transmission medium	Shielded twisted pair S/FTP, F/FTP or SF/FTP; 100 Ω, Cat 6
Buscoupler connection	2 × RJ-45
Baud rate	100 Mbit/s
Transmission performance	Class D gem. EN 50173
Protocols	EtherCAT® (direct mode)
Number of I/O modules – with bus extension	64 64
Fieldbus Input process image max Output process image max	 1024 Bytes 1024 Bytes
Configuration	Via PC

4.5.5 Accessories

Table 20: Technical Data – Accessories

Miniature WSB Quick marking system

4.5.6 Connection Type

Table 21: Technical Data – Field Wiring

Wire connection	Push-in CAGE CLAMP®
Cross section	0.08 mm² … 1.5 mm² / AWG 28 … 16
Stripped length	8 mm … 9 mm / 0.33 in

Table 22: Technical Data – Power Jumper Contacts

Power jumper contacts	Blade/spring contact, self-cleaning

Table 23: Technical Data – Data Contacts

Data contacts	Slide contact, hard gold plated, self-cleaning

4.5.7 Climatic Environmental Conditions

Table 24: Technical Data – Climatic Environmental Conditions

Surrounding air temperature, operation	0 °C … 55 °C
Surrounding air temperature, storage	−25 °C … +85 °C
Operating altitude	without temperature derating: 0 … 2000 m; with temperature derating: 2000 … 5000 m (0.5 K/100 m); max.: 5000 m
Relative humidity	Max. 5 % … 95 % without condensation
Pollution degree	2
Protection type	IP20
Resistance to harmful substances	Acc. to IEC 60068-2-42 and IEC 60068-2-43
Maximum pollutant concentration at relative humidity < 75 %	$SO_2 \leq 25$ ppm $H_2S \leq 10$ ppm
Special conditions	Ensure that additional measures for components are taken, which are used in an environment involving: – dust, caustic vapors or gases – ionizing radiation

4.5.8 Mechanical Strength

Table 25: Technical Data – Mechanical Strength

Vibration resistance	Acc. to IEC 60068-2-6 Comment to the vibration resistance: a) Type of oscillation: sweep with a rate of change of 1 octave per minute 10 Hz ≤ f < 57 Hz, const. Amplitude 0,075 mm 57 Hz ≤ f < 150 Hz, const. Acceleration 1 g b) Period of oscillation: 10 sweep per axis in each of the 3 vertical axes
Shock resistance	Acc. to IEC 60068-2-27 Comment to the shock resistance: a) Type of impulse: half sinusoidal b) Intensity of impulse: 15 g peak value, 11 ms maintenance time c) Route of impulse: 3 impulses in each pos. And neg. direction of the 3 vertical axes of the test object, this means 18 impulses in all.
Free fall	Acc. IEC 60068-2-32 ≤ 1 m (module in original packing)

4.6 Approvals

Information

More information about approvals.
Detailed references to the approvals are listed in the document "Overview Approvals **WAGO-I/O-SYSTEM 750**", which you can find via the internet under: www.wago.com > SERVICES > DOWNLOADS > Additional documentation and information on automation products > WAGO-I/O-SYSTEM 750 > System Description.

The following approvals have been granted to 750-354/000-001 fieldbus coupler/controller:

Conformity Marking	
UL508	
Korea Certification	MSIP-REM-W43-FBC750

The following Ex approvals have been granted to 750-354/000-001 fieldbus coupler/controller:

TÜV 14 ATEX 148929 X

II 3 G Ex nA IIC T4 Gc

IECEx TUN 14.0035 X

Ex nA IIC T4 Gc

$_{\text{C}}\text{UL}_{\text{US}}$ ANSI/ISA 12.12.01

Class I, Div2 ABCD T4

The following ship approvals have been granted to 750-354/000-001 fieldbus coupler/controller:

GL (Germanischer Lloyd)	Cat. A, B, C, D (EMC 1)

4.7 Standards and Guidelines

750-354/000-001 meets the following requirements on emission and immunity of interference:

EMC CE-Immunity to interference	EN 61000-6-2
	and to EN 61131-2
EMC CE-Emission of interference	EN 61000-6-3
	and to EN 61131-2
EMC marine applications-Immunity to interference	acc. to DNV GL
EMC marine applications-Emission of interference	acc. to DNV GL

5 Mounting

5.1 Installation Position

Along with horizontal and vertical installation, all other installation positions are allowed.

Note

Use an end stop in the case of vertical mounting!
In the case of vertical assembly, an end stop has to be mounted as an additional safeguard against slipping.

WAGO order no. 249-116	End stop for DIN 35 rail, 6 mm wide
WAGO order no. 249-117	End stop for DIN 35 rail, 10 mm wide

5.2 Overall Configuration

The maximum total length of a fieldbus node without fieldbus coupler/controller is 780 mm including end module. The width of the end module is 12 mm. When assembled, the I/O modules have a maximum length of 768 mm.

Examples:

- 64 I/O modules with a 12 mm width can be connected to a fieldbus coupler/controller.
- 32 I/O modules with a 24 mm width can be connected to a fieldbus coupler/controller.

Exception:

The number of connected I/O modules also depends on the type of fieldbus coupler/controller is used. For example, the maximum number of stackable I/O modules on one PROFIBUS DP/V1 fieldbus coupler/controller is 63 with no passive I/O modules and end module.

NOTICE

Observe maximum total length of a fieldbus node!
The maximum total length of a fieldbus node without fieldbus coupler/controller and without using a 750-628 I/O Module (coupler module for internal data bus extension) may not exceed 780 mm.
Also note the limitations of individual fieldbus couplers/controllers.

Note

Increase the total length using a coupler module for internal data bus extension!

You can increase the total length of a fieldbus node by using a 750-628 I/O Module (coupler module for internal data bus extension). For such a configuration, attach a 750-627 I/O Module (end module for internal data bus extension) after the last I/O module of a module assembly. Use an RJ-45 patch cable to connect the I/O module to the coupler module for internal data bus extension of another module block.
This allows you to segment a fieldbus node into a maximum of 11 blocks with maximum of 10 I/O modules for internal data bus extension.
The maximum cable length between two blocks is five meters.
More information is available in the manuals for the 750-627 and 750-628 I/O Modules.

5.3 Mounting onto Carrier Rail

5.3.1 Carrier Rail Properties

All system components can be snapped directly onto a carrier rail in accordance with the European standard EN 60175 (DIN 35).

NOTICE

Do not use any third-party carrier rails without approval by WAGO!
WAGO Kontakttechnik GmbH & Co. KG supplies standardized carrier rails that are optimal for use with the I/O system. If other carrier rails are used, then a technical inspection and approval of the rail by WAGO Kontakttechnik GmbH & Co. KG should take place.

Carrier rails have different mechanical and electrical properties. For the optimal system setup on a carrier rail, certain guidelines must be observed:

- The material must be non-corrosive.
- Most components have a contact to the carrier rail to ground electro-magnetic disturbances. In order to avoid corrosion, this tin-plated carrier rail contact must not form a galvanic cell with the material of the carrier rail which generates a differential voltage above 0.5 V (saline solution of 0.3 % at 20°C).
- The carrier rail must optimally support the EMC measures integrated into the system and the shielding of the I/O module connections.
- A sufficiently stable carrier rail should be selected and, if necessary, several mounting points (every 20 cm) should be used in order to prevent bending and twisting (torsion).
- The geometry of the carrier rail must not be altered in order to secure the safe hold of the components. In particular, when shortening or mounting the carrier rail, it must not be crushed or bent.
- The base of the I/O components extends into the profile of the carrier rail. For carrier rails with a height of 7.5 mm, mounting points are to be riveted under the node in the carrier rail (slotted head captive screws or blind rivets).
- The metal springs on the bottom of the housing must have low-impedance contact with the DIN rail (wide contact surface is possible).

5.3.2 WAGO DIN Rails

WAGO carrier rails meet the electrical and mechanical requirements shown in the table below.

Table 26: WAGO DIN Rails

Item No.	Description
210-112	35 × 7.5; 1 mm; steel; bluish, tinned, chromed; slotted
210-113	35 × 7.5; 1 mm; steel; bluish, tinned, chromed; unslotted
210-197	35 × 15; 1.5 mm; steel; bluish, tinned, chromed; slotted
210-114	35 × 15; 1.5 mm; steel; bluish, tinned, chromed; unslotted
210-118	35 × 15; 2.3 mm; steel; bluish, tinned, chromed; unslotted
210-198	35 × 15; 2.3 mm; copper; unslotted
210-196	35 × 8.2; 1.6 mm; aluminum; unslotted

5.4 Spacing

The spacing between adjacent components, cable conduits, casing and frame sides must be maintained for the complete fieldbus node.

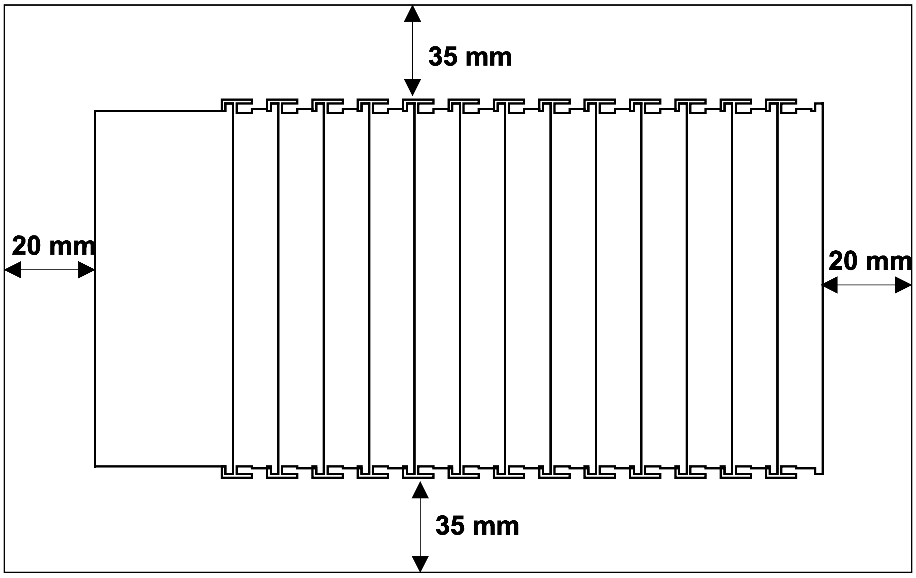


Figure 28: Spacing

The spacing creates room for heat transfer, installation or wiring. The spacing to cable conduits also prevents conducted electromagnetic interferences from influencing the operation.

5.5 Mounting Sequence

Fieldbus couplers, controllers and I/O modules of the WAGO-I/O-SYSTEM 750 are snapped directly on a carrier rail in accordance with the European standard EN 60175 (DIN 35).

The reliable positioning and connection is made using a tongue and groove system. Due to the automatic locking, the individual devices are securely seated on the rail after installation.

Starting with the fieldbus coupler or controller, the I/O modules are mounted adjacent to each other according to the project design. Errors in the design of the node in terms of the potential groups (connection via the power contacts) are recognized, as the I/O modules with power contacts (blade contacts) cannot be linked to I/O modules with fewer power contacts.

⚠ CAUTION

Risk of injury due to sharp-edged blade contacts!
The blade contacts are sharp-edged. Handle the I/O module carefully to prevent injury. Do not touch the blade contacts.

NOTICE

Insert I/O modules only from the proper direction!
All I/O modules feature grooves for power jumper contacts on the right side. For some I/O modules, the grooves are closed on the top. Therefore, I/O modules featuring a power jumper contact on the left side cannot be snapped from the top. This mechanical coding helps to avoid configuration errors, which may destroy the I/O modules. Therefore, insert I/O modules only from the right and from the top.

Note

Don't forget the bus end module!
Always plug a bus end module (750-600) onto the end of the fieldbus node! You must always use a bus end module at all fieldbus nodes with WAGO-I/O-SYSTEM 750 fieldbus couplers or controllers to guarantee proper data transfer.

5.6 Inserting and Removing Devices

DANGER

Do not work when devices are energized!
High voltage can cause electric shock or burns.
Switch off all power to the device prior to performing any installation, repair or maintenance work.

5.6.1 Inserting the Fieldbus Coupler/Controller

1. When replacing the fieldbus coupler/controller for an already available fieldbus coupler/controller, position the new fieldbus coupler/controller so that the tongue and groove joints to the subsequent I/O module are engaged.

2. Snap the fieldbus coupler/controller onto the carrier rail.

3. Use a screwdriver blade to turn the locking disc until the nose of the locking disc engages behind the carrier rail (see the following figure). This prevents the fieldbus coupler/controller from canting on the carrier rail.

With the fieldbus coupler/controller snapped in place, the electrical connections for the data contacts and power contacts (if any) to the possible subsequent I/O module are established.

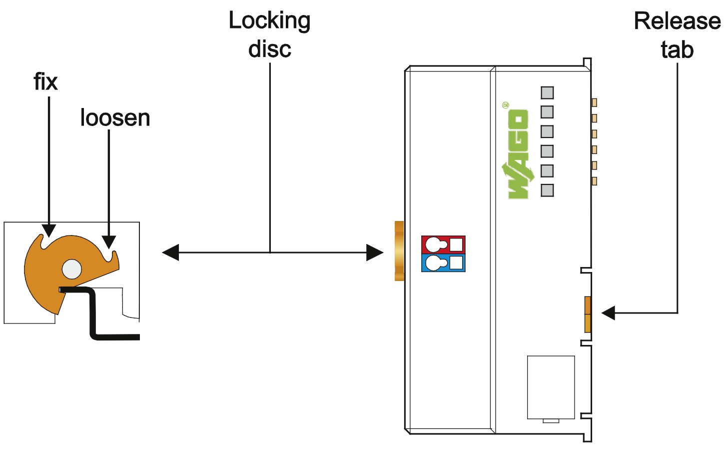


Figure 29: Release Tab

5.6.2 Removing the Fieldbus Coupler/Controller

1. Use a screwdriver blade to turn the locking disc until the nose of the locking disc no longer engages behind the carrier rail.

2. Remove the fieldbus coupler/controller from the assembly by pulling the release tab.

Electrical connections for data or power contacts to adjacent I/O modules are disconnected when removing the fieldbus coupler/controller.

5.6.3 Inserting the I/O Module

1. Position the I/O module so that the tongue and groove joints to the fieldbus coupler or controller or to the previous or possibly subsequent I/O module are
   engaged.

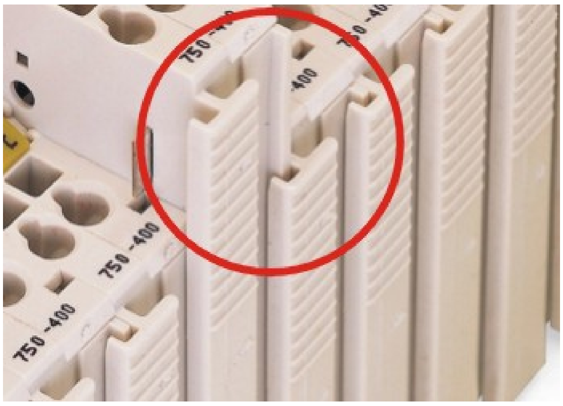


Figure 30: Insert I/O Module (Example)

2. Press the I/O module into the assembly until the I/O module snaps into the carrier rail.

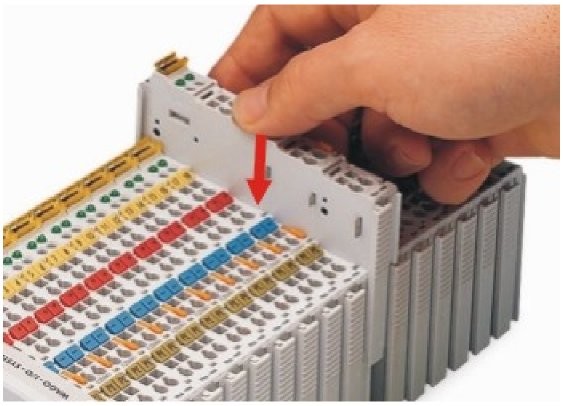


Figure 31: Snap the I/O Module into Place (Example)

With the I/O module snapped in place, the electrical connections for the data contacts and power jumper contacts (if any) to the fieldbus coupler or controller or to the previous or possibly subsequent I/O module are established.

5.6.4 Removing the I/O Module

1. Remove the I/O module from the assembly by pulling the release tab.

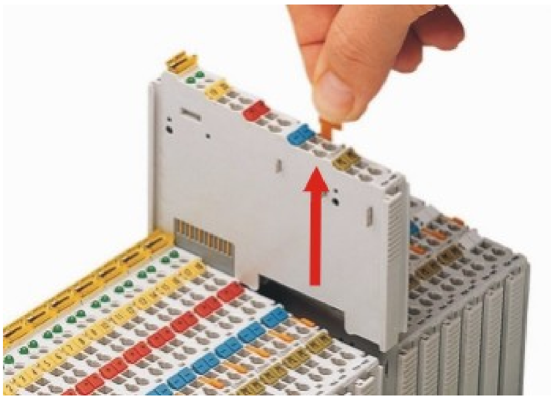


Figure 32: Removing the I/O Module (Example)

Electrical connections for data or power jumper contacts are disconnected when removing the I/O module.

6 Connect Devices

6.1 Data Contacts/Local Bus

Communication between the fieldbus coupler/controller and the I/O modules as well as the system supply of the I/O modules is carried out via the local bus. It is comprised of 6 data contacts, which are available as self-cleaning gold spring contacts.

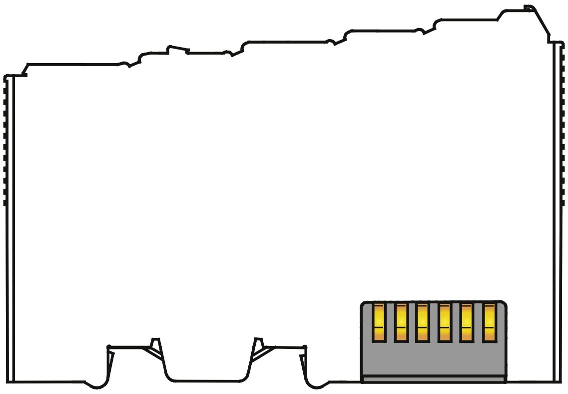


Figure 33: Data Contacts

NOTICE

Do not place the I/O modules on the gold spring contacts!
Do not place the I/O modules on the gold spring contacts in order to avoid soiling or scratching!

NOTICE

Ensure that the environment is well grounded!
The devices are equipped with electronic components that may be destroyed by electrostatic discharge. When handling the devices, ensure that the environment (persons, workplace and packing) is well grounded. Avoid touching conductive components, e.g. data contacts.

6.2 Power Contacts/Field Supply

⚠ CAUTION

Risk of injury due to sharp-edged blade contacts!
The blade contacts are sharp-edged. Handle the I/O module carefully to prevent injury. Do not touch the blade contacts.

Self-cleaning power jumper contacts used to supply the field side are located on the right side of most of the fieldbus couplers/controllers and on some of the I/O modules. These contacts come as touch-proof spring contacts. As fitting counterparts the I/O modules have male contacts on the left side.

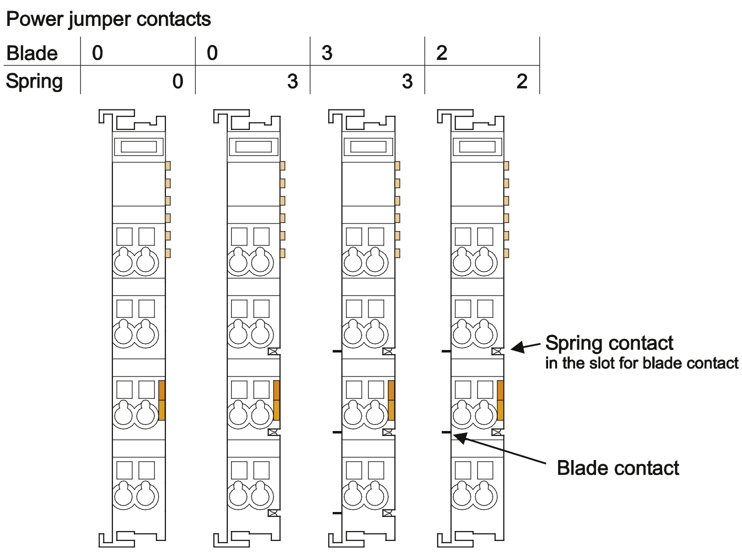


Figure 34: Example for the Arrangement of Power Contacts

Note

Field bus node configuration and test via smartDESIGNER
With the WAGO ProServe® Software smartDESIGNER, you can configure the structure of a fieldbus node. You can test the configuration via the integrated accuracy check.

6.3 Connecting a Conductor to the CAGE CLAMP®

The WAGO CAGE CLAMP® connection is appropriate for solid, stranded and finely stranded conductors.

Note

Only connect one conductor to each CAGE CLAMP®!
Only one conductor may be connected to each CAGE CLAMP®.
Do not connect more than one conductor at one single connection!

If more than one conductor must be routed to one connection, these must be connected in an up-circuit wiring assembly, for example using WAGO feed-through terminals.

1. For opening the CAGE CLAMP® insert the actuating tool into the opening above the connection.

2. Insert the conductor into the corresponding connection opening.

3. For closing the CAGE CLAMP® simply remove the tool. The conductor is now clamped firmly in place.

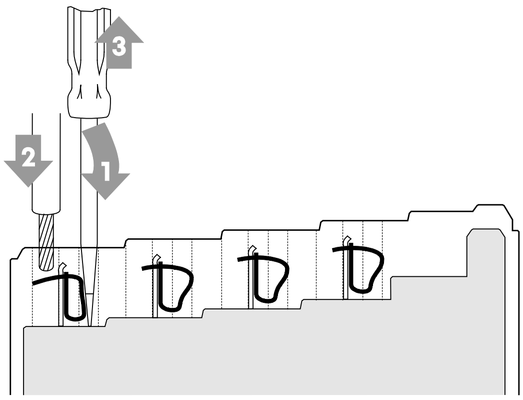


Figure 35: Connecting a Conductor to a CAGE CLAMP®

7 Function Description

7.1 Starting Up the Fieldbus Coupler

Once the master switch has been configured and the fieldbus coupler and the I/O modules have been electrically installed, the fieldbus node starts running.

After the power supply has been switched on, the fieldbus coupler implements an initialization phase. In the initialization phase, the firmware for the fieldbus coupler is started first.
During the firmware start, the I/O LED is red.

Subsequently, the fieldbus coupler determines the information from the connected I/O modules that is required to run the fieldbus node. During this phase, the I/O LED will flash red.

After a trouble-free start-up, the fieldbus coupler enters the INIT status and the I/O LED is green.

If an error occurs during start up, an error message is indicated by a blink code of the I/O LED.

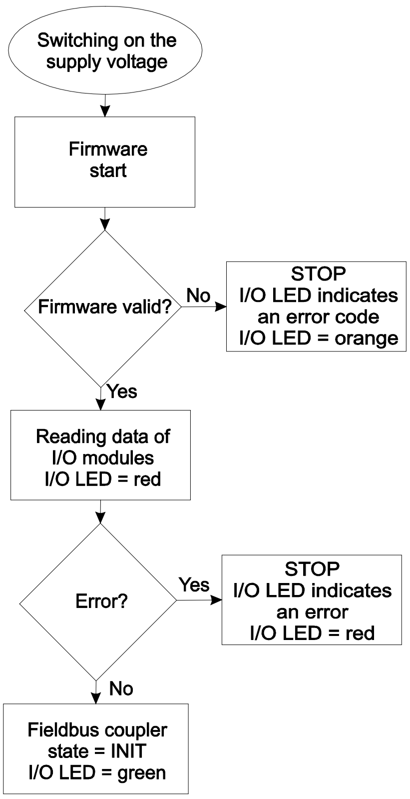


Figure 1: Starting up of the Fieldbus Coupler

Information

More information about the LED Signaling
Read the detailed description for the evaluation of the displayed LED state in the section “Diagnostics” > … > “LED Signaling”.

7.2 Start Behavior

When using the EtherCAT® fieldbus coupler, after the self test and the initiation of the local bus boot process, the EtherCAT® slave stack is started in parallel. The slave is then in INIT status.

The master can now switch the slave into PREOP status. If the request for PREOP status occurs before the local bus initialization is completed, then the transition from INIT status to PREOP status is delayed until the local bus initialization has been fully and successfully completed. In the event of an error in the local bus initialization, the transition from INIT status to PREOP status is denied with a corresponding AL status code.

Information

Additional information about AL status codes!
A list of the AL status codes and their meanings can be found in the appendix, "AL Status Codes" chapter.

Following a successful local bus initialization, the creation of the process image is determined and the corresponding objects with the terminal input and output data as well as the objects with the process data mapping (RxPDO's and TxPDO's) are entered in the Object Dictionary.
The process data architecture can only be redetermined by switching the fieldbus coupler operating voltage off and on again. If the terminal architecture is changed during operation, the fieldbus coupler enters an error state that can only exited by switching the fieldbus coupler operating voltage off and on again.

After the local bus initialization has been successfully completed, every other status transition from INIT to PREOP is accepted without delay, even if the local bus is not running due to an error.

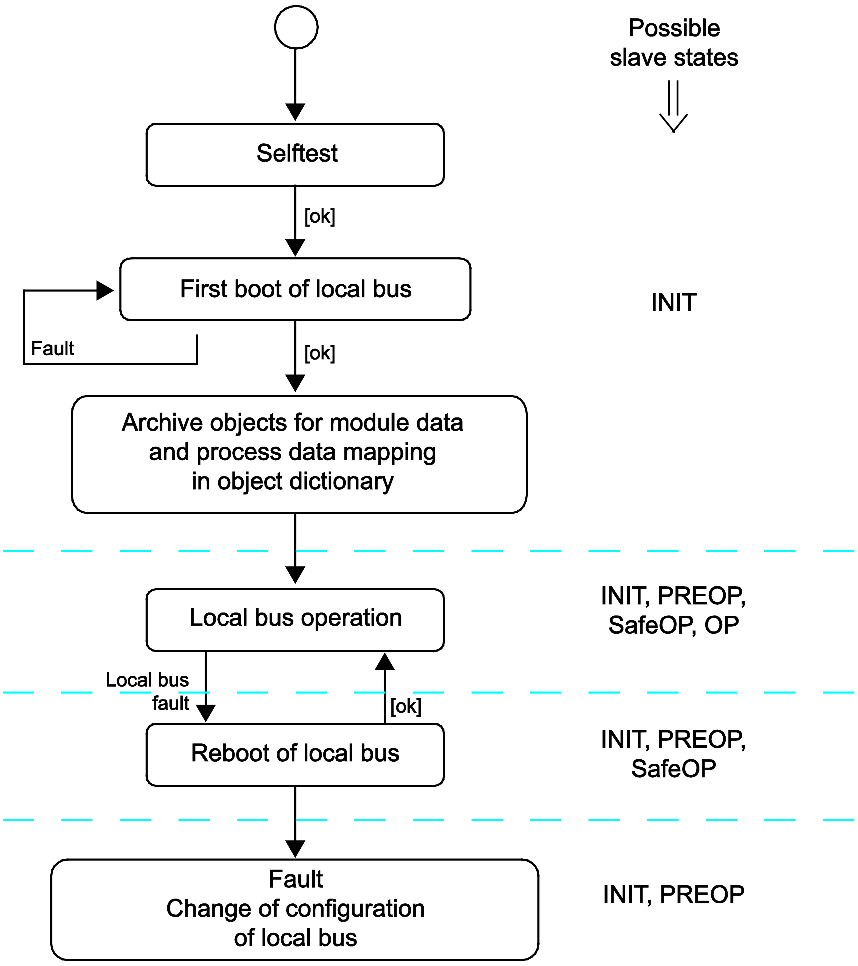


Figure 36: Start Behavior

7.3 Process Data Architecture

After switching on the supply voltage, the fieldbus coupler identifies all I/O modules connected with the node that send or receive data (data width/bit width > 0). In the maximum total extension the node can consist of a mixed arrangement of a maximum of 64 analog and digital I/O modules, connected on the fieldbus coupler.

The data of the digital I/O modules are bit-oriented; i.e., digital data are sent bit by bit. The data of the analog I/O modules are byte-oriented; i.e., analog data are sent byte by byte. The term “Analog I/O modules” represents the group of byte-oriented I/O modules, which send data byte by byte. This group includes, for example, counter modules and angle and distance measurement modules.

The fieldbus coupler stores the process data in the process images. The fieldbus coupler works with a process output data image (PIO) and a process input data image (PII).

The PIO is filled of the fieldbus master with the process output data. The PII is filled of the fieldbus coupler with the process input data.

Into the input and output process image the data of the I/O modules are stored in the sequence of its position after the fieldbus coupler in the individual process image.

First, all the byte-oriented I/O modules are stored in the process image, then the bit-oriented I/O modules. The bits of the digital I/O modules are grouped into bytes. If the amount of digital I/O information exceeds 8 bits, the fieldbus coupler automatically starts a new byte.

NOTICE

Avoid equipment damages due to addressing errors!
To avoid equipment damages within the field range, you must consider that, depending on the specific position of an I/O module in the fieldbus node, the process data of all previous byte or bit-oriented modules must be taken into account to determine its location in the process data map.

Note

Consider the Process Data size for each module!
Observe the number of input and output bits or bytes for the individual I/O modules.

For some I/O modules and their different versions, the structure of the process data depends on the fieldbus.

Information

Additional information about the fieldbus specific process image
For the fieldbus-specific process image of any WAGO-I/O-Module, please refer to the section “Structure of the Process Data”.

7.3.1 General Mechanisms for EtherCAT® to Map Process Data

EtherCAT® uses Rx and TxPDO's to map process data and to write the structure the cyclically transmitted process data. The RxPDO's and the TxPDO's are units in the object dictionary in the slave.

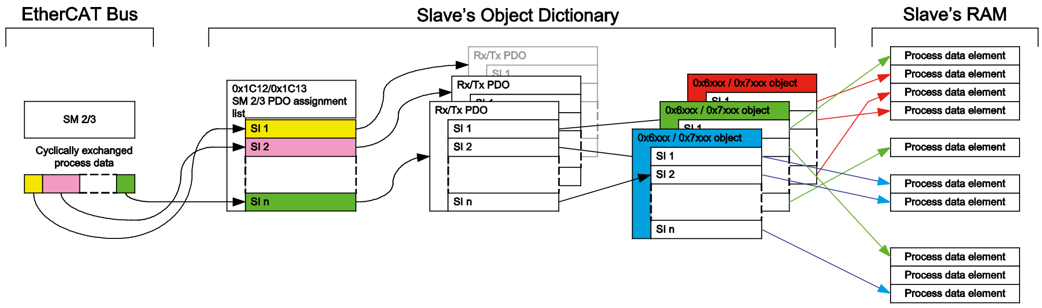


Figure 37: Mapping the Process Data

The process data are saved in the slave RAM and are represented in the input area (0x6000..0x6FFF) and output area (0x7000..0x7FFF) in the Object Dictionary in the slave by corresponding objects.
Internally, this type of object or its subindex refers solely to the location in the device where the represented data are stored. The objects and their subindices are generated and named such that they generally reflect the physical structure of the device and the user can easily understand their meaning.

Entries of objects 0x6000 and 0x7000 can in turn be referenced by so-called Rx and TxPDO's using their index and subindex. RxPDO's 0x1600 to 0x17FF are provided for output data, and TxPDO's 0x1A00 to 0x1BFF are provided for input data. A PDO therefore encapsulates several process data in a type of packet.

The Rx and TxPDO's are listed in the Sync Manager Assignment Lists 0x1C12 (output data) and 0x1C13 (input data). These lists define which Rx/TxPDO's and in which order the data referenced by the PDO's is to be cyclically transmitted.

Objects 0x6000 and 0x7000 thus specify which input and output data the slave has. The Rx/TxPDO's group the input and output data into packets and the Sync Manager Assignment Lists define which packets are cyclically transmitted in which order.

The EtherCAT® specification enables writing of the Rx/TxPDO's and the Sync Sync Manager Assignment Lists by the master, as well as read-only status. In the first case, the user can fully configure the structure of the cyclically exchanged process data.
However, for most slaves these objects are read-only and all process data must be cyclically transmitted. The master selects the objects during the slave boot-up and determines from this the exact structure of the cyclically exchanged process data. For slaves of this type, the structure of the process data in the RAM is identical with the structure of the cyclically exchanged data. In this way, the data can be directly copied internally and do not have to be resorted.

The following figure illustrates the mechanism by means of an example.
The imaginary slave has as its input data two UINT8 “data A” and “data B”, one UINT16 “data C” and two BIT4 “data D” and “data E”.

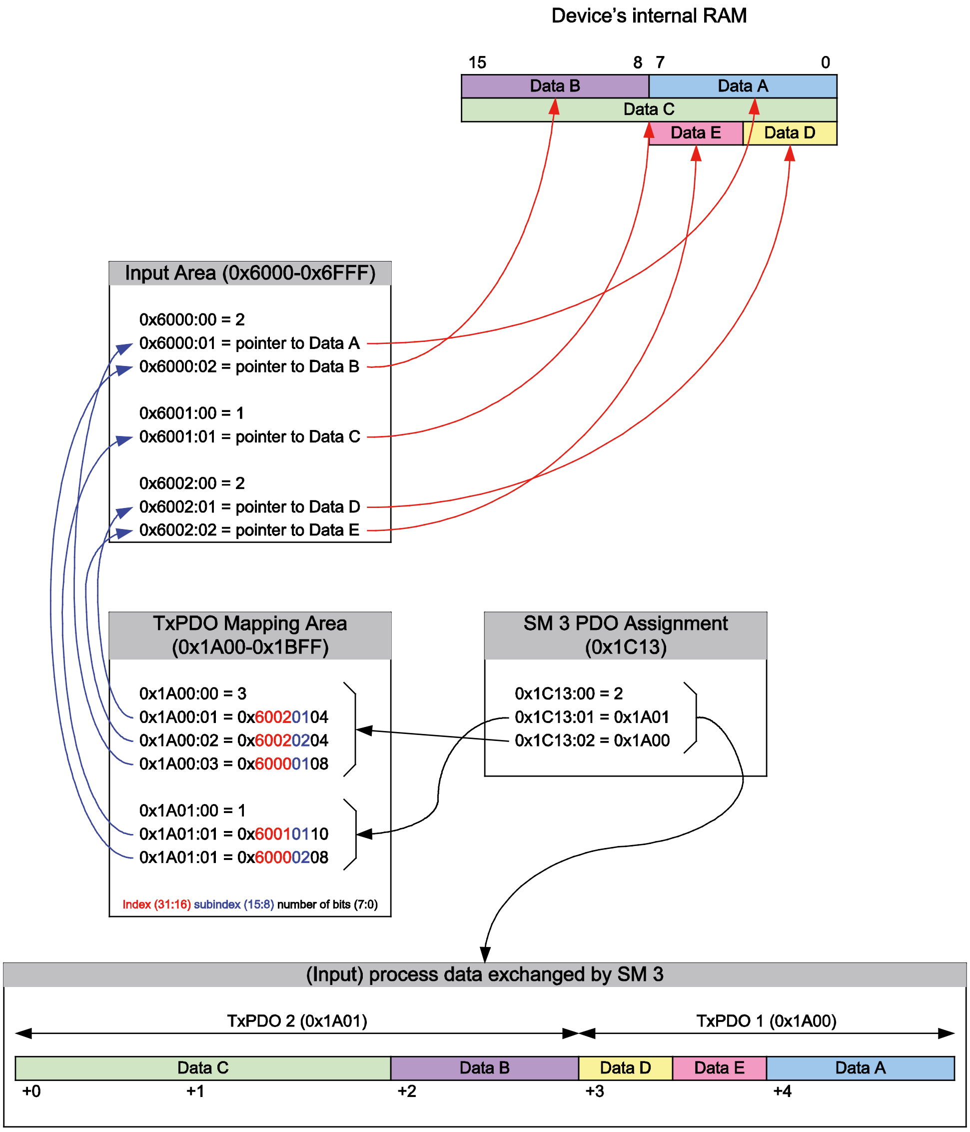


Figure 38: Example of the Principle Mapping of Process Data Using EtherCAT®

Mapping of the process data takes place using the fieldbus coupler according to the modular device profile. A special sub-profile of the modular device profile is not implemented. The mapping objects (SM Assignment Lists, Rx/TxPDO's) cannot be written.

The Modular Device Profile ensures that:

- an object 0x6000 is reserved for each I/O module with input data, and all input data of the module are represented by subindices
- an object 0x7000 is reserved for each I/O module with output data and all output data of the module are represented by subindices
- an RxPDO and/or TxPDO is reserved for each I/O module in that all input and/or output data of the I/O module are referenced
- the status data of the fieldbus coupler are in object 0xF100

- the control data for the fieldbus coupler are in object 0xF200

Information

Additional Information about the Data Structure
The structure of the cyclically exchanged data is identical with the internal structure of the process image.
You can read more about the process data architecture in the chapter "Function Description" → "Process Data Architecture".

7.3.1.1 Object 0x6000

Objects 0x6000...0x63F0 contain the input data for the I/O modules. There is exactly one object for each internal data bus with input data. The index results from the following equation:

Index = 0x6000 + (number of the I/O module - 1) ● 0x0010

If an I/O module does not have input data, then the corresponding object does not exist.

The structure of the object depends on the number of channels, the number of data per channel, and the type of data, and thus varies according to the type of I/O module. In general, an object of this type contains all input data that are supplied from the I/O module.

The following table presents the generic structure of an object 0x6000. The additional tables present the structure as a model for analog input modules 750-433 and 750-467.

Table 27: Generic Structure of Object 0x6000

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
x	Depends on IOM: BOOLEAN UINT8 UINT16 UINT32	Depends on IOM: 1 8 16 32	Depends on IOM: Channel x Status Channel x Data Byte x Word x DWord x Channel x, Byte y Channel x, Word y Channel x, DWord y
...	...	...	...

Examples

Table 28: Example Structure for Object 0x6000 for 750-433 – 4 DI

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	Channel 1 Data
2	BOOLEAN	1	Channel 2 Data
3	BOOLEAN	1	Channel 3 Data
4	BOOLEAN	1	Channel 4 Data

Table 29: Example Structure for Object 0x6000 for 750-467 – 2 AI

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1	UINT16	16	Channel 1 Data
2	UINT16	16	Channel 2 Data

7.3.1.2 Object 0x7000

Objects 0x7000...0x73F0 contain the output data for the I/O modules. This is exactly one object for each I/O module with output data. The index results from the following equation:

Index = 0x7000 + (number of the I/O module - 1) • 0x0010

If an I/O module does not have output data, then the corresponding object does not exist.

The structure of the object depends on the number of channels, the number of data per channel, and the type of data, and thus varies according to the type of I/O module. In general, an object of this type contains all input data that are supplied from the I/O module.

The following table presents the generic structure of an object 0x7000. The additional tables present the structure as a model for the analog input modules 750-504 and 750-550.

Table 30: Generic Structure of Object 0x7000

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
x	Depends on IOM: BOOLEAN UINT8 UINT16 UINT32	Depends on IOM: 1 8 16 32	Depends on IOM: Channel x Status Channel x Data Byte x Word x DWord x Channel x, Byte y Channel x, Word y Channel x, DWord y
...	...	...	...

Examples

Table 31: Example Structure for Object 0x7000 for 750-504 – 4 DO

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	Channel 1 Data
2	BOOLEAN	1	Channel 2 Data
3	BOOLEAN	1	Channel 3 Data
4	BOOLEAN	1	Channel 4 Data

Table 32: Example Structure for Object 0x7000 for 750-550 – 2 AO

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1	UINT16	16	Channel 1 Data
2	UINT16	16	Channel 2 Data

7.3.1.3 Object 0xF100

Object 0xF100 contains the status data of the fieldbus coupler according to the modular device profile. In addition, it contains the diagnostic messages for the I/O modules.

The following table presents the structure of object 0xF100.

Table 33: Structure for Object 0xF100

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	K-Bus Cycle Overrun Flag
2	BOOLEAN	1	Input Process Data Hold Acknowledge
3	BOOLEAN	1	Output Process Data Hold Acknowledge
4	BOOLEAN	1	Output Process Data Clear Acknowledge
5	UINT16	16	Diagnostics Status Word

7.3.1.4 Object 0xF200

Object 0xF200 contains the control data for the fieldbus coupler according to the modular device profile. In addition, it contains the control word for the I/O module diagnostic messages.

The following table presents the structure of object 0xF200.

Table 34: Structure of Object 0xF200

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	K-Bus Cycle Overrun Flag Disable
2	BOOLEAN	1	Input Process Data Hold Request
3	BOOLEAN	1	Output Process Data Hold Request
4	BOOLEAN	1	Output Process Data Clear Request
5	UINT16	16	Diagnostics Control Word

7.3.1.5 RxPDO and TxPDO Objects

Both an RxPDO and a TxPDO are reserved for each I/O module.

The indices for the RxPDO's are assigned to the I/O modules by the modular device profile using the following formula:

Index = 0x1600 + (number of the I/O module - 1)

If an I/O module does not have input or output data, then the corresponding Rx and/or TxPDO does not exist.

The indices for the TxPDO's are assigned to the I/O modules by the modular device profile using the following formula:

Index = 0x1A00 + (number of the I/O module - 1)

The Rx and/or TxPDO of a I/O module references in each case all entries in the objects 0x6000 and/or 0x7000 in the I/O module. If an I/O module has dummy bytes for word alignment, then these empty bits are likewise entered in the PDO by referencing object 0x000:00.

RxPDO 0x16FF references the control data for the fieldbus coupler (0xF200).
TxPDO 0x1AFF references the status data of the fieldbus coupler (0xF100).

PDO's 0x1701 and 0x1B01 are used as necessary for inserting empty bits at the end of the cyclically exchanged process data. They fill up the last 16 bit word broken up by the digital I/O modules.

The following table presents the generic structure of a Rx/TxPDO.

Table 35: Generic Structure of Rx/TxPDO (0x16xx/0x1Axx)

Sub-Index	Data Type	Number of bits	Description
0	UINT8	8	Number of Entries
1...n	UINT32	32	0xAAAABBCC A: Index of the referenced Objects B: Subindex of the referenced Objects C: Number of referenced Bits

PDO Groups

Each PDO is assigned to a PDO group. The grouping is only relevant for offline configuration.

- PDO Group 0: PDO's for the fieldbus coupler
- PDO Group 1: PDO's for complex I/O modules
- PDO Group 2: PDO's for digital I/O modules

Sync Manager Assignment Lists

The Sync Manager Assignment Lists 0x1C12 and 0x1C13 contain a list of all Rx and TxPDO's that are cyclically transmitted. The order of the indices determines the order in which the data referenced by the PDO's is transmitted.

1 PDO Group 0 is transmitted first
(PDO with the control and/or status data for the fieldbus coupler).
2 PDO Group 1 is transmitted next
(PDO's with data for the complex terminals).
3 PDO Group 2 is transmitted last
(PDO's with data for the binary terminals).
4 PDO 0x1701 and/or 0x1B01 with empty bits is transmitted at the end
(if necessary in order to achieve a 16 bit word alignment).

7.3.2 Alternative PDO Index Assignment

In some older masters, ambiguous design of the specification can cause PDOs listed in the sync manager assignment lists to be sorted in ascending order in the master according to their index. When this happens, the PDO's in the trans-mission order no longer match the actual process images. This malfunction can occur when using XML files, which only contain the description of the fieldbus coupler and no descriptions for I/O modules.

To operate the EtherCAT® Fieldbus Coupler when using defective masters of this type, allocation of the indexes for the RxPDOs and TxPDOs can be configured using object 0x2100 (PDO index assignment workaround for some masters).

Information

More Information About Object 0x2100
Object 0x2100 is described in the “Object Dictionary” section à “Object 0x2100”.

Table 36: Recommendations for Setting the 0x2100:01

		Value of object 0x2100:2		
		No XML file available	XML file with fieldbus coupler description only	XML file with fieldbus coupler and I/O module description
Processing PDO Assignment Lists in the Master	Correct	FALSE	FALSE	FALSE
	Linear (workaround needed)	TRUE	TRUE	FALSE

Note

For XML files with fieldbus coupler and I/O module descriptions, indexing according to the modular device profile!
When using the XML file that contains descriptions for fieldbus couplers and I/O modules for the configuration, always use indexing according to the module device profile (0x2100:02 = FALSE).

In the next two sections, process data mapping is present by way of example of one and the same node structure:

- Once according to the modular device profile (0x2100:02 = FALSE)
- Once with alternative indexing (0x2100:02 = TRUE)

7.3.2.1 Example According to the Modular Device Profile

This example describes process data mapping, if object 0x2100:02 has the value FALSE.

Fieldbus Node Structure

- EtherCAT® – Fieldbus Coupler
- 750-670 (Stepper Controller) – I/O Modul 1
- 750-433 (4 DI) – I/O Modul 2
- 750-504 (4 DO) – I/O Modul 3
- 750-550 (2 AO) – I/O Modul 4
- 750-476 (2 AI) – I/O Modul 5
- 750-600 (End Modul)

Process Images

Using this exemplary fieldbus node structure results in the following output and input process image represented in the fieldbus coupler.

Table 37: Example Output Process Image

Word Offset	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	Gap												Coupler Control Bits			
+1	Diagnostics Control Word															
+2	Gap								Control Byte							
+3	Byte 2								Byte 1							
+4	Byte 4								Byte 3							
+5	Byte 6								Byte 5							
+6	Byte 8								Byte 7							
+7	Byte 10								Byte 9							
+8	AO, Channel 1															
+9	AO, Channel 2															
+10	End-Gap												DO4	DO3	DO2	DO1

Legend

- Fieldbus Coupler
- 750-670, I/O Modul 1
- 750-550, I/O Modul 4
- 750-504, I/O Modul 3

Table 38: Example Input Process Image

Word Offset	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	*)												Coupler Status Bits			
+1	Diagnostics Status Word															
+2	Gap								Status Byte							
+3	Byte 2								Byte 1							
+4	Byte 4								Byte 3							
+5	Byte 6								Byte 5							
+6	Byte 8								Byte 7							
+7	Byte 10								Byte 9							
+8	AI, Channel 1															
+9	AI, Channel 2															
+10	End-Gap												DI4	DI3	DI2	DI1

*) Diag. Present

Legend

- Fieldbus Coupler
- 750-670, I/O Modul 1
- 750-476, I/O Modul 5
- 750-433, I/O Modul 2

Fieldbus Coupler Objects

The objects specified in the following are applied for the fieldbus coupler.

Table 39: Object 0xF100 (Fieldbus Coupler Status Information)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	K-Bus Cycle Overrun Flag
2	BOOLEAN	1	Input Process Data Hold Acknowledge
3	BOOLEAN	1	Output Process Data Hold Acknowledge
4	BOOLEAN	1	Output Process Data Clear Acknowledge
5	UINT16	16	Diagnostics Status Word

Table 40:Object 0xF200 (Fieldbus Coupler Control Information)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	K-Bus Cycle Overrun Flag Disable
2	BOOLEAN	1	Input Process Data Hold Request
3	BOOLEAN	1	Output Process Data Hold Request
4	BOOLEAN	1	Output Process Data Clear Request
5	UINT16	16	Diagnostics Control Word

Table 41: Object 0x16FF (Fieldbus Coupler Control PDO)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	6
1	UINT32	Coupler Control, K-Bus Cycle Overrun Flag Disable	0xF200.01.01
2	UINT32	Coupler Control, Input Process Data Hold Request	0xF200.02.01
3	UINT32	Coupler Control, Output Process Data Hold Request	0xF200.03.01
4	UINT32	Coupler Control, Output Process Data Clear Request	0xF200.04.01
5	UINT32	Gap	0x0000.00.0C
6	UINT32	Diagnostics Control Word	0xF200.05.10

Table 42: Object 0x1AFF (Fieldbus Coupler Status PDO)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	7
1	UINT32	Coupler Status, K-Bus Cycle Overrun Flag	0xF100.01.01
2	UINT32	Coupler Status, Input Process Data Hold Acknowledge	0xF100.02.01
3	UINT32	Coupler Status, Output Process Data Hold Acknowledge	0xF100.03.01
4	UINT32	Coupler Status, Output Process Data Clear Acknowledge	0xF100.04.01
5	UINT32	Gap	0x0000.00.0B
6	UINT32	Diagnostics History, new Message Available	0x10F3.04.01
7	UINT32	Diagnostics Status Word	0xF100.05.10

I/O Module Objects 750-670 (Stepper Controller)

For the first I/O module (Stepper Controller 750-670) in the fieldbus node, the following objects are created.

Table 43: Object 0x6000 (Input Data 750-670)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT8	8	Status
2	UINT8	8	Byte 1
3	UINT8	8	Byte 2
4	UINT8	8	Byte 3
5	UINT8	8	Byte 4
6	UINT8	8	Byte 5
7	UINT8	8	Byte 6
8	UINT8	8	Byte 7
9	UINT8	8	Byte 8
10	UINT8	8	Byte 9
11	UINT8	8	Byte 10

Table 44: Object 0x7000 (Output Data 750-670)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT8	8	Control
2	UINT8	8	Byte 1
3	UINT8	8	Byte 2
4	UINT8	8	Byte 3
5	UINT8	8	Byte 4
6	UINT8	8	Byte 5
7	UINT8	8	Byte 6
8	UINT8	8	Byte 7
9	UINT8	8	Byte 8
10	UINT8	8	Byte 9
11	UINT8	8	Byte 10

Table 45: Object 0x1600 (RxPDO 750-670)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	12
1	UINT32	Output Mapping Area 1	0x7000.01.08
2	UINT32	Output Mapping Area 2	0x0000.00.08
3	UINT32	Output Mapping Area 3	0x7000.02.08
4	UINT32	Output Mapping Area 4	0x7000.03.08
5	UINT32	Output Mapping Area 5	0x7000.04.08
6	UINT32	Output Mapping Area 6	0x7000.05.08
7	UINT32	Output Mapping Area 7	0x7000.06.08
8	UINT32	Output Mapping Area 8	0x7000.07.08
9	UINT32	Output Mapping Area 9	0x7000.08.08
10	UINT32	Output Mapping Area 10	0x7000.09.08
11	UINT32	Output Mapping Area 11	0x7000.0A.08
12	UINT32	Output Mapping Area 12	0x7000.0B.08

Table 46: Object 0x1A00 (TxPDO 750-670)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	12
1	UINT32	Output Mapping Area 1	0x6000.01.08
2	UINT32	Output Mapping Area 2	0x0000.00.08
3	UINT32	Output Mapping Area 3	0x6000.02.08
4	UINT32	Output Mapping Area 4	0x6000.03.08
5	UINT32	Output Mapping Area 5	0x6000.04.08
6	UINT32	Output Mapping Area 6	0x6000.05.08
7	UINT32	Output Mapping Area 7	0x6000.06.08
8	UINT32	Output Mapping Area 8	0x6000.07.08
9	UINT32	Output Mapping Area 9	0x6000.08.08
10	UINT32	Output Mapping Area 10	0x6000.09.08
11	UINT32	Output Mapping Area 11	0x6000.0A.08
12	UINT32	Output Mapping Area 12	0x6000.0B.08

I/O Module Objects 750-433 (4 DI)

For the second I/O module (4 DI, 750-433) in the fieldbus node, the following objects are created.

Table 47: Object 0x6010 (Input Data 750-433)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	Channel 1, Data
2	BOOLEAN	1	Channel 2, Data
3	BOOLEAN	1	Channel 3, Data
4	BOOLEAN	1	Channel 4, Data

Table 48: Object 0x1A01 (TxPDO 750-433)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	4
1	UINT32	Input Mapping Area 1	0x6010.01.01
2	UINT32	Input Mapping Area 2	0x6010.02.01
3	UINT32	Input Mapping Area 3	0x6010.03.01
4	UINT32	Input Mapping Area 4	0x6010.04.01

I/O Module Objects 750-504 (4 DO)

For the third I/O module (4 DO, 750-504) in the fieldbus node, the following objects are created.

Table 49: Object 0x7020 (Output Data 750-504)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	Channel 1, Data
2	BOOLEAN	1	Channel 2, Data
3	BOOLEAN	1	Channel 3, Data
4	BOOLEAN	1	Channel 4, Data

Table 50: Object 0x1602 (RxPDO 750-504)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	4
1	UINT32	Output Mapping Area 1	0x7020.01.01
2	UINT32	Output Mapping Area 2	0x7020.02.01
3	UINT32	Output Mapping Area 3	0x7020.03.01
4	UINT32	Output Mapping Area 4	0x7020.04.01

I/O Module Objects 750-550 (2 AO)

For the fourth I/O module (2 AO, 750-550) in the fieldbus node, the following objects are created.

Table 51: Object 0x7030 (Output Data 750-550)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT16	16	Channel 1, Word 1
2	UINT16	16	Channel 2, Word 1

Table 52: Object 0x1603 (RxPDO 750-550)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	2
1	UINT32	Output Mapping Area 1	0x7030.01.10
2	UINT32	Output Mapping Area 2	0x7030.02.10

I/O Module Objects 750-467 (2 AI)

For the fifth I/O module (2 AI, 750-467) in the fieldbus node, the following objects are created.

Table 53: Object 0x6040 (Input Data 750-467)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT16	16	Channel 1, Word 1
2	UINT16	16	Channel 2, Word 1

Table 54: Object 0x1A04 (TxPDO 750-467)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	2
1	UINT32	Output Mapping Area 1	0x6040.01.10
2	UINT32	Output Mapping Area 2	0x6040.02.10

Empty Bits
To fill in empty bits at the end of the input and output process image to achieve the word alignment, the PDO's specified in the following are applied.

Table 55: Object 0x1701 (RxPDO Gap at the End of the Output Process Image, Behind the DO´s)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	1
1	UINT16	RxPDO-Gap after Digital Modules	0x0000.00.0C

Table 56: Object 0x1B01 ((TxPDO Gap at the End of the Input Process Image, Behind the DI´s)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	1
1	UINT16	TxPDO-Gap after Digital Modules	0x0000.00.0C

Sync Manager Assignment Lists
The sync manager assignment lists are structured as specified below:

Table 57: Object 0x1C12 (RxPDO Assignment List for Sync Manager 2)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	5
1	UINT16	Subindex 000	0x16FF
2	UINT16	Subindex 001	0x1600
3	UINT16	Subindex 002	0x1603
4	UINT16	Subindex 003	0x1602
5	UINT16	Subindex 004	0x1701

Table 58: Object 0x1C13 (TxPDO Assignment List for Sync Manager 3)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	5
1	UINT16	Subindex 000	0x1AFF
2	UINT16	Subindex 001	0x1A00
3	UINT16	Subindex 002	0x1A04
4	UINT16	Subindex 003	0x1A01
5	UINT16	Subindex 004	0x1B01

7.3.2.2 Example According to Linear PDO Indexing

This example describes process data mapping if object 0x2100:02 has the value TRUE.

If linear PDO indexing is selected (0x2100:02 = TRUE), then the indexes for the RxPDOs and TxPDOs are assigned such that they appear in ascending order in the sync manager assignment lists (0x1C12 and 0x1C13). The structure of the process image and all objects 0x6000 and 0x7000 remains unchanged by this.

Fieldbus Node Structure

- EtherCAT® – Fieldbus Coupler
- 750-670 (Stepper Controller) – I/O Modul 1
- 750-433 (4 DI) – I/O Modul 2
- 750-504 (4 DO) – I/O Modul 3
- 750-550 (2 AO) – I/O Modul 4
- 750-476 (2 AI) – I/O Modul 5
- 750-600 (End Modul)

Process Images

Using this exemplary fieldbus node structure results in the following output and input process image represented in the fieldbus coupler.

Table 59: Example Output Process Image

Word Offset	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	Gap												Coupler Control Bits			
+1	Diagnostics Control Word															
+2	Gap								Control Byte							
+3	Byte 2								Byte 1							
+4	Byte 4								Byte 3							
+5	Byte 6								Byte 5							
+6	Byte 8								Byte 7							
+7	Byte 10								Byte 9							
+8	AO, Channel 1															
+9	AO, Channel 2															
+10	End Gap												DO4	DO3	DO2	DO1

Legend

(purple)	Fieldbus Coupler
(light blue)	750-670, I/O Modul 1
(blue)	750-550, I/O Modul 4
(red)	750-504, I/O Modul 3

Table 60: Example Input Process Image

Word Offset	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	*)												Coupler Status Bits			
+1	Diagnostics Status Word															
+2	Gap								Status Byte							
+3	Byte 2								Byte 1							
+4	Byte 4								Byte 3							
+5	Byte 6								Byte 5							
+6	Byte 8								Byte 7							
+7	Byte 10								Byte 9							
+8	AI, Channel 1															
+9	AI, Channel 2															
+10	End-Gap												DI4	DI3	DI2	DI1

*) Diag. Present

Legend

(purple)	Fieldbus Coupler
(light blue)	750-670, I/O Modul 1
(light green)	750-476, I/O Modul 5
(yellow)	750-433, I/O Modul 2

Fieldbus Coupler Objects

The objects specified in the following are applied for the fieldbus coupler.

Table 61: Object 0xF100 (Fieldbus Coupler Status Information)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	K-Bus Cycle Overrun Flag
2	BOOLEAN	1	Input Process Data Hold Acknowledge
3	BOOLEAN	1	Output Process Data Hold Acknowledge
4	BOOLEAN	1	Output Process Data Clear Acknowledge
5	UINT16	16	Diagnostics Status Word

Table 62:Object 0xF200 (Fieldbus Coupler Control Information)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	K-Bus Cycle Overrun Flag Disable
2	BOOLEAN	1	Input Process Data Hold Request
3	BOOLEAN	1	Output Process Data Hold Request
4	BOOLEAN	1	Output Process Data Clear Request
5	UINT16	16	Diagnostics Control Word

Table 63: Object 0x1600 (Fieldbus Coupler Control PDO)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	6
1	UINT32	Coupler Control, K-Bus Cycle Overrun Flag Disable	0xF200.01.01
2	UINT32	Coupler Control, Input Process Data Hold Request	0xF200.02.01
3	UINT32	Coupler Control, Output Process Data Hold Request	0xF200.03.01
4	UINT32	Coupler Control, Output Process Data Clear Request	0xF200.04.01
5	UINT32	Gap	0x0000.00.0C
6	UINT32	Diagnostics Control Word	0xF200.05.10

Table 64: Object 0x1A00 (Fieldbus Coupler Status PDO)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	7
1	UINT32	Coupler Status, K-Bus Cycle Overrun Flag	0xF100.01.01
2	UINT32	Coupler Status, Input Process Data Hold Acknowledge	0xF100.02.01
3	UINT32	Coupler Status, Output Process Data Hold Acknowledge	0xF100.03.01
4	UINT32	Coupler Status, Output Process Data Clear Acknowledge	0xF100.04.01
5	UINT32	Gap	0x0000.00.0B
6	UINT32	Diagnostics History, new Message Available	0x10F3.04.01
7	UINT32	Diagnostics Status Word	0xF100.05.10

I/O Module Objects 750-670 (Stepper Controller)

For the first I/O module (Stepper Controller 750-670) in the fieldbus node, the following objects are created.

Table 65: Object 0x6000 (Input Data 750-670)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT8	8	Status
2	UINT8	8	Byte 1
3	UINT8	8	Byte 2
4	UINT8	8	Byte 3
5	UINT8	8	Byte 4
6	UINT8	8	Byte 5
7	UINT8	8	Byte 6
8	UINT8	8	Byte 7
9	UINT8	8	Byte 8
10	UINT8	8	Byte 9
11	UINT8	8	Byte 10

Table 66: Object 0x7000 (Output Data 750-670)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT8	8	Control
2	UINT8	8	Byte 1
3	UINT8	8	Byte 2
4	UINT8	8	Byte 3
5	UINT8	8	Byte 4
6	UINT8	8	Byte 5
7	UINT8	8	Byte 6
8	UINT8	8	Byte 7
9	UINT8	8	Byte 8
10	UINT8	8	Byte 9
11	UINT8	8	Byte 10

Table 67: Object 0x1601 (RxPDO 750-670)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	12
1	UINT32	Output Mapping Area 1	0x7000.01.08
2	UINT32	Output Mapping Area 2	0x0000.00.08
3	UINT32	Output Mapping Area 3	0x7000.02.08
4	UINT32	Output Mapping Area 4	0x7000.03.08
5	UINT32	Output Mapping Area 5	0x7000.04.08
6	UINT32	Output Mapping Area 6	0x7000.05.08
7	UINT32	Output Mapping Area 7	0x7000.06.08
8	UINT32	Output Mapping Area 8	0x7000.07.08
9	UINT32	Output Mapping Area 9	0x7000.08.08
10	UINT32	Output Mapping Area 10	0x7000.09.08
11	UINT32	Output Mapping Area 11	0x7000.0A.08
12	UINT32	Output Mapping Area 12	0x7000.0B.08

Table 68: Object 0x1A01 (TxPDO 750-670)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	12
1	UINT32	Output Mapping Area 1	0x6000.01.08
2	UINT32	Output Mapping Area 2	0x0000.00.08
3	UINT32	Output Mapping Area 3	0x6000.02.08
4	UINT32	Output Mapping Area 4	0x6000.03.08
5	UINT32	Output Mapping Area 5	0x6000.04.08
6	UINT32	Output Mapping Area 6	0x6000.05.08
7	UINT32	Output Mapping Area 7	0x6000.06.08
8	UINT32	Output Mapping Area 8	0x6000.07.08
9	UINT32	Output Mapping Area 9	0x6000.08.08
10	UINT32	Output Mapping Area 10	0x6000.09.08
11	UINT32	Output Mapping Area 11	0x6000.0A.08
12	UINT32	Output Mapping Area 12	0x6000.0B.08

I/O Module Objects 750-433 (4 DI)

For the second I/O module (4 DI, 750-433) in the fieldbus node, the following objects are created.

Table 69: Object 0x6010 (Input Data 750-433)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	Channel 1, Data
2	BOOLEAN	1	Channel 2, Data
3	BOOLEAN	1	Channel 3, Data
4	BOOLEAN	1	Channel 4, Data

Table 70: Object 0x1A03 (TxPDO 750-433)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	4
1	UINT32	Input Mapping Area 1	0x6010.01.01
2	UINT32	Input Mapping Area 2	0x6010.02.01
3	UINT32	Input Mapping Area 3	0x6010.03.01
4	UINT32	Input Mapping Area 4	0x6010.04.01

I/O Module Objects 750-504 (4 DO)
For the third I/O module (4 DO, 750-504) in the fieldbus node, the following objects are created.

Table 71: Object 0x7020 (Output Data 750-504)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	BOOLEAN	1	Channel 1, Data
2	BOOLEAN	1	Channel 2, Data
3	BOOLEAN	1	Channel 3, Data
4	BOOLEAN	1	Channel 4, Data

Table 72: Object 0x1603 (RxPDO 750-504)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	4
1	UINT32	Output Mapping Area 1	0x7020.01.01
2	UINT32	Output Mapping Area 2	0x7020.02.01
3	UINT32	Output Mapping Area 3	0x7020.03.01
4	UINT32	Output Mapping Area 4	0x7020.04.01

I/O Module Objects 750-550 (2 AO)
For the fourth I/O module (2 AO, 750-550) in the fieldbus node, the following objects are created.

Table 73: Object 0x7030 (Output Data 750-550)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT16	16	Channel 1, Word 1
2	UINT16	16	Channel 2, Word 1

Table 74: Object 0x1602 (RxPDO 750-550)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	2
1	UINT32	Output Mapping Area 1	0x7030.01.10
2	UINT32	Output Mapping Area 2	0x7030.02.10

I/O Module Objects 750-467 (2 AI)

For the fifth I/O module (2 AI, 750-467) in the fieldbus node, the following objects are created.

Table 75: Object 0x6040 (Input Data 750-467)

Subindex	Data Type	Number of Bits	Description
0	UINT8	8	Number of Entries
1	UINT16	16	Channel 1, Word 1
2	UINT16	16	Channel 2, Word 1

Table 76: Object 0x1A04 (TxPDO 750-467)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	2
1	UINT32	Output Mapping Area 1	0x6040.01.10
2	UINT32	Output Mapping Area 2	0x6040.02.10

Empty Bits

To fill in empty bits at the end of the input and output process image to achieve the word alignment, the PDO's specified in the following are applied.

Table 77: Object 0x1701 (RxPDO Gap at the End of the Output Process Image)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	1
1	UINT16	RxPDO-Gap after Digital Modules	0x0000.00.0C

Table 78: Object 0x1B01 (TxPDO Gap at the End of the Input Process Image)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	1
1	UINT16	TxPDO-Gap after Digital Modules	0x0000.00.0C

Sync Manager Assignment Lists

The sync manager assignment lists are structured as specified below.

Table 79: Object 0x1C12 (RxPDO Assignment List for Sync Manager 2)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	5
1	UINT16	Subindex 000	0x1600
2	UINT16	Subindex 001	0x1601
3	UINT16	Subindex 002	0x1602
4	UINT16	Subindex 003	0x1603
5	UINT16	Subindex 004	0x1701

Table 80: Object 0x1C13 (TxPDO Assignment List for Sync Manager 3)

Subindex	Data Type	Description	Value
0	UINT8	Number of Entries	5
1	UINT16	Subindex 000	0x1A00
2	UINT16	Subindex 001	0x1A01
3	UINT16	Subindex 002	0x1A02
4	UINT16	Subindex 003	0x1A03
5	UINT16	Subindex 004	0x1B01

7.4 Object Dictionary

The EtherCAT® fieldbus coupler contains an Object Dictionary in which all available objects are entered.

Information

Additional Information about the Object Dictionary
A listing of all objects and the meaning of the entries is described in the appendix > "EtherCAT® Object Dictionary".

Certain special objects from the complete list in the appendix will be presented in more detail in the following sections.

7.4.1 Structure "Module Ident"

Some of the objects documented below use the term "Module Ident". This is defined by the modular device profile and should represent a 32-bit ID that uniquely identifies one I/O module (when possible).

Based on the history of the local bus system, a unique 32-bit ID cannot be defined for most I/O modules. All digital I/O modules, among others, are affected by this. For complex I/O modules, variants cannot be completely uniquely identified because the scope of the configuration data is too large to be included in a 32-bit ID. The distinguishing features that flow into the ID are, however, sufficient in each case to enable sensible online and offline configurations.

The "Module Ident" is structured for digital I/O modules as follows:

Table 81: Structure of "Module Ident" for Digital I/O Modules

Module Ident = 0x8aabbccd	
Code	**Explanation**
aa	An internal parameter of the module
bb	Number of diagnostics bits
cc	Number of channels
d	Input/output data: Bit 0 = Inputs present Bit 1 = Outputs present

Example of 750-418, 2-Channel Digital Input with Diagnostics
Module Ident = 0x80202023:

- 8 – digital I/O module,
- 02 – internal parameter of the module,
- 02 – two digital bits,
- 02 – two channels,
- 3 – inputs present, outputs present (acknowledgement bits)

The "Module Ident" is structured for complex I/O modules as follows:

Table 82: Structure of "Module Ident" for Complex I/O Modules

Module Ident = 0xaaaabbbb	
Code	**Explanation**
aaaa	Second part of the WAGO item number, (example: 750-aaaa for 750-0559) and bit 14 for I/O modules with diagnostics embedded in the process image
bbbb	Combination of data type (internal system parameter), number of input and output bytes in the I/O module: bbbb = (data type × 49 + number of output bytes) × 49 + number of input bytes

Example of 750-559, 4-Channel Analog Output with Diagnostics
Module Ident = 0x55943F3:

- 559 – 0x0559 (item number)
- 43F3 – Combination based on the form from the "Structure of "Module Ident" for Complex I/O Modules" table

7.4.2 Object 0x10F3 (Diagnostic History)

Fieldbus coupler diagnostic messages are recorded in object 0x10F3. At the same time, filters can be set for recording diagnostic messages and sending EMCY messages can be configured.

Note

Object 0x10F3 records only fieldbus coupler diagnoses!
Diagnostic messages for the fieldbus coupler only are recorded in object 0x10F3, not for the I/O modules.

Information

Additional information about the I/O module diagnostics
The I/O module diagnostics are described in "Diagnostics" > "I/O Module Diagnostics".

The recording of a diagnostic message is triggered by the sending of an EMCY message.
If, for example, a local bus error occurs in the fieldbus coupler, then a corresponding EMCY is sent to the fieldbus master and the content of the EMCY is simultaneously recorded in object 0x10F3 in the form of a "diagnostic entry".
The format of the header of a diagnostic entry is determined by the EtherCAT® Technology Group. Type and structure of the data in a diagnostic entry are manufacturer specific.

For the WAGO EtherCAT® fieldbus coupler, these diagnostic entries correspond to the information sent in the EMCY. The structure of the diagnostic entry is shown in the following table.

Table 83: Structure of a Diagnostic Entry in Object 0x10F3 (Diagnosis History)

Byte Offset	Data Type	Name	Description
+0 +1 +2 +3	uint32_t	diagCode	Diagnosis code used to identify the diagnosis message. Here: 0xXXXXE800 with XXXX = field „error code“ of the corresponding EMCY.
+4 +5	uint16_t	flags	Message Type: 0x0000: Info message 0x0001: Warning message 0x0002: Error message
+6 +7	uint16_t	textID	Text ID as reference to the device description file or SII/EEPROM 0 = no text < >0 = ID of a text to be displayed to the user
+8 ... +11	2 x uint32_t / uint64_t	time-stamp[0]	64 bit timestamp
+12 ... +15		time-stamp[1]	
+16 +17	uint16_t	flagsP1	Data Type of parameter #1 Here: 0x0005 = parameter #1 is UINT8
+18	uint8_t	P1	Parameter #1. Here: Value of the field “Error Register” of the corresponding EMCY.
+19 +20	uint16_t (note: packed)	flagsP2	Data Type of parameter #2 Here: 0x1005 = parameter #2 is a Byte array with 5 elements
+21 ... +25	uint8_t[5]	P2	Parameter #2. Here: Value of the field “Data” of the corresponding EMCY.

The diagnostic messages are stored in a ring buffer whose storage cells (called slots) can be read using object 0x10F3. The Diagnostic History Object operates in the so-called “Acknowledge Mode,” i.e., before a slot can be overwritten, the diagnostic message recorded in it must be acknowledged.

The Diagnosis History Object 0x10F3 is a RECORD type. The structure of this object is shown in the following table.

Table 84: Structure of the Diagnosis History Object 0x10F3.

Subindex	Data Type	R/W	Description
0	UNSIGNED8	RO	Number of subindices (= 17)
1	UNSIGNED8	RO	Number of message slots in this object (= 12)
2	UNSIGNED8	RO	Subindex of the newest logged diagnosis message (SI must be in the range 6 ... 17)
3	UNSIGNED8	RW	Subindex of the last acknowledged message
4	BOOLEAN	RO	TRUE = At least one unacknowledged message is available FALSE = No message or all messages have been acknowledged By spec, this equals the expression (SI2 != SI3)
5	UNSIGNED16	RW	Control Flags Bit 0: 1 = enable EMCY sending Bit 1: 1 = do not log info messages Bit 2: 1 = do not log warning messages Bit 3: 1 = do not log error messages
		RO	Bit 4: 1 = Acknowledge Mode (RO) Bit 5: 1 = message buffer is full and a new message will be discarded (RO)
6 ... 17	OCTETSTRING	RO	Diagnosis messages according to the table above „Structure of a Diagnostic Entry“ Little Endian!

Bit 0 … 3 can be written in Subindex 5 (abbreviation: SI5) by the master to determine which type of message should be recorded (info, warning or error messages) and whether the EMCY message should be sent to the master for each diagnostic event.
Bit 4 indicates that the diagnostic history object is in “Acknowledge Mode”. Before a diagnostic message can be overwritten, it must be acknowledged. Bit 5 is set if the ring memory is full. Bit 4 and 5 can only be read and not overwritten by the master.
During the write operations to Subindex 5, bit 4 must always be written as 1; the values for bit 5 are ignored (discarded).
If an incorrect value is written to SI5, then write access is denied with SDO abort code 0x06090030.

The message slots are located in subindices 6 to 17. These slots represent a ring memory.

Subindices 2 and 3 function like pointers to these ring memory slots and show the value of the subindex of the newest logged message, SI2, and the value of the last acknowledged message, SI3.

With each new recording and acknowledgement, the pointers are increased by one slot until, after reaching the last slot (SI17), they return again to the first slot (SI6) for the subsequent recording.

SI4 indicates whether unacknowledged diagnostic messages are present.

Note

Max. 11 unacknowledged messages are possible!
Note that unacknowledged messages cannot be overwritten.
If you have no unacknowledged messages, then after 11 entries having unacknowledged messages, all slots will be filled except for the last. In this case, an EMCY message, "History memory full", will be generated and sent to the master. This EMCY message will be recorded in the last free slot instead of a final entry for the last occurring error and SI5 bit 5 set.

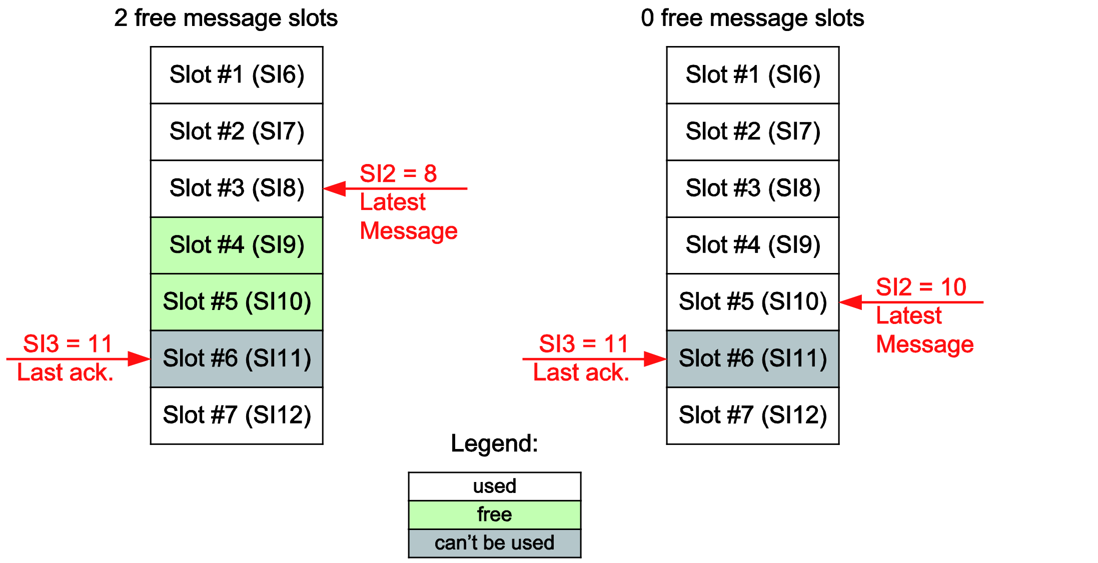


Figure 39: Number of Free Slots

Subindex 3 contains the subindex number of the last acknowledged message. If a new value is written to SI3, all messages up to the written subindex number are acknowledged.

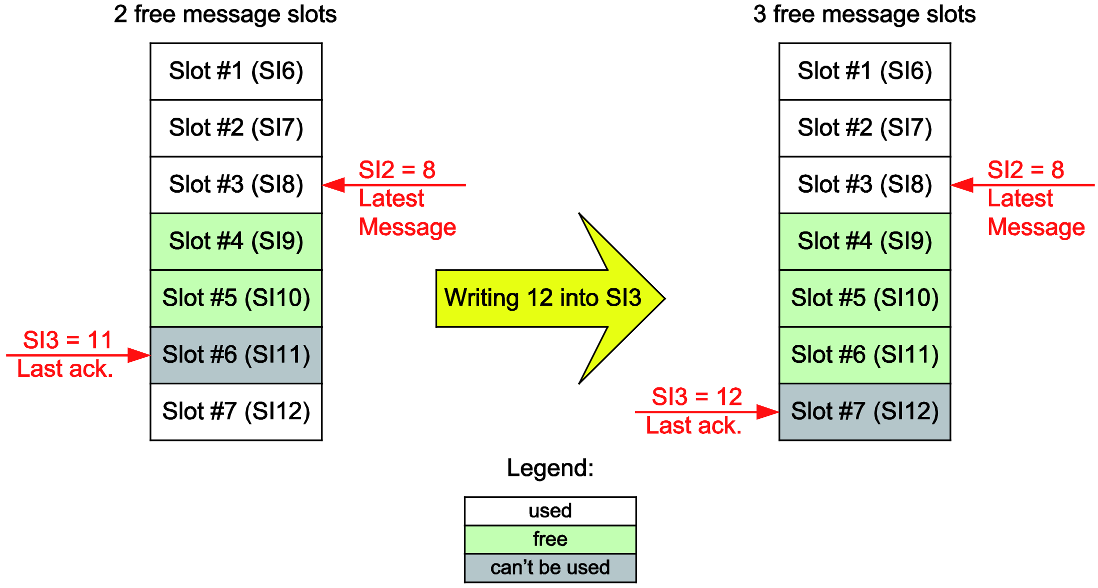


Figure 40: Acknowledgement of a Diagnostic Message

If, for example, SI3 points to the last acknowledged message in SI11, then the next unacknowledged diagnostic message, which is stored in SI12, is acknowledged by writing 12 in subindex 3.

In case a value other than the subindex number of an unacknowledged message is written in SI3, the write access is denied by SDO Abort Code 0x06090030.
In case there are no unacknowledged messages, the write attempt is likewise denied by SDO Abort Code 0x06090030.

7.4.3 Objects 0x1C32 and 0x1C33

Objects 0x1C32 and 0x1C33, shown in the following table, control which synchronization process is used by the slave to read its inputs and outputs and to exchange data with the EtherCAT® fieldbus.

The EtherCAT® fieldbus coupler synchronizes the local bus cycles to the write access on Sync Manager 2 so that a local bus cycle is initiated by the arrival of new output data.

If the EtherCAT® cycle is faster than the local bus cycle, the output data is discarded and input data transmitted multiple times unchanged. This is called a cycle time overrun or a local bus overrun.

Information

Additional information about cycle time overruns
You can read about cycle time overruns in more detail in "Function Description" > "Data Exchange" > "Control of the Process Images" > "Behavior of the Fieldbus Controller in Cycle Time Overruns".

The minimum EtherCAT® cycle time that supports the fieldbus coupler, without causing an appreciable number of local bus overruns (Assuming no CoE access and no I/O module diagnostics) is specified in each case in subindex 5 of both objects.

If new output data are written while a local bus cycle is running, then the counter in SI11 is increased. Any further writings of output data during this local bus cycle do not lead to increases in the counter. The counter is increased again only by a new overrun in the next local bus cycle.

The following figure shows this by means of an example in which the local bus cycle time is two and a half times that of the EtherCAT® cycle time.

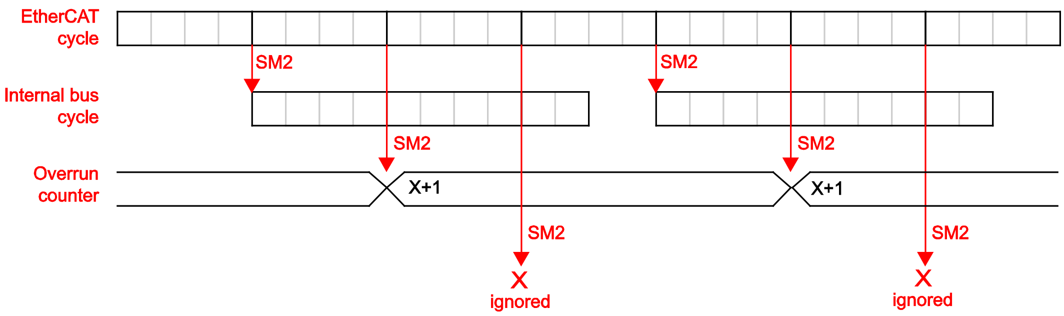


Figure 41: Overrun Counter in Subindex 11

Table 85: Structure of Sync Manager 2 Synchronization Object 0x1C32

Subindex	Data Type	Description
0	UINT8	Number of entries
1	UINT16	Synchronization procedure 0x0001: Synchronous -- synchronized with SM Event
4	UINT16	Supported synchronization procedure Bit 1: Synchronous supported (The EtherCAT®-Fieldbus coupler only supports the value 0x0002 - - Synchronous)
5	UINT32	Minimum cycle time supported by the slave in ns
11	UINT32	Overrun counter for cycle time too small events This error counter is incremented when the cycle time is too small so that the local cycle cannot be completed and input data cannot provided before the next SM event.

Table 86: Structure of the Sync Manager 3 Synchronization Object 0x1C33

Subindex	Data Type	Description
0	UINT8	Number of entries
1	UINT16	Synchronization procedure 0x22: Synchronous with SM2 Event (used when outputs are transmitted in SAFE-OP and OP)
4	UINT16	Supported synchronization procedure Bit 1: Synchronous supported (The EtherCAT® Fieldbus coupler only supports the value 0x0002 - - synchronous)
5	UINT32	Minimum cycle Same time as 0x1C32: 05
11	UINT32	Counter for cycle time too small events Same time as 0x1C32: 0B

7.4.4 Objects 0x2000 … 0x2009

Objects 0x2000 to 0x2009 can be used to read all registers of an I/O module.

Table 87: Register List Objects 0x2000 … 0x2009

Index	Name		Description
0x2000	Table fill command object		This command object is used to trigger a read sequence loading the registers of a specific terminal into the object 0x2002 … 0x2009.
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices
	1	Command	Writing the terminal number into this subindex starts the table read.
	2	Status	Command status code (see Chapter “Feldbuskommunikation” > “EtherCAT®” > “EtherCAT®-Schnittstellen” > “CoE Interface (CAN application layer over EtherCAT®)” > “Kommando Objekte”)
	3	Reply	Return code from Common Code indicating success / error.
Index	**Name**		**Description**
0x2001	Terminal Number		Number of the terminal, whose registers can be read via the objects 0x2002 … 0x2009. Zero means none.
Index	**Name**		**Description**
0x2002	Terminal Channel 1		Terminal Channel 1
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object t length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63
Index	**Name**		**Description**
0x2003	Terminal Channel 2		Channel 2 of the terminal
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63
Index	**Name**		**Description**
0x2004	Terminal Channel 3		Channel 3 of the terminal
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63
Index	**Name**		**Description**
0x2005	Terminal Channel 4		Channel 4 of the terminal
	Subindex	**Name**	**Beschreibung**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63

Table 87: Register List Objects 0x2000 … 0x2009

Index	Name		Description
0x2006	Terminal Channel 5		Channel 5 of the terminal
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63
Index	**Name**		**Description**
0x2007	Terminal Channel 6		Channel 6 of the terminal
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63
Index	**Name**		**Description**
0x2008	Terminal Channel 7		Channel 7 of the terminal
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63
Index	**Name**		**Description**
0x2009	Terminal Channel 8		Channel 8 of the terminal
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices = object length (0 or 64)
	1	Register 0	Register 0
	…	…	…
	64	Register 63	Register 63

Information

Additional Information about the Object Dictionary
A description of the objects is available in the Appendix > Section “EtherCAT® Object Dictionary”.

7.4.5 Objects 0x2010 and 0x2011

Objects 0x2010 and 0x2011 allow read/write of individual registers. The data in these objects is optimized for evaluation and processing by machine control systems.

Table 88: Single Register Access Objects 0x2010, 0x2011

Index	Name		Description
0x2010	Single Register Access Data		This object contains the data involved into a single register read/write access.
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices
	1	Terminal	Number of the terminal (1 … 64)
	2	Table	Table number (0 ... 7)
	3	Register	Register number (0 … 63)
	4	Data	Data read or data to be written (UINT16)
Index	**Name**		**Description**
0x2011	Single Register read/write command object		This command object can be used to start a read or write access to a single register.
	Subindex	**Name**	**Description**
	0	Number of entries	Number of subindices
	1	Command	Writing the ASCII code of ‘R’ (0x52) into this subindex will start a read. Writing the ASCII code of ‘W’ (0x57) into this subindex will start a write.
	2	Status	Command status code (see Chapter “Feldbuskommunikation” > “EtherCAT®” > “EtherCAT®-Schnittstellen” > “CoE Interface (CAN application layer over EtherCAT®)” > “Kommando Objekte”)
	3	Reply	Return code from Common Code indicating success / error.

Information

Additional Information about the Object Dictionary

A description of the objects is available in the Appendix > Section “EtherCAT® Object Dictionary”.

7.4.7 Object 0x2100

The alternative PDO index assignment can be switched on and off using object 0x2100.

SI2 can be used to read whether the alternative PDO index assignment is currently active.

Information

Additional information about the alternative PDO index assignment
You can read about the alternative PDO index assignment in more detail in "Function Description" > "Process Data Architecture" > "Alternative PDO Index Assignment".

Note

It is necessary to restart the system to switch the alternative PDO index assignment on or off!
Note that you will have to restart the fieldbus coupler after switching the alternative PDO index assignment on or off in object 0x2100 in order to transfer this setting. Setting the object without restarting the fieldbus coupler merely determines whether the alternative PDO index assignment will be used or not in the next startup of the firmware (power cycle or reset). It is not possible to switch the alternative PDO assignment index on or off during operation.

Table 89: PDO Index Assignment Object 0x2100

Sub-Index	Data Type	Description
0	UINT8	Number of entries
1	BOOLEAN	Control switch. Determines whether the alternative PDO index assignment should be used in the next cold boot of the coupler. TRUE = use (PDO's are numbered consecutively in the SM Assignment Lists) FALSE = do not use (Indices of the PDO's according to Modular Device Profile)
2	BOOLEAN	Status. Shows whether the alternative PDO index assignment is currently switched on. TRUE = is used now (PDO's are numbered consecutively in the SM Assignment Lists) FALSE is not currently used (Indices of the PDO's according to Modular Device Profile)

7.4.8 Object 0x2134

Object 0x2134 can be used to set the upper eight bits for the "Explicit Device ID" (EDI).

The value defined in object 0x2134 is stored as a non-volatile value.

The factory default setting is 0x00.

Table 90: Explicit Device ID (Upper 8 Bits) Object 0x2134

Sub-Index	Data type	Description
0	UNSIGNED8	Bit 8 to Bit 15 for the “Explicit Device ID”

7.4.9 Objects 0x9000 ... 0x93F0

For every I/O module there is an object in the information area (0x9000 to 0x93F0) with the Module PDO Group (SI9) and the Module Ident (SI10).

Information

Additional information about Module Ident
You can read about Module Ident in more detail in "Function Description" > "Object Dictionary" > "Structure of Module Ident".

The indices are assigned according to the following formula:

Index = 0x9000 + (number of the I/O module - 1) × 0x0010

7.4.10 Objects 0xF030 and 0xF050

Objects 0xF030 and 0xF050 concern two lists with the Module Ident's of the connected I/O modules. The two objects serve to allow the master to compare the node structure with an expected structure.

Information

Additional information about Module Ident
You can read about Module Ident in more detail in "Function Description" > "Object Dictionary" > "Structure of Module Ident".

The master has two opportunities to check the node structure.

Using object 0xF050, the master can read the structure of the local bus node. SI0 supplies the number of inserted I/O modules and the remaining subindices supply the Module Ident's of the individual I/O modules. The master can read these and compare them to a set list.

Alternatively, the master can write the number of expected I/O modules in SI0 from object 0xF030 and finally write the expected Module Ident's in the remaining subindices from object 0xF030. In case of a difference, the write access is denied by the fieldbus coupler using the SDO Abort Code 0x8000020.

7.5 Data Exchange

The cyclic exchange of process data takes place using the EtherCAT® protocol.

EtherCAT® works according to the master/slave principle. The EtherCAT® master controller can be a PC or a PLC.

The fieldbus coupler is a slave device.

The fieldbus coupler is essentially equipped with two interfaces for cyclical exchange of process data:

- interface to the fieldbus (fieldbus master)
- interface to the I/O modules.

The exchange of process data takes place between the EtherCAT® master and the I/O modules via the fieldbus coupler. In a process data cycle, the EtherCAT® master sends the output process data to the fieldbus coupler and receives input process data from the fieldbus coupler.

The process image of the fieldbus coupler comprises a memory area with a size of 1024 bytes or 512 words.

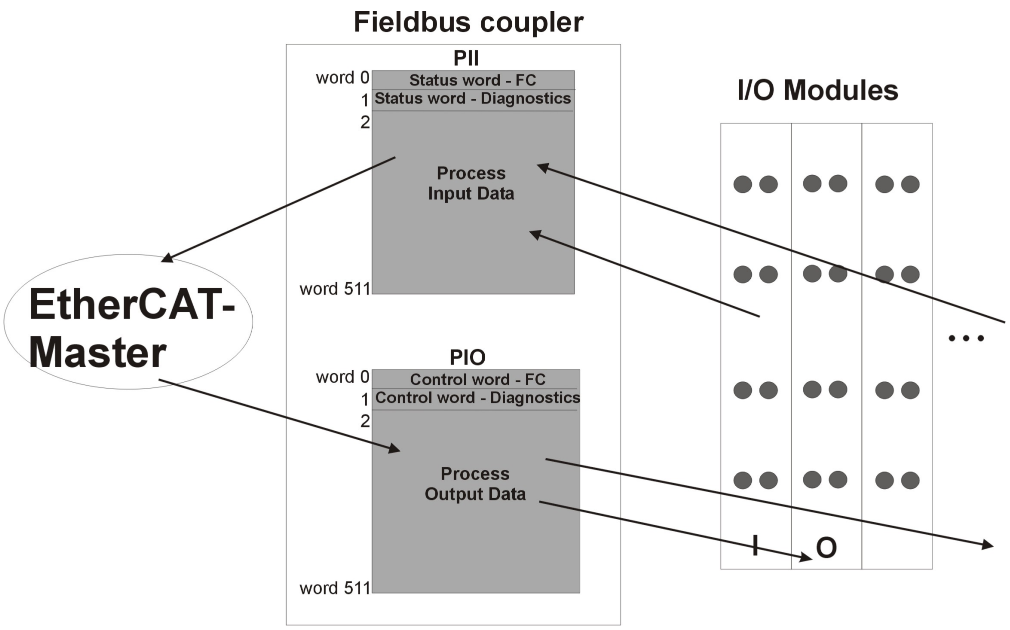


Figure 42: Memory Areas and Data Exchange of the Fieldbus Coupler

The fieldbus coupler records the input process data in the input process image (PAE). The output process data are stored by the fieldbus coupler in the output process image (PAA).

The fieldbus coupler reads the process data received from the EtherCAT® master from the PAA. The fieldbus coupler forwards the output data via the I/O module interface to the I/O modules.

The fieldbus coupler writes the input process data received from the I/O modules into the PAE. The input process data are sent in the next fieldbus cycle from the fieldbus coupler to the EtherCAT® master.

7.5.1 Arrangement of the Process Data in the PAA/PAE

The input and output data from the I/O modules are arranged by the fieldbus in the process images according to a fixed schematic. The physical arrangement of the I/O modules on a fieldbus node is arbitrary. The order in which the data of the connected I/O modules is arranged depends on the type of I/O module (input module, output module, etc).

Initially the byte-oriented I/O modules corresponding to the insertion order behind the fieldbus coupler/controller are considered in the process images. The process data from these I/O modules are therefore arranged beginning with word 2 in the process images.

Then the data from the digital I/O modules is processed. This follows the insertion order, whereby byte by byte is filled with data. As soon as a complete byte is occupied by the process data of a bit-oriented I/O module, the process begins automatically with the next byte.

The layout of the fieldbus coupler's process images is generated depending on the I/O module configuration. The fieldbus coupler arranges the control and status information in front of the process data in the PAA and/or PAE.

The following figure shows the arrangement of the process data and also the arrangement of the control and status information in the PAA and/or PAE.

The number of words for output process data and input process data is equal in the example presented, therefore the structure of the input and output data is symmetrical. When the number of words for output and input process data is different, then the structure of the process data is correspondingly asymmetrical.

Table 91: Process Data Architecture -- Example

Word Offset	Output Process Image	Input Process Image	
0	Fieldbus coupler control word	Fieldbus coupler status word	
1	Diagnostic control word	Diagnostic status word	
2	Output process data for the first byte-oriented I/O module	Input process data for the first byte-oriented I/O module	N = process data size for the first byte-oriented I/O module
2+N	Output process data for the second byte-oriented I/O module	Input process data for the second byte-oriented I/O module	M = process data size for the second byte-oriented I/O module
2+N+M	Output process data for additional byte-oriented I/O modules	Input process data for additional byte-oriented I/O modules	K = process data size for all byte-oriented I/O modules
2+K	Output process data for the first bit-oriented I/O module	Input process data for the first bit-oriented I/O module	The word is filled in with the data of subsequent bit-oriented I/O modules if the process data size of the first bit-oriented I/O module is smaller than 16 bits.
2+K+1 .... 2+K+X	Output process data for additional bit-oriented I/O modules	Input process data for additional bit-oriented I/O modules	X = process data size for all bit-oriented I/O modules

7.5.2 Determining the Process Data Arrangement Using the EtherCAT® Master

The EtherCAT® master must enter the input and output process data corresponding to the arrangement described above in the process images.

The EtherCAT® master determines the arrangement of the process data in the process images by reading from the objects. These objects describe the process data arrangement.

Table 92: Process Data Objects

Object to be read	Description
0x1C12 (RxPDO Assignment)	Assignment between the output process data and the I/O modules
0x1C13 (TxPDO Assignment)	Assignment between the input process data and the I/O modules

Objects 0x1C12 and 0x1C13 provide information in their subindices about the arrangement and size of the process data from the connected I/O modules.

Note

Hardware changes can result in changes of the process image!
If the hardware configuration is changed by adding, changing, removing or reparametrisation of I/O modules with a data width > 0 bit, this result in a new process image structure. The process data addresses would then change. If adding I/O modules, the process data of all previous I/O modules has to be taken into account.

Note

Consider the Process Data size for each module!
Observe the number of input and output bits or bytes for the individual I/O modules.

7.5.3 Controlling the Process Images

Using the fieldbus coupler control word, the EtherCAT® master can influence the content of the output process data exported to the I/O modules and also the input process data received from the I/O modules. Control commands that are activated in the fieldbus coupler control word are acknowledged by the fieldbus coupler in the fieldbus coupler status word.

7.5.3.1 Fieldbus Coupler Control Word

Table 93: Fieldbus Coupler Control Word

Fieldbus Coupler Control Word												Reset Value: 0x0000			
High-byte								**Low-byte**							
D15	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
0	0	0	0	0	0	0	0	0	0	0	0	OUTDIS	FREEZE	SYNC	OVRDIS

Bit	**Function**
D15 ... D4	This bit field must be set by the EtherCAT® master to "0".
OUTDIS	This bit controls the transmission of the output **process** data received by the EtherCAT® master to the I/O modules. 0: The output **process** data are transmitted corresponding to the status of the EtherCAT® system to the I/O modules. 1: The output **process** data are set to the value "0". The status of the SYNC bits is not followed when OUTDIS is set to "1".
FREEZE	This bit controls the transmission of input **process** data read from the I/O modules to the EtherCAT® master. 0: The input data is sent from the I/O modules to the EtherCAT® master. 1: The input data are maintained in the status initially read from the I/O modules. The input data are maintained from the time point at which the fieldbus coupler recognized that the FREEZE bit had the status "1".
SYNC	This bit controls the transmission of output process data from the EtherCAT® master to the I/O modules. 0: The output **process** data are transmitted corresponding to the status of the EtherCAT® system to the I/O modules. 1: The output data are maintained in the status initially read from the EtherCAT® master. The output data are maintained from the time point at which the fieldbus coupler recognized that the SYNC bit had the status "1". The fieldbus coupler discards output data sent by the EtherCAT® master while the SYNC bit is switched to "1".
OVRDIS	This bit controls the indication of the cycle time overrun. 0: The OVRRUN bit in the fieldbus coupler status word is set during a cycle time overrun. 1: The OVRRUN bit in the fieldbus coupler status word is not set during a cycle time overrun. Cycle time overruns are not displayed if OVRDIS is set to “1”.

7.5.3.2 Fieldbus Coupler Status Word

Table 94: Fieldbus Coupler Status Word

Fieldbus Coupler Status Word												Reset Value: 0x0000			
High-byte								Low-byte							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
DIAG_PRESENT	X	X	X	X	X	X	X	X	X	X	X	OUTDISACK	FREEZEACK	SYNCACK	OVRRUN

Bit	Function
DIAG_PRESENT	This bit indicates the occurrence of a diagnostic event in object 0x10F3. 0: There is no diagnostic event. 1: Object 0x10F3 contains at least one diagnostic event.
D14 ... D4	This bit field may not be evaluated by the EtherCAT® master.
OUTDISACK	This bit is the fieldbus coupler's acknowledgement for the OUTDIS bit in the fieldbus coupler control word. 0: The EtherCAT® master's output process data have been forwarded to the I/O modules. 1: The fieldbus coupler recognized the OUTDIS request and maintains the I/O module's output process data at the value of "0".
FREEZEACK	This bit is the fieldbus coupler's acknowledgement for the FREEZE bit in the fieldbus coupler control word. 0: The I/O module's input process data have been forwarded to the EtherCAT® master. 1: The fieldbus coupler has recognized the FREEZE request and maintains the input process data at the last value read from the I/O modules.
SYNCACK	This bit is the fieldbus coupler's acknowledgement for the SYNC but in the fieldbus coupler control word. 0: The EtherCAT® master's output process data have been forwarded to the I/O modules. 1: The fieldbus coupler has recognized the SYNC request and maintains the output process data at the last value received from the EtherCAT® master.
OVRRUN	This bit reports a cycle time overrun to the EtherCAT® master. A cycle time overrun occurs when the fieldbus coupler cannot end the internal processing of the process data before an additional process data cycle is implemented by the EtherCAT® master. The bit is set for at least 1000 ms. This time is restarted with every initiation of a cycle time overrun. 0: The internal process data processing could be completed within an EtherCAT® process data cycle or the reporting of the cycle time overrun is switched off. 1: The fieldbus couplers internal process data processing could not be completed within an EtherCAT® process data cycle.

7.5.3.3 Behaviour of the Fieldbus Coupler During a Cycle Time Overrun

The process data exchange is implemented by the EtherCAT® master in a specific temporal pattern (EtherCAT® cycle time). The fieldbus coupler receives requests to process the process data from the EtherCAT® switch in this temporal pattern. Upon receipt of a request, the fieldbus coupler implements a communication cycle with the I/O modules. The output process data are transmitted to the I/O modules and the input process data are read from the I/O modules in this communication cycle. The duration of the communication cycle is determined by the number and type of the I/O modules.

A cycle time overrun occurs when the EtherCAT® master requests an additional process data exchange before the fieldbus coupler has completed the communication cycle with the I/O modules. The fieldbus coupler behaves as follows during a cycle time overrun:

- The output process data sent from the EtherCAT® master are discarded.
- The input process data from the previous process data exchange are resent to the EtherCAT® master.
- The fieldbus coupler signals the cycle time overrun in the fieldbus coupler status word.
- The communication cycle between the fieldbus coupler and the I/O modules is reprocessed.

After completion of the communication cycle between the fieldbus coupler and the I/O modules, the fieldbus coupler is again ready to process requests for process data exchange.

Note

Pay attention when setting the EtherCAT® cycle time!
Ensure that the EtherCAT® cycle time is set in the EtherCAT® master such that no cycle time overruns occur for the fieldbus coupler.

For some I/O modules and their different versions, the structure of the process data depends on the fieldbus.

Information

Additional information about the fieldbus specific process image
For the fieldbus-specific process image of any WAGO-I/O-Module, please refer to the section “Structure of the Process Data”.

7.6 Hot Connect

Hot Connect is a feature of the EtherCAT® technology that allows you to connect preconfigured areas of the network – so-called Hot Connect groups – to the EtherCAT® network and to disconnect without any effect on existing communication from the master to other slaves. Hot Connect works both before starting and during the operation. A “Hot Connect” group can consist of multiple slave or a single slave.

Although star topology is best suited for this functionality, Hot Connect can be used in any network topology.

The advantages of the applications include:

- Connecting or disconnecting slaves or slave groups (drives, I/O modules, sensors, position sensors) in a modular production line
- High availability of the network in the event of a subscriber or group failure
- Protection against transposing identical slaves (e.g., drives)

The slave must have an “Explicit Device ID” to be Hot Connect capable.
The ID is set locally on the slave and allows unique identification of the slave in the network. Identification of a slave via the identify object (0x1018) is not enough to distinguish the same item from one manufacture in the network if there are multiple slaves of the same type or from the same manufacture in the network.

Note

Difference between “Explicit Device ID” and “Physical Addressing”
Identification via the “Explicit Device ID” should not be confused with slave addressing. And EtherCAT® master never uses the “Explicit Device ID” for physically addressing the slave.
Some masters can use “Configured Station Alias” for physical addressing. In such case, the “Explicit Device ID” can be used as the physical address via the “Configured Station Alias” method.

There are currently four methods by which the “Explicit Device ID” can be read by the master:

- “Requesting ID” method
- “Direct ID” method
- “Configured Station Alias” method
- “Legacy Mode” method

However, the “Legacy Mode” method is no longer recommended due to its error rate.

The name of the respective method can vary from master to master.
How these methods work is described in detail in the EtherCAT® specification (ETG.1020).

Which method the respective EtherCAT® slave supports is defined in the ESI file.
WAGO slaves support the “Requesting ID” method recommended by ETG.
What the method is actually called in the respective master is specified in the

documentation for your master (for TwinCAT®, the method is called e.g., ADO – “Explicit Device Identification (ADO 0x0134)”).

Note

Select the correct method!
Make sure that the right method is selected for your master. How to select the right method is described in the manual for the respective master.

The “Explicit Device ID” value is determined at the WAGO slave using the ID selector switch and object 0x2134. How to set the right value is described in Section “Device Description” > … > “Object 0x2134”.

Note

Compare WAGO slave identification to “Explicit Device ID” value!
Make sure that the WAGO slave ID set in the master matches the “Explicit Device ID” of your slave.

8 Commissioning

This section shows a step-by-step procedure for starting up exemplarily a WAGO fieldbus node.

Note

Exemplary Example!
This description is exemplary and is limited here to the execution of a local start-up of one individual fieldbus node with a non-interlaced computer running Windows.

For start-up, three steps are necessary. The description of these work steps can be found in the corresponding following sections:

- **Connecting Client PC and Fieldbus Nodes**
- **Load XML-File in the Master**
- **Testing the Function of the EtherCAT® Fieldbus Node**

8.1 Connecting Client PC and Fieldbus Nodes

1. Mount the fieldbus node on the TS 35 carrier rail.
   Follow the mounting instructions found in the "Mounting" section.

2. Connect the 24 V power supply to the supply terminals.

3. Connect the PC's ETHERNET interface to the fieldbus coupler's ETHERNET interface (RJ-45).

4. Turn the operating voltage on.

The fieldbus coupler is initialized. The coupler determines the I/O module configuration and creates a process image.
During start-up, the I/O LED (red) flashes.
If the I/O LED lights up green after a brief period, the fieldbus coupler is operational.

8.2 Load XML-File in the Master

1. Download the EtherCAT® device description file (XML file) for the EtherCAT® fieldbus coupler from WAGO's homepage on the Internet.

Information

Additional information about the EtherCAT® XML file.
The Device Description File "ESI-Dateien für EtherCAT Koppler / Serie 750, 753” is available for download from the WAGO homepage **www.wago.com** by Search Term: **“ESI”**

2. Then load the XML file into your master.
   Depending on your master, there are a two different possibilities. Therefore proceed accordingly:
   a) Either copy the file into the directory provided for ii in your master and then restart your master.
   b) Or use the import function in your master.

The XML file contains basic settings for the device to enable mailbox communication. In addition, it lists all pluggable I/O modules from the Series 750.

In case you would like to implement a purely online configuration, the XML file is then completely optional insofar as your master is able to boot devices based on the information in the SII EEPROM.

8.3 Testing the Function of the EtherCAT® Fieldbus Node

1. Prompt the master to search for new devices.
   After the search, the master should have recognized the EtherCAT® fieldbus coupler and determined the process data architecture.
   According to the master you are using, you should be presented with a list or a tree with the inserted I/O modules and their input and output data or Rx and TxPDO's.

2. Start the cyclical process data exchange at the master and switch the fieldbus coupler to OP status.
   The RUN LED on the fieldbus coupler should now light continuously green.
   The master will now exchange cyclical process data with the fieldbus coupler.

3. Select the output data or the corresponding RxPDO from a specific I/O module (preferably a digital bus output module) and manually write a value to an output.
   (Use the documentation associated with your master to learn how to manually write a value to an output.)
   The corresponding I/O module should now signal a change in the signal status by means of the LED display.
   Alternately, you can use a multimeter to measure the output signals.

8.4 Testing I/O Modules using WAGO-I/O-*CHECK*

The WAGO-I/O-*CHECK* commissioning tool can be used to read (Monitor Mode) and write (Control Mode) process data. In addition, configuration/parameterization is also possible for certain I/O modules.

For WAGO-I/O-*CHECK*, the maximum process image size of the total node is 512 bytes. At that size, the process image is displayed correctly in WAGO-I/O-*CHECK* and the process data of the I/O modules can be read and written.

Note

System-related Limitation of the Control Mode
If the entire node exceeds the process image size of 512 bytes, access via WAGO-I/O-*CHECK* to the I/O modules is no longer possible. In that case, WAGO-I/O-*CHECK* outputs an error message.

9 Diagnostics

The fieldbus coupler supplies diagnostic information about:

- The communication between the fieldbus coupler and the EtherCAT® master
- The communication between the fieldbus coupler and the I/O modules
- The diagnostic status of the I/O modules.

9.1 Fieldbus Coupler Diagnostics

9.1.1 LED Signaling

For on-site diagnostics, the fieldbus coupler has several LEDs that indicate the operational status of the fieldbus coupler or the entire node (see following figure).

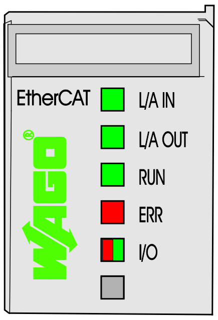


Figure 43: Display Elements

The diagnostics displays and their significance are explained in detail in the following section.

The LEDs are assigned in groups to the various diagnostics areas:

Table 95: LED Assignment for Diagnostics

Diagnostics area	LEDs
Fieldbus status	• L/A IN • L/A OUT • RUN • ERR
Node status	• I/O

9.1.2 Evaluating Fieldbus Status

The status of the fieldbus is signaled through the top LED group ('L/A IN', 'L/A OUT', 'RUN' and 'ERR').

The light emitting diode labeled "L/A IN" is assigned to interface X1. The light emitting diode labeled "L/A OUT" is assigned to interface X2. The light emitting diodes "RUN" and "ERR" display general information about the status of the EtherCAT® slave.

Table 96: LED L/A IN, L/A OUT

LED Status	Meaning
Off	No connection for the respective interface
On	There is a connection for the respective interface. No data is being exchanged via this interface.
Flashing	There is a connection for the respective interface. Data is being exchanged via this interface.

Table 97: LED RUN

LED Status	Meaning
Off	The fieldbus coupler is in the INIT state.
Flashing (200 ms on / 200 ms off)	The fieldbus coupler is in the PREOP state.
Flashing (200 ms on / 1000 ms off)	The fieldbus coupler is in the SAFEOP state.
On	The fieldbus coupler is in the OP state.

Table 98: LED ERR

LED Status	Meaning
Off	The fieldbus node is ready for operation.
Flashing (200 ms on / 200 ms off)	The EtherCAT® master could not activate the configuration of the fieldbus node.
Flashing (200 ms on / 1000 ms off)	The fieldbus coupler has changed its EtherCAT® state due to an execution error.
Flashing (200 ms on / 200 ms off / 200 ms on / 1000 ms off)	No process data was exchanged between the EtherCAT® master and fieldbus node within the monitoring time of 100 ms.

9.1.3 Evaluating Node Status – I/O LED (Blink Code Table)

The communication status between fieldbus coupler/controller and the I/O modules is indicated by the I/O LED.

Table 99: Node Status Diagnostics – Solution in Event of Error

LED Status	Meaning	Solution
I/O		
green	The fieldbus node is operating correctly.	Normal operation.
orange flashing	Start of the firmware. 1 … 2 seconds of rapid flashing indicates start-up.	-
red	Fieldbus coupler/controller hardware defect	Replace the fieldbus coupler/controller.
red flashing	Flashing with approx. 10 Hz indicates the initialization of the local bus or a local bus error.	Note the following flashing sequence.
red cyclical flashing	Up to three successive flashing sequences indicate local bus errors. There are short intervals between the sequences.	Evaluate the flashing sequences based on the following blink code table. The blinking indicates an error message comprised of an error code and error argument.
off	No data cycle on the local bus.	The fieldbus coupler/controller supply is off.

Device boot-up occurs after turning on the power supply. The I/O LED flashes orange.

Then the local bus is initialized. This is indicated by flashing red at 10 Hz for 1 … 2 seconds.

After a trouble-free initialization, the I/O LED is green.

In the event of an error, the I/O LED continues to blink red. Blink codes indicate detailed error messages. An error is indicated cyclically by up to 3 flashing sequences.

After elimination of the error, restart the node by turning the power supply of the device off and on again.

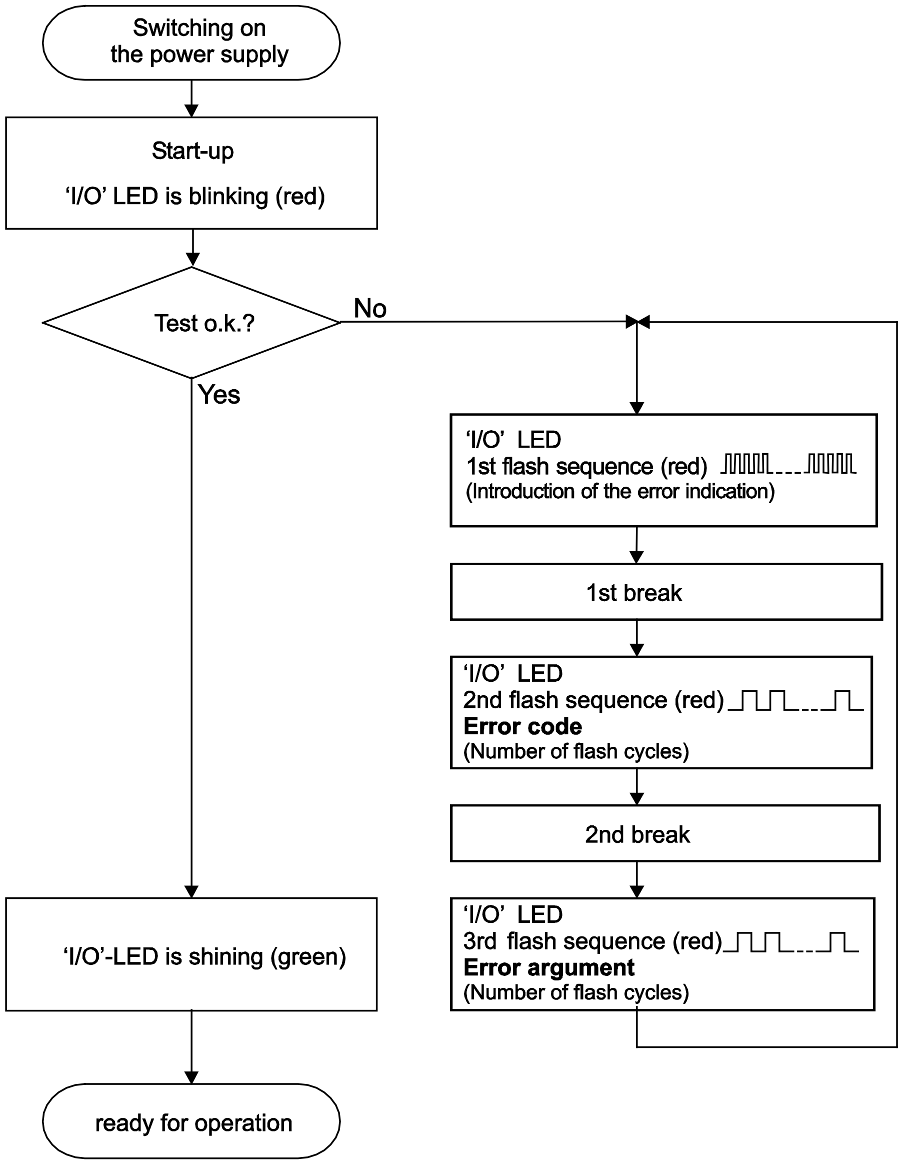


Figure 44: Node Status – I/O LED Signaling

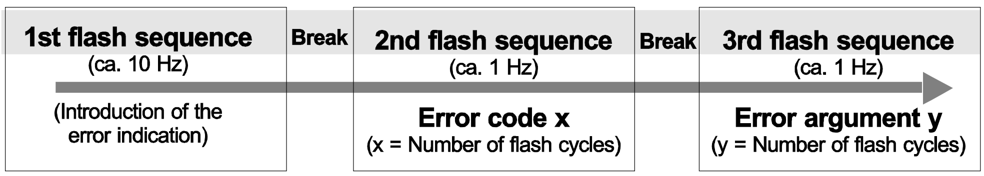


Figure 45: Error Message Coding

Example of a module error:

- The I/O LED starts the error display with the first flashing sequence (approx. 10 Hz).
- After the first break, the second flashing sequence starts (approx. 1 Hz):
  The I/O LED blinks four times.
  Error code 4 indicates “data error internal data bus”.
- After the second break, the third flashing sequence starts (approx. 1 Hz):
  The I/O LED blinks twelve times.
  Error argument 12 means that the local bus is interrupted behind the twelfth I/O module.

The thirteenth I/O module is either defective or has been pulled out of the assembly.

Table 100: Blink Code Table for the I/O LED Signaling, Error Code 1

Error code 1: "Hardware and configuration error"		
Error argument	**Error description**	**Solution**
1	Overflow of the internal buffer memory for the inline code	1. Turn off the power for the node. 2. Reduce the number of I/O modules and turn the power supply on again. 3. If the error remains, replace the fieldbus coupler.
2	I/O module(s) with unknown data type	1. Determine the faulty I/O module by first turning off the power supply. 2. Plug the end module into the middle of the node. 3. Turn the power supply on again. 4. – LED continues to flash? – Turn off the power supply and plug the end module into the middle of the first half of the node (toward the fieldbus coupler). – LED not flashing? – Turn off the power and plug the end module into the middle of the second half of the node (away from the fieldbus coupler). 5. Turn the power supply on again. 6. Repeat the procedure described in step 4 while halving the step size until the faulty I/O module is detected. 7. Replace the faulty I/O module. 8. Inquire about a firmware update for the fieldbus coupler.
3	Invalid check sum in the parameter area of the fieldbus coupler	1. Turn off the power for the node. 2. Replace the fieldbus coupler. 3. Turn the power supply on again.
4	Fault when writing in the serial EEPROM	1. Turn off the power for the node. 2. Replace the fieldbus coupler. 3. Turn the power supply on again.
5	Fault when reading the serial EEPROM	1. Turn off the power for the node. 2. Replace the fieldbus coupler. 3. Turn the power supply on again.
6	Changed I/O module configuration found after AUTORESET	1. Restart the fieldbus coupler by turning the power supply off and on.
8	Timeout during serial EEPROM access.	1. Turn off the power for the node. 2. Replace the fieldbus coupler. 3. Turn the power supply on again.
9	Fieldbus coupler initialization error	1. Turn off the power for the node. 2. Replace the fieldbus coupler. 3. Turn the power supply on again.
14	Maximum number of gateway modules or mailbox modules exceeded.	1. Turn off the power for the node. 2. Reduce the number of correspondent modules to a valid number.

Table 101: Blink Code Table for the I/O LED Signaling, Error Code 2

Error code 2: -not used-		
Error argument	**Error description**	**Solution**
-	not used	-

Table 102: Blink Code Table for the I/O LED Signaling, Error Code 3

Error code 3: " Protocol error, local bus"		
Error argument	**Error description**	**Solution**
-	Local bus communication is faulty, defective module cannot be identified.	- Are passive power supply modules (750-613) located in the node? - 1. Check that these modules are supplied correctly with power. 2. Determine this by the state of the associated status LEDs. - Are all modules connected correctly or are there any 750-613 Modules in the node? - 1. Determine the faulty I/O module by turning off the power supply. 2. Plug the end module into the middle of the node. 3. Turn the power supply on again. 4. - LED continues to flash? - Turn off the power supply and plug the end module into the middle of the first half of the node (toward the fieldbus coupler). - LED not flashing? - Turn off the power and plug the end module into the middle of the second half of the node (away from the fieldbus coupler). 5. Turn the power supply on again. 6. Repeat the procedure described in step 4 while halving the step size until the faulty I/O module is detected. 7. Replace the faulty I/O module. 8. Inquire about a firmware update for the fieldbus coupler.

Table 103: Blink Code Table for the I/O LED Signaling, Error Code 4

Error code 4: "Physical error, local bus"		
Error argument	**Error description**	**Solution**
-	Local bus data transmission error or interruption of the local bus at the fieldbus coupler	1. Turn off the power supply to the node. 2. Plug in an end module behind the fieldbus coupler. 3. Turn the power supply on. 4. Observe the error argument signaled. - Is no error argument indicated by the I/O LED? - 5. Replace the fieldbus coupler. - Is an error argument indicated by the I/O LED? - 5. Identify the faulty I/O module by turning off the power supply. 6. Plug the end module into the middle of the node. 7. Turn the power supply on again. 8. - LED continues to flash? - Turn off the power and plug the end module into the middle of the first half of the node (toward the fieldbus coupler). - LED not flashing? - Turn off the power and plug the end module into the middle of the second half of the node (away from the fieldbus coupler). 9. Turn the power supply on again. 10. Repeat the procedure described in step 6 while halving the step size until the faulty I/O module is detected. 11. Replace the faulty I/O module. 12. If there is only one I/O module on the fieldbus coupler and the LED is flashing, either the I/O module or fieldbus coupler is defective. Replace the defective component.
n*)	Interruption of the local bus behind the n^{th} bus module with process data	1. Turn off the power supply to the node. 2. Replace the (n+1) I/O module containing process data. 3. Turn the power supply on.

*) The number of light pulses (n) indicates the position of the I/O module.
I/O modules without data are not counted (e.g., supply modules without diagnostics)

Table 104: Blink Code Table for the I/O LED Signaling, Error Code 5

Error code 5: "Initialization error, internal bus"		
Error argument	**Error description**	**Solution**
n*)	Error in register communication during local bus initialization	1. Turn off the power supply to the node. 2. Replace the (n+1) I/O module containing process data. 3. Turn the power supply on.

*) The number of light pulses (n) indicates the position of the I/O module.
I/O modules without data are not counted (e.g., supply modules without diagnostics)

Table 105: Blink Code Table for the I/O LED Signaling, Error Code 6

Error code 6: "Fieldbus specific errors"		
Error argument	**Error description**	**Solution**
1	Read error EEPROM fieldbus ASIC	1. Restart the fieldbus coupler by turning the power supply off and on again. 2. If the error still exists, replace the fieldbus coupler.
2	Write error EEPROM fieldbus ASIC	1. Restart the fieldbus coupler by turning the power supply off and on again. 2. If the error still exists, replace the fieldbus coupler.
3	Timeout when writing to the EEPROM fieldbus ASIC	1. Restart the fieldbus coupler by turning the power supply off and on again. 2. If the error still exists, replace the fieldbus coupler.
4	Checksum error in the settings for the fieldbus ASIC	1. Inquire about a firmware update for the fieldbus controller. 2. If the error still exists, replace the fieldbus coupler.
5	Execution error when starting the fieldbus stack	1. Inquire about a firmware update for the fieldbus controller. 2. If the error still exists, replace the fieldbus coupler.
6	Timeout when resetting the fieldbus ASIC	1. Restart the fieldbus coupler by turning the power supply off and on again. 2. If the error still exists, replace the fieldbus coupler.
7	Supply voltages of the fieldbus connection out of range	1. Restart the fieldbus coupler by turning the power supply off and on again. 2. If the error still exists, replace the fieldbus coupler.
8	Internal execution error when activating time monitoring functions	1. Inquire about a firmware update for the fieldbus controller. 2. If the error still exists, replace the fieldbus coupler.
9	Internal execution error when enabling time monitoring functions	1. Inquire about a firmware update for the fieldbus controller. 2. If the error still exists, replace the fieldbus coupler.
10	Internal execution error in the memory management	1. Inquire about a firmware update for the fieldbus controller. 2. If the error still exists, replace the fieldbus coupler.
11	Error when accessing the memory of the fieldbus connection	1. Replace the fieldbus coupler.
12	Maximal process image size exceeded	1. Turn off the power supply to the node. 2. Reduce number of I/O modules. 3. Turn the power supply on.

Errors reported with blink code 6 cause the fieldbus connection to shut down. Communication with the EtherCAT® master is not possible.

9.2 I/O Module Diagnostics

The fieldbus coupler provides the EtherCAT® master with information about the diagnostic status of the I/O modules. The information is transmitted to the EtherCAT® master by means of a 16 bit word (diagnostic status word) in the input process image. The EtherCAT® master can control the diagnostic information display by means of a 16 bit word (diagnostic control word) in the output process image.

The fieldbus coupler stores upstream diagnostic events, like, e.g. a line break that occurred. Downstream diagnostic events are not recorded.

The fieldbus coupler has a buffer for recording diagnostic events. A maximum of 64 events can be stored in this buffer.

9.2.1 Diagnostic Control Word

The diagnostic control word is used to control I/O module diagnostics. The diagnostic control word is located in the output process image and is transferred from the EtherCAT® master to the fieldbus coupler. The fieldbus coupler processes the diagnostic control word in every process data cycle.

Note

Processing is delayed if the EtherCAT® cycle time is short!
If the EtherCAT® cycle time is shorter than the fieldbus coupler's internal cycle time, processing is delayed until the fieldbus coupler has completed its internal processing and can once again process the process data.

Table 106: Diagnostic Control Word

Diagnostic Control Word												Reset value:0x0000			
High-byte								Low-byte							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
DIAGACK	CLEAR	SBZ													

Bit	Function
SBZ.	This bit field must be set by the EtherCAT® master to "0".
CLEAR	This bit provides the EtherCAT® master with the possibility of purging the diagnostic buffer. If the EtherCAT® master has switched this bit to "1", then it has to wait for the acknowledgement of the bits by the fieldbus coupler. 0: The content of the diagnostic buffer is not purged. Entries are reported to the EtherCAT® master in the diagnostic control word. 1: If the acknowledgement bit “RESACK” in the diagnostic status word is reported with status “1”, the fieldbus coupler is requested to delete the content of the diagnostics buffer.
DIAGACK	This bit allows the acknowledgement of the event that is reported in the diagnostic status word. The fieldbus coupler sets the content of the diagnostic status word to "0" if the status "1" of the bit was recognized and processed. The fieldbus coupler reports further diagnostic events only in the case where the EtherCAT® master has reset this bit to a value of "0". 0: The content of the diagnostic status word is transmitted to the EtherCAT® master unchanged as long as a valid diagnostic event is reported. The content of the diagnostic status word is automatically changed if a valid event is not indicated and a new event occurs. 1: The EtherCAT® master requests that the indicated diagnostic event be deleted.

9.2.2 Diagnostic Status Word

The diagnostic status word serves to indicate the I/O module diagnosis in the input process image and contains information about a diagnostic event and also the acknowledgement status of diagnostic events.

The following table shows the meaning of the bits of the diagnostic status word.

Table 107: Diagnostic Status Word

Diagnostic Status Word												Reset value:0x0000			
High byte								Low byte							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
EVENT	CLRACK	EVTCODE			CHNUM			TRMNLNUM							

Bit	Function
TRMNLNUM	This bit field contains the position of the I/O module that reported the diagnostic event. '0x00' ... '0x3F': I/O module position 0 ... 63.
CHNUM	This bit field contains the number of the signal channel by means of which the diagnostic event was determined. '000' ... '111': number of the signal channels 0 ... 7.
EVTCODE	This bit field contains the coding of the diagnostic event. '111': The diagnostic event was reported by an analog I/O module. All further bit combinations are assigned to digital I/O modules. The diagnostic coding is entered in the bit field as it is provided by the I/O module. The meaning of the coding is available in the documentation associated with the I/O module.
CLRACK	This bit acknowledges the receipt of the CLEAR command from the EtherCAT® master. The EtherCAT® master can keep the CLEAR command active for as long as it takes until the fieldbus coupler acknowledges the receipt of the command via this bit. Then the EtherCAT® master can purge the CLEAR command in the diagnostic control word. 0: The fieldbus coupler has not received a CLEAR command or has not yet processed a CLEAR command. 1: The fieldbus coupler has received and processed a CLEAR command. All entries in the diagnostic buffer have been purged.
EVENT	This bit indicates that at least one event is stored in the diagnostic buffer. 0: No event is present in the diagnostic buffer. The status of bits D13:D0 of the diagnostic status word may not be observed by the EtherCAT® master. 1: At least one event is stored in the diagnostic buffer. The bits D13:D0 of the diagnostic status word contain the description of the diagnostic event. Additional events, which are stored in the diagnostic buffer as necessary, are only displayed after this event has been acknowledged by the EtherCAT® master.

9.3 Behavior of the Fieldbus Coupler during Interruption of Operations

An interruption of operation occurs when the fieldbus coupler can no longer exchange process data with the master and/or the I/O modules.

9.3.1 Loss of Power

In the case loss of power outage or falling below the minimum level of the power supply to the fieldbus coupler, the communication with the master and the I/O modules will be interrupted. The I/O modules connected to the fieldbus coupler will switch their output data to a value of “0”.

9.3.2 Loss of Fieldbus

The fieldbus coupler determines that a loss of the fieldbus has occurred when the communication to the master is interrupted. A loss of fieldbus can be caused by losing the master itself or by an interruption in the communication connection.

A loss of fieldbus additionally means that the fieldbus coupler cannot receive any output **process** data from the master nor can it send any input **process** data to the master.

During a loss of fieldbus, the fieldbus coupler switches the output signal of the I/O modules to a value of “0”.

9.3.3 Local Bus Error

The fieldbus coupler determines that a local bus error has occurred when the communication with the I/O modules is disrupted or interrupted. A local bus error can occur due to the removal e.g. of an I/O module from the fieldbus node.

In addition, a local bus error means that the fieldbus coupler cannot exchange any more process data with the I/O modules.

The I/O modules switch their output signals to a value of “0” in the case of an error.

The fieldbus coupler reports a local bus error by sending a blink code. To send the blink code, the fieldbus coupler uses the I/O LED.

Information

More information about the LED Signaling
Read the detailed description for the evaluation of the displayed LED-Signals in the chapter "Diagnostics", "Evaluating Node Status - I/O LED (Blink Code Table)".

The fieldbus coupler reports local bus errors to the EtherCAT® master when the status of the EtherCAT® communication has at least reached PREOP status. The fieldbus coupler reports local bus errors to the EtherCAT® master by means of EMCY messages.

Information

Additional information about the EMCY code!
A list of the EMCY codes and their meanings is available in the appendix, “EMCY Codes” chapter.

If a local bus error occurs in OP status, the EtherCAT® fieldbus coupler switches into SAFEOP+ERR status.
The outputs of the I/O modules are switched to “0”.

In SAFEOP or OP status, a local bus error leads to a change to SAFEOP+ERR status with the corresponding AL status code.

Information

Additional information about AL status codes!
A list of the AL status codes and their meanings can be found in the appendix, “AL Status Codes” chapter.

The master can acknowledge the error and bring the slave into SAFEOP status. A change in OP status is, however, only possible if the local bus has been restarted. The Sync Managers 2 and 3 are deactivated until the local bus error has been remedied.

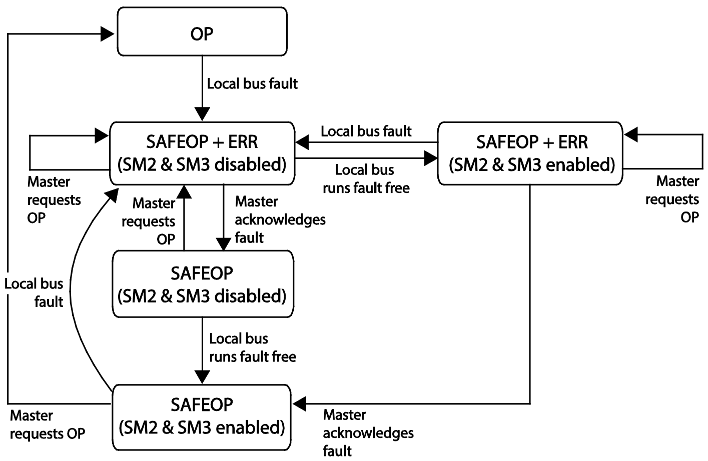


Figure 46: Behavior During a Local Bus Error

No status change occurs in INIT or PREOP states because the device is not participating in the cyclical exchange of process data and the local bus error therefore does not influence the EtherCAT® communication.

If the local bus is not running, a status change from PREOP to SAFEOP is not possible.

If the fieldbus coupler is in PREOP, SAFEOP, or OP states, an EMCY message is sent using the LED blink code assigned to local bus error.

Information

Additional information about EMCY messages
You can read about the EMCY messages in more detail in the appendix, “EMCY Codes” chapter.

Information

Additional Information about the LED blink codes
You can read about the LED blink codes in more detail in the “Diagnostics” chapter, → “Fieldbus Coupler Diagnostics” → “Node Status Evaluation – I/O LED (Blink Code Table)”.

9.4 Behavior during Other Operating Errors

During regular operation, the following errors can occur during certain circumstances:

- Incorrect configuration of a Sync Manager by the EtherCAT® master
- Interruption of the cyclical process data exchange
- Error in CoE access

9.4.1 Configuration Error in a Sync Manager

Configuration errors in one of the Sync Managers can occur during status change from INIT to PREOP or from PREOP to SAFEOP. These lead to a denial of the requested status change by the corresponding AL status code.

Information

Additional information about AL status codes!
A list of the AL status codes and their meanings can be found in the appendix, "AL Status Codes" chapter.

The settings of Sync Managers 0 and 1 (mailbox) are checked during status change from INIT to PREOP.
The master extracts the settings either from the SII (slave's EEPROM) or from the optional XML file with the device description of the EtherCAT® fieldbus

coupler. The EtherCAT® fieldbus coupler requests the settings specified in the following lists for the Mailbox Sync Managers 0 and 1.

Table 108: Settings for the Sync Managers SM0 and SM1

Sync Master	**Minimum size**	**Maximum size**	**Start address**	**Control register**	**Activate register**
0	32	1024	0x1000	$001X0110_2$	$XX0000X1_2$
1	32	1024	0x1400	$001X0010_2$	$XX0000X1_2$

The settings for Sync Managers 2 and 3 (process data) is checked during status change from PREOP to SAFEOP. The master calculates the size of the cyclically transmitted input and output data using the Sync Manager Assignment Lists and the Rx/TxPDO's.

Information

Additional information about mapping process data!
More information about mapping process data is located in the “Function Description” chapter, → “Process Data Architecture” → “General Mechanisms used by the EtherCAT® to Map Process Data”.

The master extracts the usual parameters either from the SII (slave's EEPROM) or from the optional XML file with the device description of the EtherCAT® fieldbus coupler. The EtherCAT® fieldbus coupler requests the settings specified in the following list for the Mailbox Sync Managers 2 and 3.

Table 109: Settings for the Sync Managers SM2 and SM3

Sync Master	Size	Start address	Control register	Activate register
2	According to mapping	0x1800	$011X0100_2$	$XX0000X1_2$
3	According to mapping	0x2400	$000X0000_2$	$XX0000X1_2$

If the status change from PREOP to SAFEOP is denied due to an incorrectly set Sync Manager, then either one or more additional EMCY messages are sent to the master with the expected settings for the Sync Manager:

Table 110: Sync Manager Length Error EMCY Data

<table>
<tr><th colspan="7">Sync Manager Length Error EMCY data</th></tr>
<tr><th>Error Code</th><th>Error Register</th><th>Byte 0</th><th>Byte 1</th><th>Byte 2</th><th>Byte 3</th><th>Byte 4</th></tr>
<tr><td rowspan="2">0xA000</td><td rowspan="2">0x11</td><td rowspan="2">0x00
+
SM
channel*4</td><td>low byte</td><td>high byte</td><td>low byte</td><td>high byte</td></tr>
<tr><td colspan="2">Minimum length (WORD)</td><td colspan="2">Maximum length (WORD)</td></tr>
</table>

Table 111: Sync Manager Address Error EMCY Data

<table>
<tr><th colspan="7">Sync Manager Address Error EMCY data</th></tr>
<tr><th>Error Code</th><th>Error Register</th><th>Byte 0</th><th>Byte 1</th><th>Byte 2</th><th>Byte 3</th><th>Byte 4</th></tr>
<tr><td rowspan="2">0xA000</td><td rowspan="2">0x11</td><td rowspan="2">0x01
+
SM
channel*4</td><td>low byte</td><td>high byte</td><td>low byte</td><td>high byte</td></tr>
<tr><td colspan="2">Minimum length (WORD)</td><td colspan="2">Maximum length (WORD)</td></tr>
</table>

Table 112: Sync Manager Settings Error EMCY Data

<table>
<tr><th colspan="35">Sync Manager Settings Error EMCY data</th></tr>
<tr><th>Error Code</th><th>Error Register</th><th>Byte 0</th><th colspan="8">Byte 1</th><th colspan="8">Byte 2</th><th colspan="8">Byte 3</th><th colspan="8">Byte 4</th></tr>
<tr><td rowspan="3">0xA000</td><td rowspan="3">0x11</td><td rowspan="3">0x02
+
SM
channel*
4</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">2</td><td>1</td><td>1</td><td colspan="2">2</td><td colspan="2">2</td><td colspan="8">8</td><td colspan="7">7</td><td>1</td><td colspan="8">8</td></tr>
<tr><td colspan="2">x1</td><td>x2</td><td>x3</td><td colspan="2">x4</td><td colspan="2">x5</td><td colspan="8">x6</td><td colspan="7">x7</td><td>x8</td><td colspan="8">x9</td></tr>
</table>

x1	Byte 1, Bit 7-6	reserved
x2	Byte 1, Bit 5	Expected AL Event Enable
x3	Byte 1, Bit 4	reserved
x4	Byte 1, Bit 3-2	Expected Direction
x5	Byte 1, Bit 1-0	Expected Buffer Type
x6	Byte 2, Bit 7-0	reserved
x7	Byte 3, Bit 7-1	reserved
x8	Byte 3, Bit 0	Expected Channel Enable
x9	Byte 4, Bit 7-0	reserved

9.4.2 Interruption of the Cyclical Process Data Exchange

Interruptions in the cyclical process data exchange occur as a result of a severed or disrupted network connection. These interruptions are recognized by the slave with the assistance of the process data watchdog.

The watchdog's interval is predefined by the master and is usually 100 ms.

If an interruption of the process data exchange is detected while the EtherCAT® fieldbus coupler is in OP status, the fieldbus coupler changes to SAFEOP+ERR status and the Sync Manager 2 is deactivated.

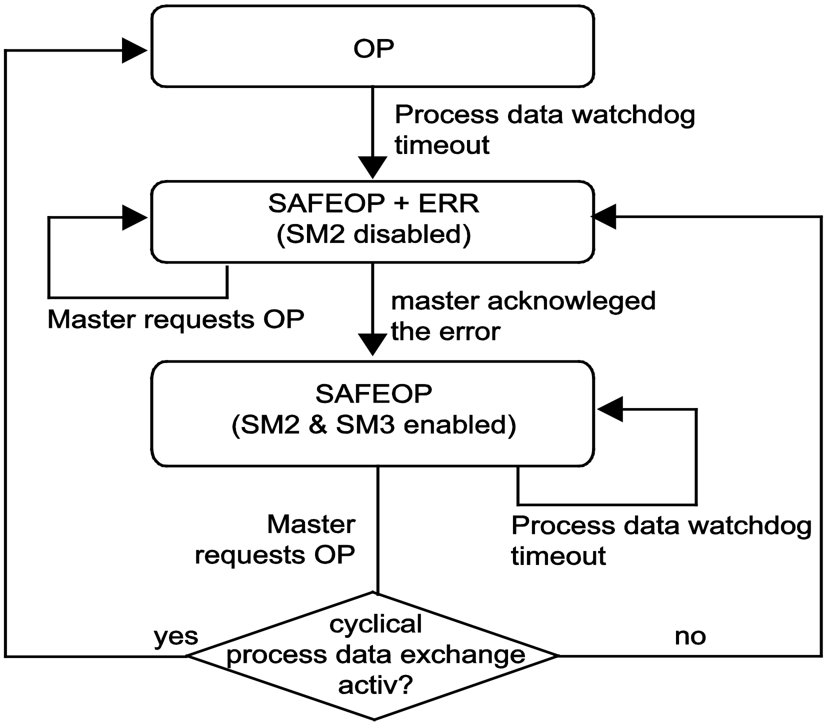


Figure 47: Error Behavior of the Process Data Watchdog

If the master acknowledges the error, then the slave switches to SAFEOP status and the Sync Manager 2 is reactivated. The fieldbus coupler can be switched to OP status as soon as the cyclical data exchange is running again.

If the fieldbus coupler is in INIT or PREOP status, then an interruption of the process data exchange does not lead to an error because the device is not participating in the cyclical process data exchange in these states.

If the interruption of the process data exchange is recognized in SAFEOP status, then it does not lead to an error. However, the current output data have to be transmitted prior to the switch to OP status. Otherwise, the change of status will be denied.

9.4.3 Errors During CoE Accesses

If an error occurs during an SDO access of the Object Dictionary, then the access is denied with an SDO Abort Request and an SDO Abort Code.

Information

Additional information about SDO Abort Codes!
You can find more information in the list of SDO Abort Codes in the appendix, “SDO Abort Codes” chapter.

If a write access is denied, then the data in the object, which was to be written, remain unchanged.

10 Fieldbus Communication

10.1 EtherCAT®

10.1.1 General

The real time, Ethernet solution, EtherCAT® (Ethernet for Control Automation Technology), is a 2005 IEC standardized, open protocol, which distinguishes itself through extremely short cycle times ≤ 100 µs and achieves an exact synchronization using distributed clocks and a jitter of ≤ 1 µs.
Thus, the EtherCAT® protocol is equally suitable for very high and for very low real time demands. For example, EtherCAT® can be ideally used in situations in which spatially distributed processes should run simultaneously, e.g. when several servo axles are supposed to execute simultaneously coordinated movements.

During the structuring of your network, EtherCAT® additionally offers the possibility for optionally selecting and combining topologies, such as linear, star, and tree, for a total of up to 65535 participants. In addition, by using switches and media converters, different transmission media can be used for each individual section, such as standard Ethernet patch cable (100 base TX), fiber optics, and copper wire. Thus, the flexibility of the network is practically endless and hardware costs can be kept relatively low.

EtherCAT® uses Timestamp for a swift, precise and problem-free diagnosis. In addition, EtherCAT® communications can be exactly monitored using any standard Ethernet monitoring tool due to the fact that EtherCAT®, as an Ethernet based fieldbus, uses standard Ethernet Frames according to IEEE 802.3. This also enables a very easy integration of EtherCAT® into an existing Ethernet network or, via gateways, into other fieldbus systems, like CANopen®, DeviceNet, or PROFIBUS.

In order to further develop the technology, numerous interested parties, manufacturers, and users of EtherCAT® have united in a large Industrial Ethernet EtherCAT® Technology Group, which provides additional information via the Internet.

Information

Additional information about EtherCAT®
More information about the EtherCAT® technologies can by found at the EtherCAT® association's Internet site at: www.ethercat.org.

10.1.2 Network Structure

10.1.2.1 Transmission Medium

EtherCAT® relies on standard Ethernet hardware according to IEC 8802-3. Standard Ethernet cables (twisted pair cables), as well as transmission media such as fiber optics and copper wire using switches or media converters, are used. A combination of different media can also be used for individual sections. The network flexibility is thus practically limitless.

The use of shielded cables (with S/UTP or S/STP shielding) is recommended when used in an industrial environment (environments with interference). These cables have a high immunity to interference due to double shielding consisting of a copper wire mesh and an aluminum foil. The usual name for this type of cable is Cat5e.
All EtherCAT® devices are full-duplex-capable and use the Ethernet transmission standard 100Base-TX (copper) to transmit data. The maximum cable length of 100 m specified for Ethernet 100Base-TX between two EtherCAT® devices is thus supported. When using fiber optic cable, a distance of up to 2 km between two fieldbus nodes can be achieved.
Thanks to the autocrossover function (Auto-MDI/MDI-X), which automatically detects the send and receive data direction, crossed and/or non-crossed patch cables can also be used.

Information

Additional information about Ethernet
For more documents and information about Ethernet, please refer to the IEEE Homepage: www.ieee.org

10.1.2.2 Network Topology

The two RJ-45 interfaces in the EtherCAT® fieldbus coupler enable the generation of diverse typologies.

In addition to the standard Ethernet star topology with switches, line, star, and tree topologies can be constructed using EtherCAT®, and the various topologies can also be combined without requiring additional switches.

The construction of a ring topology additionally offers the advantage of cable redundancy as well as the ability to connect, disconnect, or exchange a fieldbus node during operation.

From the point of view of EtherCAT®, it does not matter in which position within the selected topology the EtherCAT® fieldbus coupler is used. The user can decide this at will.

10.1.2.3 Couplers

For the star topology, the classic Ethernet structure, switches can be optionally used as couplers.

The use of routers enables communication with an additional sub-network.

10.1.3 Network Communication

10.1.3.1 Communication Principle

The EtherCAT® fieldbus uses data transmission via standard Ethernet Frames through a special Ethertype (0x88A4).

The communication principle can therefore be clearly compared using a telegram train of 100 Mbit/s that drives through the transmission lines and never stops. The EtherCAT® data are conveyed on this train (that is, in the Ethernet Frame) like passengers. The passengers can thus be individual bits or also several bytes. The sub-telegrams embody the individual train cars and have varying lengths. The passengers (data) jump onto or off of the moving train at specific stations, that is, fieldbus nodes. Entry and exit of the train occurs “on the fly”. The telegram train is only delayed by a few nanoseconds in each case. In addition, counters in the train are updated based on the exchange.
An exchange within a fieldbus node only occurs, however, if this node is addressed.
From the point of view of each fieldbus node, the telegram as a whole is not seen. Instead, the entire telegram runs by a type of viewing window in the fieldbus node, which continually observes the telegram. The data exchange takes place immediately, as soon as the station address is recognized.

10.1.3.2 Addressing

The physical order of the Ethernet I/O modules in the network is not relevant to the data link order or to the addressing. Addresses can be freely selected. EtherCAT® uses an implicit addressing. Slaves are automatically assigned addresses. In a pure EtherCAT® network, there is therefore no need for manual participant addressing or external switches.

The addressing remains constant, even during subsequent changes, and therefore does not require manual setting. An EtherCAT® network can have a maximum total of 65535 participants.

The slaves can communicate with each other using broadband, multicast, and cross communication.

If communication occurs between control computers and EtherCAT® devices in one and the same subnet, the transmission is set directly to Ethernet Frames.

If, on the other hand, EtherCAT® communication is expanded to other subnets, routers and also EtherCAT® UDP can be used. In this process, the EtherCAT® protocol is packed in a UDP/IP datagram. Thus, every controller that has an Ethernet Protocol stack can address EtherCAT® systems. The EtherCAT® network response times are thus minimally limited because the UDP datagram is always only unpacked in the first station.

The allocation of data in the process image can be freely configured. Depending on the configuration, the data are copied directly to the desired location in the process image which has a logical address space of 4 gigabytes.

10.1.3.3 Configuration

The configuration of master boards with individual station data is not necessary in order to construct a connection between SPS and the fieldbus devices. The real layout is quasi recreated in the configuration tool for this. The necessary software for configuration, start-up and diagnosis of the EtherCAT® network is part of the delivery of the master cards, PC cards or is also in the master software used in EtherCAT®.

The data required by WAGO EtherCAT® fieldbus couplers are made available through the integration of ESI files (EtherCAT® slave information) as EtherCAT® device profiles in the configuration software.
The device profiles describe the application parameters and the functional behavior of the devices including the device-class specific finite status machines. The device manufacturer supplies an individual ESI file for each EtherCAT® device. During configuration, the data, objects, and parameters defined in the ESI file are selected in order to assign them to the actual layout and the EtherCAT® devices.

The device classes for EtherCAT® correspond to the many device classes that already exist, e.g. for I/O devices, drives, or valves. Likewise, the parameters and tools for them are already known from other device profiles.
Therefore, instead of separate device profiles, EtherCAT® offers simple interfaces for existing device profiles. By adapting the previous fieldbus to EtherCAT®, the conversion is facilitated for the device manufacturer and above all for the customer.

Information

Additional information about the EtherCAT® XML file.
The Device Description File "ESI-Dateien für EtherCAT Koppler / Serie 750, 753” is available for download from the WAGO homepage **www.wago.com** by Search Term: “**ESI**”

10.1.3.4 EtherCAT® State Machine (ESM)

Each slave contains a so-called EtherCAT® State Machine (ESM) that controls the interplay between maser and slave during start-up and during operation.

For simple slaves, the EtherCAT® State Machine always follows the status requested by the master, whereas complex slaves can autonomously implement status changes and can also deny the status change requested by the master, e.g. in the case of a configuration error.

The status of EtherCAT® slaves is controlled using the EtherCAT® State Machine. The EtherCAT® slave runs through the following states:

- INIT
- PREOP (Pre-Operational)
- SAFEOP (Safe-Operational)
- OP (Operational)
- BOOT (Bootstrap), only optionally implemented

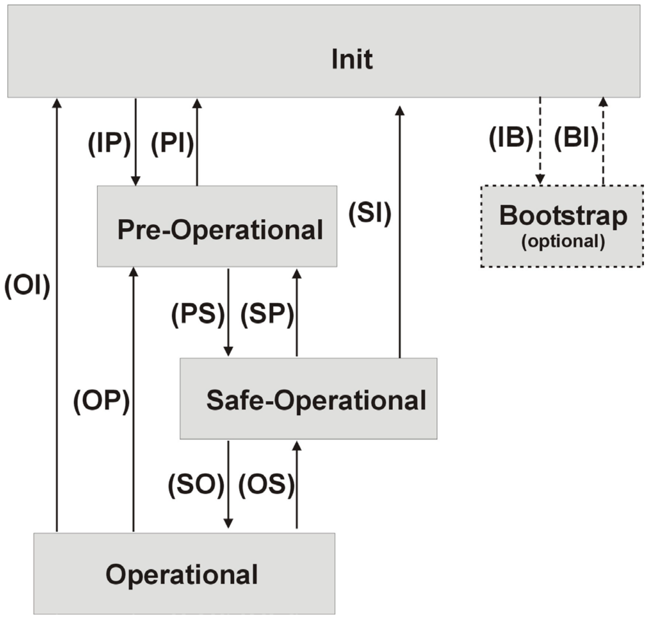


Figure 48: Status Diagram of an EtherCAT® Slave

The master always specifies a set-status (AL control) which the slave attempts to engage. The status transitions in a complex slave are partially related to conditions and can trigger different actions in the slave. The current status of a slave (AL status) can be read by the master at any time. In addition to status, the AL status also contains an error bit (ERR) that can be read together with the status. When necessary, the master can read an error code (AL status code) from the slave.

The error bit in the AL status can be set in INIT, PREOP, and SAFEOP states. This is shown in an appended +ERR in the name of the status: INIT+ERR, PREOP+ERR, SAFEOP+ERR.

If the error bit of the slave is set, then the AL status code contains a word different from null.

Information

Additional information about AL status codes!
A list of the AL status codes and their meanings can be found in the appendix, "AL Status Codes" section.

Before another status can be engaged from an error status, the error has to be acknowledged by the master. This takes place via an additional Acknowledge bit in the AL control, which is written in the slave together with the desired status.

The states have the following meaning in connection with a complex slave:

Table 113: EtherCAT® State Machine (ESM)

Status	Transition	Communication	Procedure
Init	As soon as the EtherCAT® slave is switched on, it is in INIT status.	Neither mailbox nor process data communication is possible in this status.	The EtherCAT® master reads the relevant register of the EtherCAT® slave ASIC and the content of the EEPROM (SII -- slave information interface) connected to it and initializes the Sync Manager channels 0 and 1 for mailbox communication.
Pre-Operational	During the transition from INIT to PREOP status, the EtherCAT® slave checks whether the mailbox has been correctly initialized. Conditions for the transition are that the EtherCAT® slave has initiated at least once successfully and has constructed the process image.	Mailbox communication is now possible; however, process data communication is not yet available.	The EtherCAT® master now initializes the Sync Manager channels (beginning with Sync Manager channel 2) for process data, as well as the FMMU channels (fieldbus memory management unit) and, in case the slave supports configurable mapping, the PDO mapping or the Sync Manager PDO assignment. In addition, the settings for the process data transmission are transmitted in this state, as well as, when necessary, the settings for the module-specific parameters that deviate from the default settings.
Safe-Operational	During the transition from PREOP to SAFEOP status, the EtherCAT® slave checks whether the Sync Manager channels for process data communication are correct, as well as, if necessary, whether the settings for the distributed clocks are correct.	Mailbox and process data communication are both possible.	During communication, the slave keeps its outputs in a secure state. The input data are, however, cyclically updated.
Operational	Before the EtherCAT® master switches the EtherCAT® slave from SAFEOP to OP status, it must already transmit valid output data.	Mailbox and process data communication are both possible	In this status, the slave copies the output data of the master in its outputs.
Boot (optional)*)	This status can only be reached via INIT status	Neither process data nor mailbox communication is possible, with the exception of mailbox communication via the File-Access protocol over EtherCAT® (FoE).	In this status, an update of the slave firmware can be carried out.

*) The BOOT status is not implemented in the EtherCAT® fieldbus coupler.

10.1.3.5 Synchronization Using Distributed Clocks

For spatially distributed processes, in which actions are temporally calibrated with each other or have to be simultaneously executed, such as for the coordinated movements of servo axles, it is extremely important that the synchronization works. However, a fully synchronous communication in a network always has the disadvantage that the quality of the synchronization is immediately degraded as soon as disruptions occur in the communication.
In contrast, EtherCAT® uses distributed clocks, which can be exactly compared to a main clock for synchronization.

The time on the main clock is transmitted to the distributed clocks via EtherCAT® for this purpose. This main clock is located in an EtherCAT® slave. Therefore, no special hardware is required in the master.
Based on the logical ring structure for EtherCAT® communication, the main clock can detect the run time offset in the slave clocks easily and exactly, and readjust the resulting correction values to a run time compensation that corresponds to this.
By this means, a highly exact network-wide time basis is available with a jitter that is significantly less than a microsecond. Thus, for example, a deviation of ± 20 ns can be achieved in the case of 300 participants and a cable length of 120 m.

By using distributed clocks, the communication system is not sensitive to possible delays caused by disruptions.
This type of synchronization is described in the new IEEE Standard 1588.

There are, in addition, more advantages for high-resolution clocks, namely, the supply of exact information for the local time of data acquisition. Thus, controls from consecutively measured positions can, for example, calculate speeds. EtherCAT® has an expanded Timestamp data type for these timestamps in addition to the timestamps transmitted in addition to the reference data. Using the measured value, the local time is connected to a resolution of up to 10 ns. Thus, there is no longer a dependence on the communication system's jitter.

Another expanded data type is the Oversampling Data Type.
The Oversampling Data Type allows multiple sampling of a process datum within a communication cycle. The oversampling factor indicates the number of samplings within a communication cycle. Therefore, sampling rates of 200 kHz are easily possible.
The data are transmitted in an array (outputs from the previous transmission, inputs from the subsequent transmission).
The sampling is triggered by the local clock and/or the system time.

10.1.3.6 Performance

EtherCAT® network performance is enormous and enables control and regulation concepts that could not be realized using classical fieldbus systems.
The high performance of EtherCAT® technologies can be achieved because the entire protocol processing takes place in the hardware, and is thus independent of protocol stack runtime, CPU performance, or software implementation.

In addition, the reference data rate is extremely high because data can be exchanged in the sending as well as in the receiving direction. Actual data rates of more than 100 Mbit/s (> 90% reference data rate of 2 x 100 Mbit/s) can be achieved by the full duplex characteristic of 100BASE-TX.
The principle of EtherCAT® technology is not linked to 100 Mbit/s, it is scalable and even allows for an expansion to Gigabit Ethernet.

Furthermore, bandwidth utilization is maximized because each individual user and each individual datum does not require its own frame. This results in extremely short cycle times ≤ 100 µs. Up to 1486 bytes of process data can be exchanged in a single Ethernet frame, which corresponds to almost 12,000 digital inputs and outputs. Only 300 µs are required to transmit such a large quantity of data.
A power current control of distributed drives can take place using the Ethernet system in addition to the speed control, among others.
The large bandwidth enables the system to transmit more status information per datum.

As an example: The update time for 100 servo axles having 8 bytes of input and output data respectively is only 100 µs. In this time period, all axles are provided with nominal values and control data and report their actual positions and status. Using distributed clocks, the axles can be synchronized with a deviation of significantly less than a microsecond.
For 200 analog input and output data, for example, only 50 µs (at 20 kHz) are required, and only 30 µs for 1000 distributed digital input and output data.

10.1.3.7 Diagnostics

Diagnostics demonstrates another outstanding strength of using EtherCAT®: Disruptions are recognized quickly and precisely, and are clearly localized so that they can be remedied in an extremely short period of time.

In the start-up procedure, a check is made at the system start as to whether the actual input module and output module configurations are identical with the stored set configurations. The same applies for the typology. In addition, the network can be automatically read by the configuration upload.

If bit errors occur during the transmission, they are reliably recognized by the evaluation of the CRC checksum.
The quality of each individual transmission section can be individually monitored in addition to the recognition and localization of break points. Critical network segments are exactly localized by the automatic evaluation of the corresponding

error counter, even if the error sources only exist temporarily, such as damage to cables, defective plug connections, or EMI influences.

10.1.4 EtherCAT® Interfaces

EtherCAT® technology is not only completely Ethernet compatible, it also harmonizes well with other services and protocols based on Ethernet TCP/IP that share the same physical network. Thus, all Internet technologies can also be used in the EtherCAT® environment, such as integrated webservers, E-mail, FTP transfers, etc.
For industrial applications, other fieldbus systems, like CANopen, DeviceNet, and PROFIBUS can also be integrated into an EtherCAT® network via gateways.
All of this causes only minimal network impact.

The interfaces and the integration of the WAGO EtherCAT® fieldbus coupler into other fieldbus systems will be explained in more detail in the following chapters.

10.1.4.1 Slave Information Interface (SII)

An EEPROM in the EtherCAT® slave is designated as an SII, which can be read directly by the master via the EtherCAT® fieldbus chip.
In addition to information about the configuration of the EtherCAT® chip, it also contains the following information for the master:

- Information about the identification of the slave (manufacturer ID, product ID, version, series number)
- Parameters for setting the Sync Managers 0 and 1 (mailbox)
- Information about supported mailbox protocols (e.g. CoE)
- Process data mapping and parameters for setting the Sync Managers 2 and 3 (for simple slaves with fixed process data architecture)

10.1.4.2 CoE Interface (CAN Application Layer Over EtherCAT®)

Intelligent EtherCAT® slaves have a CoE interface and an Object Dictionary (OD). The Object Dictionary contains:

- Information about the device
- Process data
- Mapping of the process data
- Error and status information
- Objects for setting the device parameters
- Additional manufacturer-specific objects

By using a mailbox with the assistance of CANopen®, known SDOs can be accessed from the entries in the Object Dictionary.

Likewise, EtherCAT® can be implemented on CANopen® devices with very little effort.

Communication mechanisms known from CANopen® – object dictionary, PDO (process data objects) and SDO (service data objects) – can be used with EtherCAT®. This allows you to use a major portion of the CANopen® firmware; the network management is also comparable.
When you include the greater bandwidth of EtherCAT®, it is possible to expand the objects even further.

The CoE comprises the components described in the following table:

Table 114: CoE Components

CoE Components	Description
EtherCAT® State Machine	The status of the EtherCAT® slaves is controlled using the EtherCAT® State Machine.
Object Dictionary	All EtherCAT® slaves that support the CoE interface have an Object Dictionary. This contains all parameters, diagnostics, process and other data that can be read or written using EtherCAT®. The ESI file supplies the Object Dictionary that is selected by the SDO information service. For this reason, the information service has to be laid out such that the object description part, beginning with Index 0x1000 of each object respectively, can be selected, as this contains the data type, length, access rights, and information about whether the object can be used as a process datum and imaged in a PDO.
Process Data	The mapping, i.e. the arrangement of a devices' process data with the EtherCAT® process data, takes place using the PDO mapping objects. The Sync Manager PDO Assign Objects also writes which objects are to be transmitted from the Object Dictionary as process data using EtherCAT®. The minimum requirement for an EtherCAT® slave is that this PDO mapping and Sync Manager PDO Assign Objects are readable. If the EtherCAT® slave also has process data mapping that can be configured by the EtherCAT® master, then the PDO mapping and the Sync Manager PDO Assign Object are also writable. The Sync Manager Communication Objects serve to determine the cycle time with which the related process data will be transmitted via EtherCAT® as well as the form of the synchronization for this transmission.
EtherCAT® Start-Up	The link between the EtherCAT® Start Machine, the process data mapping, and setting the device parameters in the run-up of the EtherCAT® network is written by the EtherCAT® start-up.
Command Objects	Command objects allow actions to start in a device that require a certain time until the results are available.
Emergencies	Emergencies are status messages that transmit diagnostics or process results with timestamps, e.g. for the record function of an event logger. The status messages about the current status of the device, which must be synchronous with the actual process data in the control application, should, in contrast, be transmitted directly with the process data.

PDO (Process Data Objects)

Process Data Objects (PDO) are used to quickly and efficiently exchange real-time data, such as input and output data, set and actual values.
No objects are addressed in the EtherCAT® telegram. The contents of the process data are sent directly from the previously mapped parameters.

SDO (Service Data Objects)

The Service Data Objects (SDO) are used to access reading and writing to the Object Dictionary and to transmit device parameters. Based on the non-cyclical transmission of parameters, such as only once during run-up, the SDO's are low priority.

Command Objects

Command objects can be used when actions or commands must be implemented that cannot, however, be realized via a single SDO upload or SDO download service, e.g. because request and response data are necessary, or because the execution takes too long temporally, such that an SDO timeout would occur.

Using a command object divides the command into at least two SDO services.
The command is started by writing the command object on subindex 1 using the SDO download.
The answer is retrieved by reading subindex 3 using the SDO upload.

If the answer is not available during the reading of subindex 3, the first byte of the answer data can provide information about the progress.
Subindex 1 is not writable until the command has been ended and the answer has been read, so that the accesses can no longer be mutually overwritten by different applications.

Subindex 2 is only defined for compatibility reasons using the appropriate CANopen® DS301 definition.

The data type of the object description for command objects is 0x25.

Table 115: Command Object Structure

Sub-Index	Description	Data Type	Value
1	Command	OCTET_STRING	Byte 0-n: Service Request Data The command is implemented by a write access to the command data.
2	Status	UNDESIGNED8	0: last command implemented, no error, no answer 1: last command implemented, no error, answer available 2: last command implemented, error, no answer 3: last command implemented, error, answer available 4-99: reserved for later use 100-200: shows in % how far the command has been implemented (100=0%, 200=100%) 201-254: reserved for later application 255: command is not immediately implemented (if the percent display is not supported)
3	Reply	OCTET_STRING	Byte 0: like subindex 2 Byte 1: not used Byte 2-n: Service Response Data

11 I/O Modules

11.1 Overview

For modular applications with the WAGO-I/O-SYSTEM 750, different types of I/O modules are available

- Digital Input Modules
- Digital Output Modules
- Analog Input Modules
- Analog Output Modules
- Communication Modules, Supply and Segment Modules
- Function and Technology Modules

For detailed information on the I/O modules and the module variations, refer to the manuals for the I/O modules.

You will find these manuals on the WAGO web pages under www.wago.com.

Information

More Information about the WAGO-I/O-SYSTEM
Current information on the modular WAGO-I/O-SYSTEM is available in the Internet under: www.wago.com.

11.2 Process Data Architecture for EtherCAT®

In the case of an EtherCAT® fieldbus coupler, the process image uses a word structure (with word alignment). The internal mapping method for data greater than one byte conforms to the Intel format.

The following section describes the process image for various WAGO-I/O-SYSTEM 750 and 753 I/O modules when using an EtherCAT® fieldbus coupler.

The fieldbus coupler processes diagnostic information provided by the I/O modules in addition to the process images. Diagnostic information is reported by the fieldbus coupler in the diagnostic status word.

When using digital I/O modules with diagnostics, the status of the diagnostic information is entered, like that provided by the I/O module, in the EVTCODE bit field of the diagnostic status word.

Information

Additional information about the diagnostic status word
The structure of the diagnostic status word is available in the "Diagnostics" chapter, → "I/O Module Diagnostics" → "Diagnostic Status Word"

When using analog I/O modules, the EVTCODE bit field is set to the value 7' during the occurrence of a diagnostic event.

Note

Equipment damage due to incorrect address!
Depending on the specific position of an I/O module in the fieldbus node, the process data of all previous byte or bit-oriented modules must be taken into account to determine its location in the process data map.

11.2.1 Digital Input Modules

Digital input modules supply one bit of data per channel to specify the signal state for the corresponding channel. These bits are mapped into the Input Process Image.

Digital input modules with diagnostics have one or more diagnostic bits available in addition to the process data. The diagnostic bits are evaluated by the fieldbus coupler. In the event of a diagnostic message, the fieldbus coupler enters the state of the diagnostic bit in the diagnostic status word. The entries in the diagnostic status word are made channel-specific.

When analog input modules are also present in the node, the digital data is always appended after the analog data in the Input Process Image, grouped into bytes.

11.2.1.1 1 Channel Digital Input Module with Diagnostics

750-435

Table 116: 1 Channel Digital Input Module with Diagnostics

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Status bit S 1	Data bit DI 1

11.2.1.2 2 Channel Digital Input Modules

750-400, -401, -405, -406, -410, -411, -412, -427, -438 (and all variations),
753-400, -401, -405, -406, -410, -411, -412, -427

Table 117: 2 Channel Digital Input Modules

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

11.2.1.3 2 Channel Digital Input Module with Diagnostics

750-421, -424, -425
753-421, -424, -425

Table 118: 2 Channel Digital Input Module with Diagnostics

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

Table 119: Diagnostic data of 2 Channel Digital Input Module with Diagnostics in the Diagnostic Status Word

Diagnostic Status Word (Extract)					
EVTCODE 2	**EVTCODE 1**	**EVTCODE 0**	**CHNUM 2**	**CHNUM 1**	**CHNUM 0**
0	0	Diagnostic bit Signal channel 1	0	0	0
0	0	Diagnostic bit Signal channel 2	0	0	1

For a detailed description of the diagnostic status word: see Section „Diagnostics“ > … > „Diagnostic Status Word“.

11.2.1.4 2 Channel Digital Input Module with Diagnostics and Output Process Data

750-418
753-418

In addition to the process values in the input process image, the digital input module also supplies 4-bit data that is also represented in the output process image.

Table 120: 2 Channel Digital Input Module with Diagnostics and Output Process Data

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

Output Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Acknow-ledgement bit Q 2 Channel 2	Acknow-ledgement bit Q 1 Channel 1	0	0

Table 121: Diagnostic data 2 Channel Digital Input Module with Diagnostics in the Diagnostic Status Word and Output Process Data

Diagnostic Status Word (Extract)					
EVTCODE 2	**EVTCODE 1**	**EVTCODE 0**	**CHNUM 2**	**CHNUM 1**	**CHNUM 0**
0	0	Diagnostic bit Signal Channel 1	0	0	0
0	0	Diagnostic bit Signal Channel 2	0	0	1

For a detailed description of the diagnostic status word: see Section „Diagnostics“ > … > „Diagnostic Status Word“.

11.2.1.5 4 Channel Digital Input Modules

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433, -1420, -1421, -1422
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

Table 122: 4 Channel Digital Input Modules

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Data bit DI 4 Channel 4	Data bit DI 3 Channel 3	Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

11.2.1.6 8 Channel Digital Input Modules

750-430, -431, -436, -437, -1415, -1416, -1417
753-430, -431, -434

Table 123: 8 Channel Digital Input Modules

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Data bit DI 8 Channel 8	Data bit DI 7 Channel 7	Data bit DI 6 Channel 6	Data bit DI 5 Channel 5	Data bit DI 4 Channel 4	Data bit DI 3 Channel 3	Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

11.2.1.7 16 Channel Digital Input Modules

750-1400, -1402, -1405, -1406, -1407

Table 124: 16 Channel Digital Input Modules

Input Process Image															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Data bit DI 16 Channel 16	Data bit DI 15 Channel 15	Data bit DI 14 Channel 14	Data bit DI 13 Channel 13	Data bit DI 12 Channel 12	Data bit DI 11 Channel 11	Data bit DI 10 Channel 10	Data bit DI 9 Channel 9	Data bit DI 8 Channel 8	Data bit DI 7 Channel 7	Data bit DI 6 Channel 6	Data bit DI 5 Channel 5	Data bit DI 4 Channel 4	Data bit DI 3 Channel 3	Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

11.2.2 Digital Output Modules

Digital output modules use one bit of data per channel to control the output of the corresponding channel. These bits are mapped into the Output Process Image.

Digital output modules with diagnostics have one or more diagnostic bits available. The diagnostic bits are evaluated by the fieldbus coupler. In the event of a diagnostic message, the fieldbus coupler enters the state of the diagnostic bit in the diagnostic status word. The entries in the diagnostic status word are made channel-specific.

When analog output modules are also present in the node, the digital image data is always appended after the analog data in the Output Process Image, grouped into bytes.

11.2.2.1 1 Channel Digital Output Module with Input Process Data

750-523

In addition to the process value bit in the output process image, the digital output modules also supply 1 bit that is represented in the input process image. This status image shows "manual mode".

Table 125: 1 Channel Digital Output Module with Input Process Data

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						not used	Status bit "manual mode"

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						not used	controls DO 1 Channel 1

11.2.2.2 2 Channel Digital Output Modules

750-501, -502, -509, -512, -513, -514, -517, -535 (and all variations),
753-501, -502, -509, -512, -513, -514, -517

Table 126: 2 Channel Digital Output Modules

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						controls DO 2 Channel 2	controls DO 1 Channel 1

11.2.2.3 2 Channel Digital Input Modules with Diagnostics and Input Process Data

750-507 (-508), -522,
753-507

Table 127: 2 Channel Digital Input Modules with Diagnostics and Input Process Data

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						controls DO 2 Channel 2	controls DO 1 Channel 1

Table 128: Diagnostic data 2 Channel Digital Input Modules with Diagnostics in the Diagnostic Status Word

Diagnostic Status Word (Extract)					
EVTCODE 2	**EVTCODE 1**	**EVTCODE 0**	**CHNUM 2**	**CHNUM 1**	**CHNUM 0**
0	0	Diagnostic bit Signal Channel 1	0	0	0
0	0	Diagnostic bit Signal Channel 2	0	0	1

For a detailed description of the diagnostic status word: see Section „Diagnostics“ > … > „Diagnostic Status Word“.

750-506,
753-506

Table 129: 2 Channel Digital Input Modules with Diagnostics and Input Process Data 75x-506

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				0	0	controls DO 2 Channel 2	controls DO 1 Channel 1

Table 130: Diagnostic data 2 Channel Digital Input Modules with Diagnostics in the Diagnostic Status Word

Diagnostic Status Word (Extract)					
EVTCODE 2	**EVTCODE 1**	**EVTCODE 0**	**CHNUM 2**	**CHNUM 1**	**CHNUM 0**
0	Diagnostic bit 2 Signal Channel 1	Diagnostic bit 1 Signal Channel 1	0	0	0
0	Diagnostic bit 2 Signal Channel 2	Diagnostic bit 1 Signal Channel 2	0	0	1

For a detailed description of the diagnostic status word: see Section „Diagnostics“ > … > „Diagnostic Status Word“.

11.2.2.4 4 Channel Digital Output Modules

750-504, -516, -519, -531,
753-504, -516, -531, -540

Table 131: 4 Channel Digital Output Modules

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

11.2.2.5 4 Channel Digital Output Modules with Diagnostics and Input Process Data

750-532

In addition to the 4-bit process values in the output process image, the digital output modules also supply 4-bit data that is represented in the input process image. A diagnostic bit for each output channel indicates an overload, short circuit or wire breakage.

Table 132: 4 Channel Digital Output Modules with Diagnostics and Input Process Data

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

Table 133: 4 Channel Digital Output Modules with Diagnostics in the Diagnostic Status Word

Diagnostic Status Word (Extract)					
EVTCODE 2	**EVTCODE 1**	**EVTCODE 0**	**CHNUM 2**	**CHNUM 1**	**CHNUM 0**
0	0	Diagnostic bit S1 Signal Channel 1	0	0	0
0	0	Diagnostic bit S2 Signal Channel 2	0	0	1
0	0	Diagnostic bit S3 Signal Channel 3	0	1	0
0	0	Diagnostic bit S4 Signal Channel 4	0	1	1

Diagnostic bit = ‘0’ no Error
Diagnostic bit = ‘1’ overload, short circuit, or broken wire

For a detailed description of the diagnostic status word: see Section „Diagnostics“ > … > „Diagnostic Status Word“.

11.2.2.6 8 Channel Digital Output Module

750-530, -536, -1515, -1516
753-530, -534

Table 134: 8 Channel Digital Output Module

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
controls DO 8 Channel 8	controls DO 7 Channel 7	controls DO 6 Channel 6	controls DO 5 Channel 5	controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

11.2.2.7 8 Channel Digital Output Modules with Diagnostics and Input Process Data

750-537

In addition to the 8-bit process values in the output process image, the digital output modules also supply 8-bit data that is represented in the input process image. A diagnostic bit for each output channel indicates an overload, short circuit or wire breakage.

Table 135: 8 Channel Digital Output Modules with Diagnostics and Input Process Data

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
controls DO 8 Channel 8	controls DO 7 Channel 7	controls DO 6 Channel 6	controls DO 5 Channel 5	controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

Table 136: 8 Channel Digital Output Modules with Diagnostics in the Diagnostic Status Word

Diagnostic Status Word (Extract)					
EVTCODE 2	**EVTCODE 1**	**EVTCODE 0**	**CHNUM 2**	**CHNUM 1**	**CHNUM 0**
0	0	Diagnostic bit S1 Signal Channel 1	0	0	0
0	0	Diagnostic bit S2 Signal Channel 2	0	0	1
0	0	Diagnostic bit S3 Signal Channel 3	0	1	0
0	0	Diagnostic bit S4 Signal Channel 4	0	1	1
0	0	Diagnostic bit S5 Signal Channel 5	1	0	0
0	0	Diagnostic bit S6 Signal Channel 6	1	0	1
0	0	Diagnostic bit S7 Signal Channel 7	1	1	0
0	0	Diagnostic bit S8 Signal Channel 8	1	1	1

Diagnostic bit S = ‘0’ no Error
Diagnostic bit S = ‘1’ overload, short circuit, or broken wire

For a detailed description of the diagnostic status word: see Section „Diagnostics“ > … > „Diagnostic Status Word“.

11.2.2.8 16 Channel Digital Output Modules

750-1500, -1501, -1504, -1505

Table 137: 16 Channel Digital Output Modules

Output Process Image															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
controls DO 16 Channel 16	controls DO 15 Channel 15	controls DO 14 Channel 14	controls DO 13 Channel 13	controls DO 12 Channel 12	controls DO 11 Channel 11	controls DO 10 Channel 10	controls DO 9 Channel 9	controls DO 8 Channel 8	controls DO 7 Channel 7	controls DO 6 Channel 6	controls DO 5 Channel 5	controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

11.2.2.9 8 Channel Digital Input/Output Modules

750-1502, -1506

Table 138: 8 Channel Digital Input/Output Modules

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Data bit DI 8 Channel 8	Data bit DI 7 Channel 7	Data bit DI 6 Channel 6	Data bit DI 5 Channel 5	Data bit DI 4 Channel 4	Data bit DI 3 Channel 3	Data bit DI 2 Channel 2	Data bit DI 1 Channel 1

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
controls DO 8 Channel 8	controls DO 7 Channel 7	controls DO 6 Channel 6	controls DO 5 Channel 5	controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

11.2.3 Analog Input Modules

The hardware of an analog input module has 16 bits of measured analog data per channel. The 16 bits of analog data per channel are grouped as words and mapped in Intel format in the Input Process Image of the EtherCAT® fieldbus coupler.

Information

Information on the structure of control and status bytes
For detailed information on the structure of a particular I/O module's control/status bytes, please refer to that module's manual. Manuals for each module can be found on the Internet at www.wago.com.

11.2.3.1 1 Channel Analog Input Modules

750-491, (and all variations)

Table 139: 1 Channel Analog Input Modules

Input Process Image			
Offset	**Byte Destination**		**Description**
	High Byte	**Low Byte**	
0	D1	D0	Measured Value U_D
1	D3	D2	Measured Value U_{ref}

11.2.3.2 2 Channel Analog Input Modules

750-452, -454, -456, -461, -462, -464 (2-Channel Operation) -465, -466, -467, -469, -470, -472, -473, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -487, -492, (and all variations),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, -476, -477, -478, -479, -483, -492, (and all variations)

Table 140: 2 Channel Analog Input Modules

Input Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	D1	D0	Measured Value Channel 1
1	D3	D2	Measured Value Channel 2

11.2.3.3 2 Channel Analog Input Modules HART

750-482, -484, (and all variations),
753-482

The HART I/O module provides two different process images depending on the set operating mode.

For the pure analog values 4 mA ... 20 mA, the HART I/O module transmits 16 bit measured values per channel as an analog input module, which are mapped by word.

In operating mode "6 Byte Mailbox", the HART I/O module provides the fieldbus coupler / controller with a 12-byte input and output process image via a logical channel. For the control/status byte and the dummy byte, an acyclic channel (mailbox) for the process value communication is embedded in the process image, which occupies 6 bytes of data. This is followed by the measured values for channels 1 and 2.

HART commands are executed via the WAGO-IEC function blocks of the "WagoLibHart_0x.lib" library. The data is tunneled to the application via the mailbox and decoded by means of the library, so that the evaluation and processing takes place directly at the application level.

The operating mode is set using the WAGO-I / O-*CHECK* commissioning tool.

Table 141: 2-Channel Analog Input Modules HART

<table>
<tr><th colspan="4">Input Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2">Description</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>D1</td><td>D0</td><td>Measured Value Channel 1</td></tr>
<tr><td>1</td><td>D3</td><td>D2</td><td>Measured Value Channel 2</td></tr>
</table>

Table 142:: 2 Channel Analog Input Modules HART + 6 bytes Mailbox

<table>
<tr><th colspan="5">Input Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2" colspan="2">Description</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>Internal Use</td><td>S0</td><td>Internal used</td><td>Status byte</td></tr>
<tr><td>1</td><td>MBX_RES</td><td>MBX_RES</td><td rowspan="3" colspan="2">Response data from mailbox</td></tr>
<tr><td>2</td><td>MBX_RES</td><td>MBX_RES</td></tr>
<tr><td>3</td><td>MBX_RES</td><td>MBX_RES</td></tr>
<tr><td>4</td><td>D1</td><td>D0</td><td colspan="2">Measured Value Channel 1</td></tr>
<tr><td>5</td><td>D3</td><td>D2</td><td colspan="2">Measured Value Channel 2</td></tr>
</table>

<table>
<tr><th colspan="4">Output Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2">Description</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>-</td><td>C0</td><td>Control byte</td></tr>
<tr><td>1</td><td>MBX_REQ</td><td>MBX_REQ</td><td rowspan="3">Request data from mailbox</td></tr>
<tr><td>2</td><td>MBX_REQ</td><td>MBX_REQ</td></tr>
<tr><td>3</td><td>MBX_REQ</td><td>MBX_REQ</td></tr>
<tr><td>4</td><td>-</td><td>-</td><td rowspan="2">Not used</td></tr>
<tr><td>5</td><td>-</td><td>-</td></tr>
</table>

11.2.3.4 4 Channel Analog Input Modules

750-450, -453, -455, -457, -459, -460, -463, -464 (4-Channel Operation), -468, -471, -468, (and all variations),
753-453, -455, -457, -459

Table 143: 4 Channel Analog Input Modules

<table>
<tr><th colspan="4">Input Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2">Description</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>D1</td><td>D0</td><td>Measured Value Channel 1</td></tr>
<tr><td>1</td><td>D3</td><td>D2</td><td>Measured Value Channel 2</td></tr>
<tr><td>2</td><td>D5</td><td>D4</td><td>Measured Value Channel 3</td></tr>
<tr><td>3</td><td>D7</td><td>D6</td><td>Measured Value Channel 4</td></tr>
</table>

11.2.3.5 8 Channel Analog Input Modules

750-451, 750-458, 750-496, 750-497

Table 144: 8 Channel Analog Input Modules

Input Process Image			
Offset	**Byte Destination**		**Description**
	High Byte	**Low Byte**	
0	D1	D0	Measured Value Channel 1
1	D3	D2	Measured Value Channel 2
2	D5	D4	Measured Value Channel 3
3	D7	D6	Measured Value Channel 4
4	D9	D8	Measured Value Channel 5
5	D11	D10	Measured Value Channel 6
6	D13	D12	Measured Value Channel 7
7	D15	D14	Measured Value Channel 8

11.2.3.6 3-Phase Power Measurement Module

750-493

The above Analog Input Modules have a total of 9 bytes of user data in both the Input and Output Process Image (6 bytes of data and 3 bytes of control/status). The following tables illustrate the Input and Output Process Image, which has a total of 6 words mapped into each image.
Word alignment is applied.

Table 145: 3-Phase Power Measurement Module

Input Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	-	S0	Status byte 0
1	D1	D0	Input data word 1
2	-	S1	Status byte 1
3	D3	D2	Input data word 2
4	-	S2	Status byte 2
5	D5	D4	Input data word 3

Output Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	-	C0	Control byte 0
1	D1	D0	Output data word 1
2	-	C1	Control byte 1
3	D3	D2	Output data word 2
4	-	C2	Control byte 2
5	D5	D4	Output data word 3

750-494, -495, (and all variations)

The 3-Phase Power Measurement Modules 750-494, -495, (and all variations) have a total of 24 bytes of user data in both the Input and Output Process Image (16 bytes of data and 8 bytes of control/status).

Table 146: 3-Phase Power Measurement Modules 750-494, -495, (and all variations)

Input Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	S1	S0	Status word
1	S3	S2	Extended status word 1
2	S5	S4	Extended status word 2
3	S7	S6	Extended status word 3
4	D1	D0	Process value 1
5	D3	D2	
6	D5	D4	Process value 2
7	D7	D6	
8	D9	D8	Process value 3
9	D11	D10	
10	D13	D12	Process value 4
11	D15	D14	

Output Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	S1	S0	Control word
1	S3	S2	Extended control word 1
2	S5	S4	Extended control word 2
3	S7	S6	Extended control word 3
4	-	-	-
5	-	-	
6	-	-	-
7	-	-	
8	-	-	-
9	-	-	
10	-	-	-
11	-	-	

11.2.4 Analog Output Modules

The hardware of an analog output module has 16 bits of measured analog data per channel. The 16 bits of analog data per channel are grouped as words and mapped in Intel format in the Output Process Image of the EtherCAT® fieldbus coupler.

Information

Information on the structure of control and status bytes
For detailed information on the structure of a particular I/O module's control/status bytes, please refer to that module's manual. Manuals for each module can be found on the Internet at www.wago.com.

11.2.4.1 2 Channel Analog Output Modules

750-550, -552, -554, -556, -560, -562, 563, -585, -586, (and all variations),
753-550, -552, -554, -556

Table 147: 2 Channel Analog Output Modules

Output Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	D1	D0	Output Value Channel 1
1	D3	D2	Output Value Channel 2

11.2.4.2 4 Channel Analog Output Modules

750-553, -555, -557, -559,
753-553, -555, -557, -559

Table 148: 4 Channel Analog Output Modules

Output Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	D1	D0	Output Value Channel 1
1	D3	D2	Output Value Channel 2
2	D5	D4	Output Value Channel 3
3	D7	D6	Output Value Channel 4

11.2.4.3 8 Channel Analog Output Modules

750-597

Table 149: 8 Channel Analog Output Modules

Output Process Image			
Offset	Byte Destination		Description
	High Byte	Low Byte	
0	D1	D0	Output Value Channel 1
1	D3	D2	Output Value Channel 2
2	D5	D4	Output Value Channel 3
3	D7	D6	Output Value Channel 4
4	D9	D8	Output Value Channel 5
5	D11	D10	Output Value Channel 6
6	D13	D12	Output Value Channel 7
7	D15	D14	Output Value Channel 8

11.2.5 Specialty Modules

WAGO has a host of Specialty I/O modules that perform various functions. With individual modules beside the data bytes also the control/status byte is mapped in the process image.

The control/status byte is required for the bidirectional data exchange of the module with the higher-ranking control system. The control byte is transmitted from the control system to the module and the status byte from the module to the control system.
This allows, for example, setting of a counter with the control byte or displaying of overshooting or undershooting of the range with the status byte.

The control/status byte always is in the process image in the Low byte.

Information

Information about the structure of the Control/Status byte
For detailed information about the structure of a particular module's control/status byte, please refer to that module's manual. Manuals for each module can be found on the Internet under: www.wago.com.

11.2.5.1 Counter Modules

750-404, (and all variations except of /000-005),
753-404, -404/000-003

The above Counter Modules have a total of 5 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 1 byte of control/status). The counter value is supplied as 32 bits. The following tables illustrate the Input and Output Process Image, which has a total of 3 words mapped into each image. Word alignment is applied.

Table 150: Counter Modules 750-404, (and all variations except of /000-005), 753-404, -404/000-003)

Input Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	S	Status byte
1	D1	D0	Counter value
2	D3	D2	

Output Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	C	Control byte
1	D1	D0	Counter setting value
2	D3	D2	

750-404/000-005,
753-404/000-005

The above Counter Modules have a total of 5 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 1 byte of control/ status). The two counter values are supplied as 32 bits. The following tables illustrate the Input and Output Process Image, which has a total of 3 words mapped into each image. Word alignment is applied.

Table 151: Counter Modules 750-404/000-005, 753-404/000-005

Input Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	S	Status byte
1	D1	D0	Counter Value of Counter 1
2	D3	D2	Counter Value of Counter 2

Output Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	C	Control byte
1	D1	D0	Counter Setting Value of Counter 1
2	D3	D2	Counter Setting Value of Counter 2

750-633

The above Counter Module has a total of 5 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 1 byte of control/ status). The following tables illustrate the Input and Output Process Image, which has a total of 3 words mapped into each image. Word alignment is applied.
The meaning of the output data depends on the set operating mode:
1 Up counter with enable input
2 Up/down counter with U/D input
3 Frequency counter
4 Gate time counter

Table 152: Counter Modules 750-633

Input Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	S	Status byte
1	D1	D0	Counter Value
2	D3	D2	

Output Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	C	Control byte
1	D1	D0	Counter Setting Value [1,2] watchdog time [3] reserved [4]
2	D3	D2	Counter Setting Value [1,2] reserved [3] reserved [4]

[1,2] Up counter with enable input, Up /down counter with U / D input
[3] Frequency counter
[4] Gate time counter

750-638,
753-638

The above Counter Modules have a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 2 bytes of control/status). The two counter values are supplied as 16 bits. The following tables illustrate the Input and Output Process Image, which has a total of 4 words mapped into each image. Word alignment is applied.

Table 153: Counter Modules 750-638, 753-638

Input Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	S0	Status byte of Counter 1
1	D1	D0	Counter Value of Counter 1
2	-	S1	Status byte of Counter 2
3	D3	D2	Counter Value of Counter 2

Output Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	C0	Control byte of Counter 1
1	D1	D0	Counter Setting Value of Counter 1
2	-	C1	Control byte of Counter 2
3	D3	D2	Counter Setting Value of Counter 2

11.2.5.2 Pulse Width Modules

750-511, (and all variations),
753-511

The above Pulse Width modules have a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of channel data and 2 bytes of control/ status). The two channel values are supplied as 16 bits. Each channel has its own control/status byte. The following table illustrates the Input and Output Process Image, which has a total of 4 words mapped into each image. Word alignment is applied.

Table 154: Pulse Width Modules 750-511, (and all variations), 753-511

Input and Output Process			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	C0/S0	Control/Status byte of Channel 1
1	D1	D0	Data Value of Channel 1
2	-	C1/S1	Control/Status byte of Channel 2
3	D3	D2	Data Value of Channel 2

11.2.5.3 Serial Interface Modules with Alternative Data Format

750-650, (and the variations /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (and the variations /000-001, -002, -003),
750-653, (and the variations /000-002, -007),
753-650, -653

Note

The process image of the / 003-000-variants depends on the parameterized operating mode!
With the freely parameterizable variations /003 000 of the serial interface modules, the desired operating mode can be set. Dependent on it, the process image of these modules is then the same, as from the appropriate variation.

The above Serial Interface Modules with alternative data format have a total of 4 bytes of user data in both the Input and Output Process Image (3 bytes of serial data and 1 byte of control/status). The following table illustrates the Input and

Output Process Image, which have a total of 2 words mapped into each image. Word alignment is applied.

Table 155: Serial Interface Modules with Alternative Data Format

Input and Output Process Image				
Offset	**Byte Designation**		**Description**	
	High Byte	**Low Byte**		
0	D0	C/S	Data byte	Control/status byte
1	D2	D1	Data bytes	

11.2.5.4 Serial Interface Modules with Standard Data Format

750-650/000-001, -014, -015, -016,
750-653/000-001, -006

The above Serial Interface Modules with Standard Data Format have a total of 6 bytes of user data in both the Input and Output Process Image (5 bytes of serial data and 1 byte of control/status). The following table illustrates the Input and Output Process Image, which have a total of 3 words mapped into each image. Word alignment is applied.

Table 156: Serial Interface Modules with Standard Data Format

Input and Output Process Image				
Offset	**Byte Designation**		**Description**	
	High Byte	**Low Byte**		
0	D0	C/S	Data byte	Control/status byte
1	D2	D1	Data bytes	
2	D4	D3		

11.2.5.5 Serial Interface Modules

750-652,
753-652

The size of the process image for the Serial Interface Module can be adjusted to 12, 24 or 48 bytes.
It consists of two status bytes (input) or control bytes (output) and the process data with a size of 6 to 46 bytes.

Thus, each Serial Interface Module uses between 8 and 48 bytes in the process image. The sizes of the input and output process images are always the same.

The process image sizes are set with the startup tool WAGO-I/O-*CHECK*.

Table 157: Serial Interface Modules 750-652, 753-652

Input and Output Process Image					
Process image size	**Offset**	**Byte Designation**		**Description**	
		High Byte	**Low Byte**		
8 bytes	0	C1/S1	C0/S0	Control/Status byte C1/S1	Control/Status byte C0/S0
	1	D1	D0	Prozess data (6-46 bytes)	
	2	D3	D2		
	3	D5	D4		
24 bytes*	4	D7	D6		
	…				
	11	D21	D20		
48 bytes	12	D23	D22		
	…				
	23	D45	D44		

*) Factory setting

11.2.5.6 Data Exchange Module

750-654, -654/000-001

The Data Exchange modules have a total of 4 bytes of user data in both the Input and Output Process Image. The following tables illustrate the Input and Output Process Image, which has a total of 2 words mapped into each image.
Word alignment is applied.

Table 158: Data Exchange Module 750-654, -654/000-001

Input and Output Process Image			
Offset	**Byte Designation**		**Description**
	High Byte	**Low Byte**	
0	D1	D0	Data bytes
1	D3	D2	

11.2.5.7 SSI Transmitter Interface Modules

750-630 (and all variations)

Note

The process image of the / 003-000-variants depends on the parameterized operating mode!
The operating mode of the configurable /003-000 I/O module versions can be set. Based on the operating mode, the process image of these I/O modules is then the same as that of the respective version.

The above SSI Transmitter Interface modules have a total of 4 bytes of user data in the Input Process Image, which has 2 words mapped into the image.
Word alignment is applied.

Table 159: SSI Transmitter Interface Modules

Input Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	D1	D0	Data bytes
1	D3	D2	

11.2.5.8 Incremental Encoder Interface Modules

750-631/000-004, -010, -011

The above Incremental Encoder Interface modules have 5 bytes of input data and 3 bytes of output data. The following tables illustrate the Input and Output Process Image, which have 4 words into each image. Word alignment is applied.

Table 160: Incremental Encoder Interface Modules 750-631/000-004, --010, -011

Input Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	-	S	not used	Status byte
1	D1	D0	Counter word	
2	-	-	not used	
3	D4	D3	Latch word	

Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	-	C	not used	Control byte
1	D1	D0	Counter setting word	
2	-	-	not used	
3	-	-	not used	

750-634

The above Incremental Encoder Interface module has 5 bytes of input data (6 bytes in cycle duration measurement mode) and 3 bytes of output data. The following tables illustrate the Input and Output Process Image, which has 4 words mapped into each image. Word alignment is applied.

Table 161: Incremental Encoder Interface Modules 750-634

Input Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	-	S	not used	Status byte
1	D1	D0	Counter word	
2	-	(D2) *)	not used	(Periodic time)
3	D4	D3	Latch word	

*) If cycle duration measurement mode is enabled in the control byte, the cycle duration is given as a 24-bit value that is stored in D2 together with D3/D4.

Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	-	C	not used	Control byte
1	D1	D0	Counter setting word	
2	-	-	not used	
3	-	-		

750-637, (and all variations)

The above Incremental Encoder Interface Module has a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of encoder data and 2 bytes of control/status). The following table illustrates the Input and Output Process Image, which have 4 words mapped into each image. Word alignment is applied.

Table 162: Incremental Encoder Interface Modules 750-637, (and all variations)

Input and Output Process Image			
Offset	Byte Designation		Description
	High Byte	Low Byte	
0	-	C0/S0	Control/Status byte of Channel 1
1	D1	D0	Data Value of Channel 1
2	-	C1/S1	Control/Status byte of Channel 2
3	D3	D2	Data Value of Channel 2

Digital Pulse Interface module

750-635,
753-635

The above Digital Pulse Interface module has a total of 4 bytes of user data in both the Input and Output Process Image (3 bytes of module data and 1 byte of control/status). The following table illustrates the Input and Output Process Image, which have 2 words mapped into each image. Word alignment is applied.

Table 163: Digital Pulse Interface Modules 750-635, 753-635

Input and Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	D0	C0/S0	Data byte	Control/status byte
1	D2	D1	Data bytes	

11.2.5.9 DC-Drive Controller

750-636, -636/000-700, -636/000-800

The DC-Drive Controller maps 6 bytes into both the input and output process image. The data sent and received are stored in up to 4 input and output bytes (D0 ... D3). Two control bytes (C0, C1) and two status bytes (S0/S1) are used to control the I/O module and the drive.

In addition to the position data in the input process image (D0 … D3), it is possible to display extended status information (S2 … S5). Then the three control bytes (C1 … C3) and status bytes (S1 … S3) are used to control the data flow.

Bit 3 of control byte C1 (C1.3) is used to switch between the process data and the extended status bytes in the input process image (Extended Info_ON). Bit 3 of status byte S1 (S1.3) is used to acknowledge the switching process.

Table 164: DC-Drive Controller 750-636, -636/000-700, -636/000-800

Input Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	S1	S0	Status byte S1	Status byte S0
1	D1*) / S3**)	D0*) / S2**)	Actual position*) / Extended status byte S3**)	Actual position (LSB) / Extended status byte S2**)
2	D3*) / S5**)	D2*) / S4**)	Actual position (MSB) / Extended status byte S3**)	Actual position*) / Extended status byte S4**)

*) ExtendedInfo_ON = ‘0’.
**) ExtendedInfo_ON = ‘1’.

Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	C1	C0	Control byte C1	Control byte C0
1	D1	D0	Setpoint position	Setpoint position (LSB)
2	D3	D2	Setpoint position (MSB)	Setpoint position

11.2.5.10 Stepper Controller

750-670, -671, -672

The Stepper controller provides the fieldbus coupler/controller 12 bytes input and output process image via 1 logical channel. The data to be sent and received are stored in up to 7 output bytes (D0 … D6) and 7 input bytes (D0 … D6), depending on the operating mode.

Output byte D0 and input byte D0 are reserved and have no function assigned.

One I/O module control and status byte (C0, S0) and 3 application control and status bytes (C1 ... C3, S1 ... S3) provide the control of the data flow.

Switching between the two process images is conducted through bit 5 in the control byte (C0 (C0.5). Activation of the mailbox is acknowledged by bit 5 of the status byte S0 (S0.5).

Table 165: Stepper Controller 750-670, -671, -672

Input and Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	Reserviert	C0/S0	reserved	Control/Status byte C0/S0
1	D1	D0	Process data*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	S3	D6	Control/Status byte C3/S3	Process data*) / reserved**)
5	C1/S1	C2/S2	Control/Status byte C1/S1	Control/Status byte C2/S2

*) Cyclic process image (Mailbox disabled)
**) Mailbox process image (Mailbox activated)

11.2.5.11 RTC Module

750-640

The RTC Module has a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of module data and 1 byte of control/status and 1 byte ID for command). The following table illustrates the Input and Output Process Image, which have 3 words mapped into each image. Word alignment is applied.

Table 166: RTC Module 750-640

Input and Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	ID	C/S	Command byte	Control/status byte
1	D1	D0	Data bytes	
2	D3	D2		

11.2.5.12 DALI/DSI Master Module

750-641

The DALI/DSI Master module has a total of 6 bytes of user data in both the Input and Output Process Image (5 bytes of module data and 1 byte of control/status). The following tables illustrate the Input and Output Process Image, which have 3 words mapped into each image. Word alignment is applied.

Table 167: DALI/DSI Master Module 750-641

Input Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	D0	S	DALI Response	Status byte
1	D2	D1	Message 3	DALI Address
2	D4	D3	Message 1	Message 2

Output Process Image				
Offset	Byte Designation		Description	
	High Byte	Low Byte		
0	D0	C	DALI command, DSI dimming value	Control byte
1	D2	D1	Parameter 2	DALI Address
2	D4	D3	Command extension	Parameter 1

11.2.5.13 DALI Multi-Master Module

753-647

The DALI Multi-Master module occupies a total of 24 bytes in the input and output range of the process image.

The DALI Multi-Master module can be operated in “Easy” mode (default) and “Full” mode. “Easy” mode is used to transmit simply binary signals for lighting control. Configuration or programming via DALI master module is unnecessary in "Easy" mode.

Changes to individual bits of the process image are converted directly into DALI commands for a pre-configured DALI network. 22 bytes of the 24-byte process image can be used directly for switching of electronic ballasts (ECG), groups or scenes in “Easy” mode. Switching commands are transmitted via DALI and group addresses, where each DALI and each group address is represented by a 2-bit pair.

In full mode, the 24 bytes of the process image are used to tunnel a protocol using a mailbox interface. The process image consists of 1 byte for control / status and 23 bytes for the acyclic data.

The structure of the process data is described in detail in the following tables.

Table 168: DALI Multi-Master Module 753-647 in the “Easy” Mode

<table>
<tr><th colspan="5">Input Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Designation</th><th rowspan="2" colspan="2">Note</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>-</td><td>S</td><td>res.</td><td>Status, activate broadcast
Bit 0: 1-/2-button mode
Bit 2: Broadcast status ON/OFF
Bit 1,3-7: -</td></tr>
<tr><td>1</td><td>DA4…DA7</td><td>DA0…DA3</td><td colspan="2" rowspan="8">Bit pair for DALI address DA0:
Bit 1: Bit set = ON
Bit not set = OFF
Bit 2: Bit set = Error
Bit not set = No error
Bit pairs DA1 … DA63 similar to DA0.</td></tr>
<tr><td>2</td><td>DA12…DA15</td><td>DA8…DA11</td></tr>
<tr><td>3</td><td>DA20…DA23</td><td>DA16…DA19</td></tr>
<tr><td>4</td><td>DA28…DA31</td><td>DA24…DA27</td></tr>
<tr><td>5</td><td>DA36…DA39</td><td>DA32…DA35</td></tr>
<tr><td>6</td><td>DA44…DA47</td><td>DA40…DA43</td></tr>
<tr><td>7</td><td>DA52…DA55</td><td>DA48…DA51</td></tr>
<tr><td>8</td><td>DA60…DA63</td><td>DA56…DA59</td></tr>
<tr><td>9</td><td>GA4…GA7</td><td>GA0…GA3</td><td colspan="2" rowspan="2">Bit pair for DALI group address GA0:
Bit 1: Bit set = ON
Bit not set = OFF
Bit 2: Bit set = Error
Bit not set = No error
Bit pairs GA1 … GA15 similar to GA0.</td></tr>
<tr><td>10</td><td>GA12…GA15</td><td>GA8…GA11</td></tr>
<tr><td>11</td><td>-</td><td>-</td><td colspan="2">Not used</td></tr>
</table>

DA = DALI address
GA = Group address

Output Process Image				
Offset	Byte Designation		Note	
	High Byte	Low Byte		
0	-	S	res.	Broadcast ON/OFF and activate: Bit 0: Broadcast ON Bit 1: Broadcast OFF Bit 2: Broadcast ON/OFF/dimming Bit 3: Broadcast short ON/OFF Bits 4 … 7: reserved
1	DA4…DA7	DA0…DA3	Bit pair for DALI address DA0: Bit 1: short: DA switch ON long: dimming, brighter Bit 2: short: DA switch OFF long: dimming, darker Bit pairs DA1 … DA63 similar to DA0.	
2	DA12…DA15	DA8…DA11		
3	DA20…DA23	DA16…DA19		
4	DA28…DA31	DA24…DA27		
5	DA36…DA39	DA32…DA35		
6	DA44…DA47	DA40…DA43		
7	DA52…DA55	DA48…DA51		
8	DA60…DA63	DA56…DA59		
9	GA4…GA7	GA0…GA3	Bit pair for DALI group address GA0: Bit 1: short: GA switch ON long: dimming, brighter Bit 2: short: GA switch OFF long: dimming, darker Bit pairs GA1 … GA15 similar to GA0.	
10	GA12…GA15	GA8…GA11		
11	Bit 8…15	Bit 0…7	Switch scene 0…15	

DA = DALI address
GA = Group address

Table 169: DALI Multi-Master Module 753-647 in the “Full” Mode

Input and Output Process Image				
Offset	Byte Designation		Note	
	High Byte	Low Byte		
0	MBX_C/S	C0/S0	Mailbox control/status byte	control/status byte
1	MBX1	MBX0	Mailbox	
2	MBX3	MBX2		
3	MBX5	MBX4		
4	MBX7	MBX6		
5	MBX9	MBX8		
6	MBX11	MBX10		
7	MBX13	MBX12		
8	MBX15	MBX14		
9	MBX17	MBX16		
10	MBX19	MBX18		
11	MBX21	MBX20		

11.2.5.14 EnOcean Radio Receiver

750-642

The EnOcean radio receiver has a total of 4 bytes of user data in both the Input and Output Process Image (3 bytes of module data and 1 byte of control/status). The following tables illustrate the Input and Output Process Image, which have 2 words mapped into each image. Word alignment is applied.

Table 170: EnOcean Radio Receiver 750-642

Input Process Image				
Offset	**Byte Destination**		**Description**	
	High Byte	**Low Byte**		
0	D0	S	Data byte	Status byte
1	D2	D1	Data bytes	

Output Process Image				
Offset	**Byte Destination**		**Description**	
	High Byte	**Low Byte**		
0	-	C	not used	Control byte
1	-	-	not used	

11.2.5.15 MP Bus Master Module

750-643

The MP Bus Master Module has a total of 8 bytes of user data in both the Input and Output Process Image (6 bytes of module data and 2 bytes of control/status). The following table illustrates the Input and Output Process Image, which have 4 words mapped into each image. Word alignment is applied.

Table 171: MP Bus Master Module 750-643

Input and Output Process Image				
Offset	**Byte Destination**		**Description**	
	High Byte	**Low Byte**		
0	C1/S1	C0/S0	Extended Control/ Status byte	Control/status byte
1	D1	D0	Data bytes	
2	D3	D2		
3	D5	D4		

11.2.5.16 *Bluetooth®* RF-Transceiver

750-644

The size of the process image for the *Bluetooth®* module can be adjusted to 12, 24 or 48 bytes.
It consists of one control byte (input) or status byte (output); an empty byte; an overlay able mailbox with a size of 6, 12 or 18 bytes (mode 2); and the *Bluetooth®* process data with a size of 4 to 46 bytes.
Thus, each *Bluetooth®* module uses between 12 and 48 bytes in the process image. The sizes of the input and output process images are always the same.

The first byte contains the control/status byte; the second contains an empty byte.

Process data attach to this directly when the mailbox is hidden. When the mailbox is visible, the first 6, 12 or 18 bytes of process data are overlaid by the mailbox data, depending on their size. Bytes in the area behind the optionally visible mailbox contain basic process data. The internal structure of the *Bluetooth®* process data can be found in the documentation for the *Bluetooth®* 750-644 RF Transceiver.

The mailbox and the process image sizes are set with the startup tool WAGO-I/O-*CHECK*.

Table 172: *Bluetooth®* RF-Transceiver 750-644

<table>
<tr><th colspan="6">Input and Output Process Image</th></tr>
<tr><th rowspan="2">Process image size</th><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2" colspan="2">Description</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td rowspan="4">12 bytes</td><td>0</td><td>-</td><td>C0/S0</td><td>not used</td><td>Control/status byte</td></tr>
<tr><td>1</td><td>D1</td><td>D0</td><td rowspan="9" colspan="2">Mailbox (0, 6, 12 or 18 words)/ Process data (4 … 46 words)</td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
<tr><td>5</td><td>D9</td><td>D8</td></tr>
<tr><td rowspan="3">24 bytes</td><td>6</td><td>D11</td><td>D10</td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
<tr><td>11</td><td>D21</td><td>D20</td></tr>
<tr><td rowspan="3">48 bytes*)</td><td>12</td><td>D23</td><td>D22</td></tr>
<tr><td>...</td><td>...</td><td>...</td></tr>
<tr><td>23</td><td>D45</td><td>D44</td></tr>
</table>

*) Factory Setting

11.2.5.17 Vibration Velocity/Bearing Condition Monitoring VIB I/O

750-645

The Vibration Velocity/Bearing Condition Monitoring VIB I/O has a total of 12 bytes of user data in both the Input and Output Process Image (8 bytes of module data and 4 bytes of control/status). The following table illustrates the Input and Output Process Image, which have 8 words mapped into each image. Word alignment is applied.

Table 173: Vibration Velocity/Bearing Condition Monitoring VIB I/O 750-645

Input and Output Process Image				
Offset	Byte Destination		Description	
	High Byte	Low Byte		
0	-	C0/S0	not used	Control/status byte (log. Channel 1, Sensor input 1)
1	D1	D0	Data bytes (log. Channel 1, Sensor input 1)	
2	-	C1/S1	not used	Control/status byte (log. Channel 2, Sensor input 2)
3	D3	D2	Data bytes (log. Channel 2, Sensor input 2)	
4	-	C2/S2	not used	Control/status byte (log. Channel 3, Sensor input 1)
5	D5	D4	Data bytes (log. Channel 3, Sensor input 3)	
6	-	C3/S3	not used	Control/status byte (log. Channel 4, Sensor input 2)
7	D7	D6	Data bytes (log. Channel 4, Sensor input 2)	

11.2.5.18 AS-interface Master Module

750-655,
753-655

The length of the process image of the AS-interface master module can be set to fixed sizes of 12, 20, 24, 32, 40 or 48 bytes.
It consists of a control or status byte, a mailbox with a size of 0, 6, 10, 12 or 18 bytes and the AS-interface process data, which can range from 0 to 46 bytes.

The AS-interface master module has a total of 6 to maximally 24 words data in both the Input and Output Process Image. Word alignment is applied.

The first Input and output word, which is assigned to an AS-interface master module, contains the status / control byte and one empty byte.
Subsequently the mailbox data are mapped, when the mailbox is permanently superimposed (Mode 1).

In the operating mode with suppressible mailbox (Mode 2), the mailbox and the cyclical process data are mapped next.
The following words contain the remaining process dat.

The mailbox and the process image sizes are set with the startup tool WAGO-I/O-*CHECK*.

Table 174: AS-interface Master Module 750-655, 753-655

Input and Output Process Image					
Process image size	**Offset**	**Byte Designation**		**Description**	
		High Byte	**Low Byte**		
12 bytes	0	-	C0/S0	Not used	Control-/ Status byte
	1	D1	D0	Mailbox (0, 6, 10, 12 or 18 bytes)/ Process data (0-46 bytes)	
	…				
	5	D9	D8		
20 bytes	6	D11	D10		
	…				
	9	D17	D16		
24 bytes *	10	D19	D18		
	11	D21	D20		
32 bytes	12	D23	D22		
	…				
	15	D29	D28		
40 bytes	16	D31	D30		
	…				
	19	D37	D36		
48 bytes	12	D39	D38		
	…				
	23	D45	D44		

*) Factory Setting

11.2.6 System Modules

11.2.6.1 System Modules with Diagnostics

750-606

The modules provide 2 bits of diagnostics in the Input Process Image for monitoring of the internal power supply.

Table 175: System Modules with Diagnostics 750-606

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnostics bit S_out	Diagnostics bit S_in

750-610, -611

The modules provide 2 bits of diagnostics in the Input Process Image for monitoring of the internal power supply.

Table 176: System Modules with Diagnostics 750-610, -611

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnostic bit S 2 Fuse	Diagnostic bit S 1 Fuse

11.2.6.2 Filter Module

750-624/020-002, -626/020-002

The Filter Module 750-624/020-002 and 750-626/020-002 equipped with surge suppression for the field side power supply have a total of 8 bits in both the Input and Output Process Image.

Table 177: Filter Modules 750-624/020-002, 750-626/020-002

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0V_MA	0V_PA	24V_MA	24V_PA	not used	PWR_DIAG	not used	VAL

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
not used	not used	not used	not used	not used	not used	not used	GFT

Note

Diagnostic information is not in the diagnostic status word!
Note that the diagnostic information about this I/O module is not entered in the diagnostic status word.

11.2.6.3 Binary Space Module

750-622

The Binary Space Modules behave alternatively like 2 channel digital input modules or output modules and seize depending upon the selected settings 1, 2, 3 or 4 bits per channel. According to this, 2, 4, 6 or 8 bits are occupied then either in the process input or the process output image.

Table 178: Binary Space Module 750-622 (with Behavior like 2 Channel Digital Input)

Input and Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(Data bit DI 8)	(Data bit DI 7)	(Data bit DI 6)	(Data bit DI 5)	(Data bit DI 4)	(Data bit DI 3)	Data bit DI 2	Data bit DI 1

12 Use in Hazardous Environments

The **WAGO-I/O-SYSTEM 750** (electrical equipment) is designed for use in Zone 2 hazardous areas and shall be used in accordance with the marking and installation regulations.

The following sections include both the general identification of components (devices) and the installation regulations to be observed. The individual subsections of the “Installation Regulations” section must be taken into account if the I/O module has the required approval or is subject to the range of application of the ATEX directive.

12.1 Marking Configuration Examples

12.1.1 Marking for Europe According to ATEX and IECEx

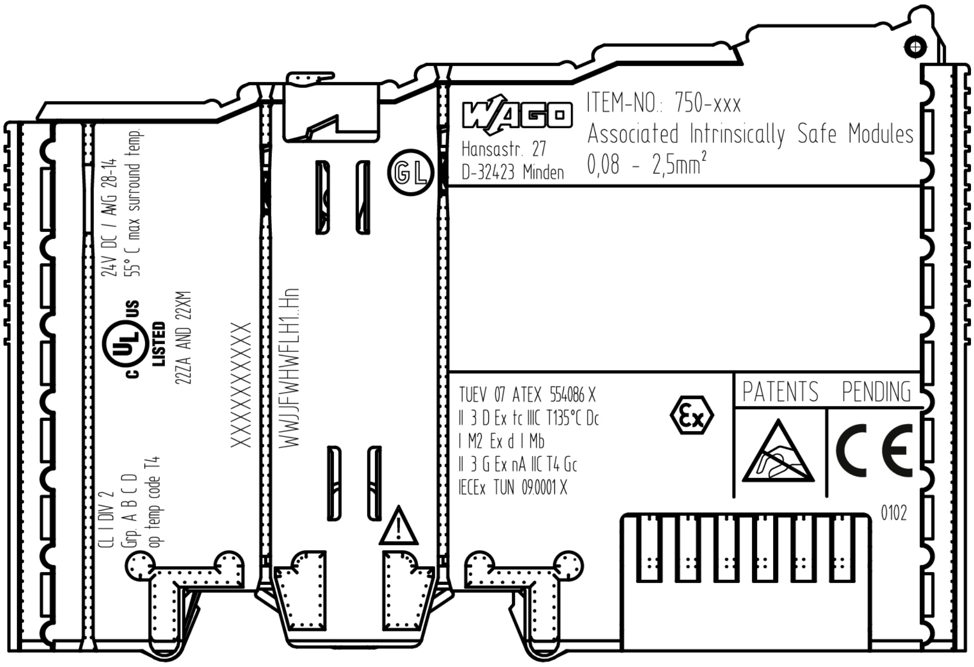


Figure 49: Marking Example According to ATEX and IECEx

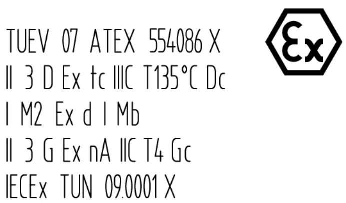


Figure 50: Text Detail – Marking Example According to ATEX and IECEx

Table 179: Description of Marking Example According to ATEX and IECEx

Marking	Description
TUEV 07 ATEX 554086 X IECEx TUN 09.0001 X	Approving authority resp. certificate numbers
Dust	
II	Equipment group: All except mining
3 D	Category 3 (Zone 22)
Ex	Explosion protection mark
tc	Type of protection: Protection by enclosure
IIIC	Explosion group of dust
T135°C	Max. surface temperature of the enclosure (without a dust layer)
Dc	Equipment protection level (EPL)
Mining	
I	Equipment group: Mining
M2	Category: High level of protection
Ex	Explosion protection mark
d	Type of protection: Flameproof enclosure
I	Explosion group for electrical equipment for mines susceptible to firedamp
Mb	Equipment protection level (EPL)
Gases	
II	Equipment group: All except mining
3 G	Category 3 (Zone 2)
Ex	Explosion protection mark
nA	Type of protection: Non-sparking equipment
IIC	Explosion group of gas and vapours
T4	Temperature class: Max. surface temperature 135 °C
Gc	Equipment protection level (EPL)

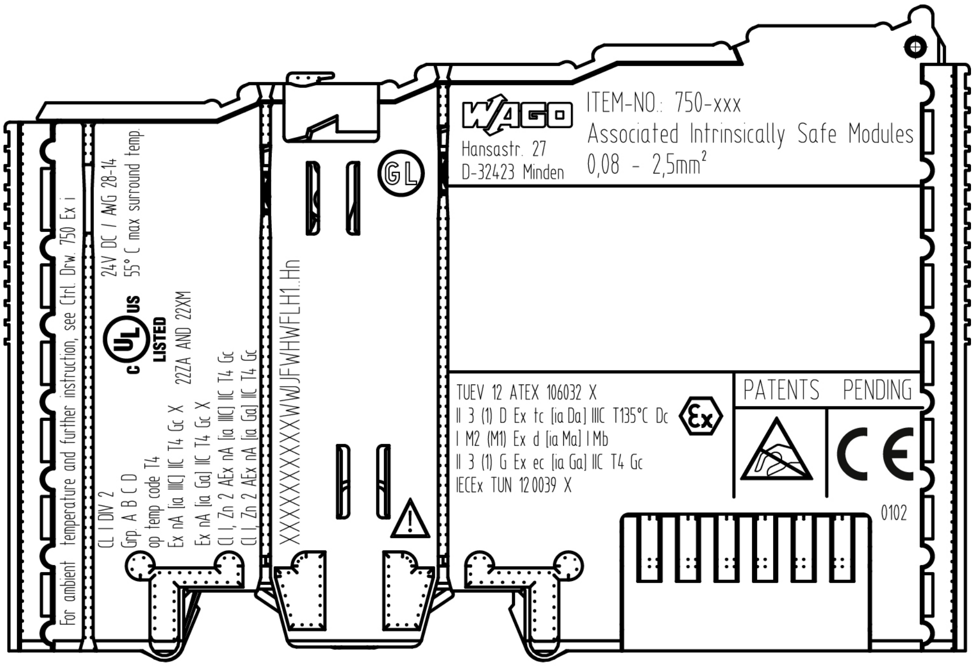


Figure 51: Marking Example for Approved Ex i I/O Module According to ATEX and IECEx

TUEV 12 ATEX 106032 X
II 3 (1) D Ex tc [ia Da] IIIC T135°C Dc
I M2 (M1) Ex d [ia Ma] I Mb
II 3 (1) G Ex ec [ia Ga] IIC T4 Gc
IECEx TUN 12 0039 X

Figure 52: Text Detail – Marking Example for Approved Ex i I/O Module According to ATEX and IECEx

Table 180: Description of Marking Example for Approved Ex i I/O Module According to ATEX and IECEx

Marking	Description
TUEV 12 ATEX 106032 X IECEx TUN 12 0039 X	Approving authority resp. certificate numbers
Dust	
II	Equipment group: All except mining
3 (1) D	Category 3 (Zone 22) equipment containing a safety device for a category 1 (Zone 20) equipment
Ex	Explosion protection mark
tc	Type of protection: Protection by enclosure
[ia Da]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety circuits for use in Zone 20
IIIC	Explosion group of dust
T135°C	Max. surface temperature of the enclosure (without a dust layer)
Dc	Equipment protection level (EPL)
Mining	
I	Equipment Group: Mining
M2 (M1)	Category: High level of protection with electrical circuits which present a very high level of protection
Ex	Explosion protection mark
d	Type of protection: Flameproof enclosure
[ia Ma]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety electrical circuits
I	Explosion group for electrical equipment for mines susceptible to firedamp
Mb	Equipment protection level (EPL)
Gases	
II	Equipment group: All except mining
3 (1) G	Category 3 (Zone 2) equipment containing a safety device for a category 1 (Zone 0) equipment
Ex	Explosion protection mark
ec	Equipment protection by increased safety “e”
[ia Ga]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety circuits for use in Zone 0
IIC	Explosion group of gas and vapours
T4	Temperature class: Max. surface temperature 135 °C
Gc	Equipment protection level (EPL)

12.1.2 Marking for the United States of America (NEC) and Canada (CEC)

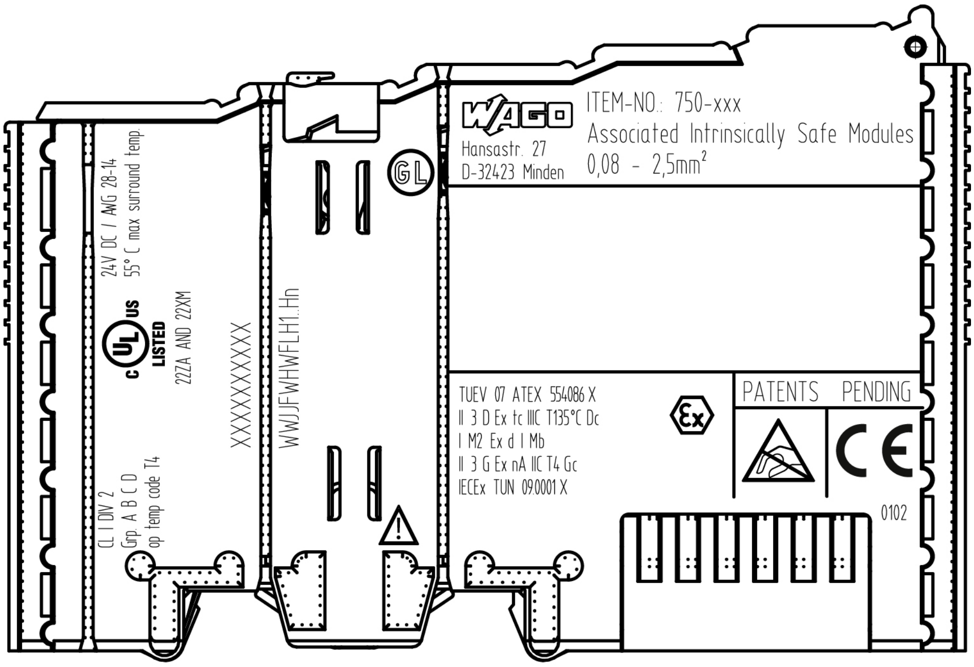


Figure 53: Marking Example According to NEC

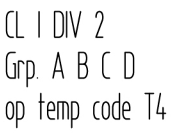


Figure 54: Text Detail – Marking Example According to NEC 500

Table 181: Description of Marking Example According to NEC 500

Marking	Description
CL I	Explosion protection (gas group)
DIV 2	Area of application
Grp. A B C D	Explosion group (gas group)
op temp code T4	Temperature class

Cl I, Zn 2 AEx nA [ia Ga] IIC T4 Gc

Figure 55: Text Detail – Marking Example for Approved Ex i I/O Module According to NEC 505

Table 182: Description of Marking Example for Approved Ex i I/O Module According to NEC 505

Marking	Description
Cl I,	Explosion protection group
Zn 2	Area of application
AEx	Explosion protection mark
nA	Type of protection
[ia Ga]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety circuits for use in Zone 20
IIC	Group
T4	Temperature class
Gc	Equipment protection level (EPL)

Cl I, Zn 2 AEx nA [ia IIIC] IIC T4 Gc

Figure 56: Text Detail – Marking Example for Approved Ex i I/O Module According to NEC 506

Table 183: Description of Marking Example for Approved Ex i I/O Modules According to NEC 506

Marking	Description
Cl I,	Explosion protection group
Zn 2	Area of application
AEx	Explosion protection mark
nA	Type of protection
[ia IIIC]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety circuits for use in Zone 20
IIC	Group
T4	Temperature class
Gc	Equipment protection level (EPL)

Ex nA [ia IIIC] IIC T4 Gc X
Ex nA [ia Ga] IIC T4 Gc X

Figure 57: Text Detail – Marking Example for Approved Ex i I/O Modules According to CEC 18 attachment J

Table 184: Description of Marking Example for Approved Ex i I/O Modules According to CEC 18 attachment J

Marking	Description
Dust	
Ex	Explosion protection mark
nA	Type of protection
[ia IIIC]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety circuits for use in Zone 20
IIC	Group
T4	Temperature class
Gc	Equipment protection level (EPL)
X	Symbol used to denote specific conditions of use
Gases	
Ex	Explosion protection mark
nA	Type of protection
[ia Ga]	Type of protection and equipment protection level (EPL): Associated apparatus with intrinsic safety circuits for use in Zone 0
IIC	Group
T4	Temperature class
Gc	Equipment protection level (EPL)
X	Symbol used to denote specific conditions of use

12.2 Installation Regulations

For the installation and operation of electrical equipment in hazardous areas, the valid national and international rules and regulations which are applicable at the installation location must be carefully followed.

12.2.1 Special Notes Regarding Explosion Protection

The following warning notices are to be posted in the immediately proximity of the WAGO-I/O-SYSTEM 750 (hereinafter “product”):

WARNING – DO NOT REMOVE OR REPLACE FUSED WHILE ENERGIZED!

WARNING – DO NOT DISCONNECT WHILE ENERGIZED!

WARNING – ONLY DISCONNECT IN A NON-HAZARDOUS AREA!

Before using the components, check whether the intended application is permitted in accordance with the respective printing. Pay attention to any changes to the printing when replacing components.

The product is an open system. As such, the product must only be installed in appropriate enclosures or electrical operation rooms to which the following applies:

- Can only be opened using a tool or key
- Inside pollution degree 1 or 2
- In operation, internal air temperature within the range of 0 °C ≤ Ta ≤ +55 °C or −20 °C ≤ Ta ≤ +60 °C for components with extension number …/025-xxx or −40 °C ≤ Ta ≤ +70 °C for components with extension number …/040-xxx
- Minimum degree of protection: min. IP54 (acc. to EN/IEC 60529)
- For use in Zone 2 (Gc), compliance with the applicable requirements of the standards EN/IEC/ABNT NBR IEC 60079-0, -7, -11, -15
- For use in Zone 22 (Dc), compliance with the applicable requirements of the standards EN/IEC/ABNT NBR IEC 60079-0, -7, -11, -15 and -31
- For use in mining (Mb), minimum degree of protection IP64 (acc. EN/IEC 60529) and adequate protection acc. EN/IEC/ABNT NBR IEC 60079-0 and -1
- Depending on zoning and device category, correct installation and compliance with requirements must be assessed and certified by a “Notified Body” (ExNB) if necessary!

Explosive atmosphere occurring simultaneously with assembly, installation or repair work must be ruled out. Among other things, these include the following activities

- Insertion and removal of components
- Connecting or disconnecting from fieldbus, antenna, D-Sub, ETHERNET or USB connections, DVI ports, memory cards, configuration and programming interfaces in general and service interface in particular:
    - Operating DIP switches, coding switches or potentiometers
    - Replacing fuses

Wiring (connecting or disconnecting) of non-intrinsically safe circuits is only permitted in the following cases

- The circuit is disconnected from the power supply.
- The area is known to be non-hazardous.

Outside the device, suitable measures must be taken so that the rated voltage is not exceeded by more than 40 % due to transient faults (e.g., when powering the field supply).

Product components intended for intrinsically safe applications may only be powered by 750-606 or 750-625/000-001 bus supply modules.

Only field devices whose power supply corresponds to overvoltage category I or II may be connected to these components.

12.2.2 Special Notes Regarding ANSI/ISA Ex

For ANSI/ISA Ex acc. to UL File E198726, the following additional requirements apply:

- Use in Class I, Division 2, Group A, B, C, D or non-hazardous areas only
- ETHERNET connections are used exclusively for connecting to computer networks (LANs) and may not be connected to telephone networks or telecommunication cables
- **WARNING** – The radio receiver module 750-642 may only be used to connect to external antenna 758-910!
- **WARNING** – Product components with fuses must not be fitted into circuits subject to overloads!
  These include, e.g., motor circuits.
- **WARNING** – When installing I/O module 750-538, “Control Drawing No. 750538” in the manual must be strictly observed!

Information

Additional Information
Proof of certification is available on request.
Also take note of the information given on the operating and assembly instructions.
The manual, containing these special conditions for safe use, must be readily available to the user.

13 Appendix

13.1 EtherCAT® Object Dictionary

Table 185: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object *Description Name*	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/Object *Description Name*
CoE Communication Area 0x1000-0x1FFF		0x1000	VARIABLE	*Device Type*	0	UNSIGNED32	RO	no	0x00001389 *Device Type*
		0x1001	VARIABLE	*Error Register*	0	UNSIGNED8	RO	no	*Error Register*
		0x1008	VARIABLE	*Manufacturer Device Name*	0	VISIBLE_ STRING	RO	no	750-354/000-001 EtherCAT fieldbus coupler *Manufacturer Device Name*
		0x1009	VARIABLE	*Manufacturer Hardware Version*	0	VISIBLE_ STRING	RO	no	xx/xx Manufacturer *Hardware Version*
		0x100A	VARIABLE	*Manufacturer Software Version*	0	VISIBLE_ STRING	RO	no	xx.xx.xx/xx Manufacturer *Software Version*
		0x1018	RECORD	*Identity Object*	0	UNSIGNED8	RO	no	4 *Number of Entries*
					1	UNSIGNED32	RO	no	0x00000021 *Vendor ID*
					2	UNSIGNED32	RO	no	0x07500354 *Product Code*
					3	UNSIGNED32	RO	no	0x00010002 *Revision Number*
					4	UNSIGNED32	RO	no	*Serial Number*
		0x10F3	RECORD	*Diagnosis History*	0	UNSIGNED8	RO	no	*Number of Entries*
					1	UNSIGNED8	RO	no	12 *Maximum Messages*
					2	UNSIGNED8	RO	no	Subindex of the newest message *Newest Message*
					3	UNSIGNED8	RW	no	Subindex of the last acknowledged message *Newest Acknowledged Message*
					4	BOOLEAN	RO	TX	*New Message Available*
					5	UNSIGNED8	RW	no	*Flags:* Bit 0 = enable EMCY sending Bit 1 = ignore infos Bit 2 = ignore warnings Bit 3 = ignore errors (can't be set to 0)
							RO	no	Bit 4: 1 = acknowedge mode Bit 5: 1 = message buffer is full and a new message will be discarded *Control Flags*
					6..17	OCTETSTRING	RO	no	See 750-354(/000-001) error history specifictation

Table 185: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object *Description Name*	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/Object *Description Name*
CoE Communication Area 0x1000-0x1FFF	Receive PDO Mapping 0x1600-0x17FF	0x1600.. 0x163F	RECORD	*RxPDO Mapping Terminal x*	0	UNSIGNED8	RO	no	*Number of Entries*
					1..X-1	UNSIGNED32	RO	no	Mapping 0xAAAABBCC A: Index B: Sub-index C: Length in Bit *Output Mapping Area x*
		0x16FF	RECORD	*Device Control PDO*	0	UNSIGNED8	RO	no	*Number of Entries*
					1	UNSIGNED32	RO	no	0xF2000101 *FC Control, K-Bus Cycle Overrun Flag Disable*
					2	UNSIGNED32	RO	no	0xF2000201 *FC Control, Input Process Data Hold Request*
					3	UNSIGNED32	RO	no	0xF2000301 *FC Control, Output Process Data Hold Request*
					4	UNSIGNED32	RO	no	0xF2000401 *FC Control, Output Process Data Clear Request*
					5	UNSIGNED32	RO	no	0x0000000C Gap
					6	UNSIGNED32	RO	no	0xF2000510 *Diagnostics Control Word*
		0x1701	RECORD	*RxPDO Gap after Digital Modules*	0	UNSIGNED8	RO	no	*Number of Entries*
					1	UNSIGNED32	RO	no	Mapping 0xAAAABBCC A: Index B: Sub-index C: Length in Bit *RxPDO Gap after Digital Modules*

Table 185: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object *Description Name*	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/Object *Description Name*
CoE Communication Area 0x1000-0x1FFF	Transmit PDO Mapping 0x1A00-0x1BFF	0x1A00 .. 0x1A3F	RECORD	*TxPDO Mapping Terminal x*	0	UNSIGNED8	RO	no	*Number of Entries*
					1..X-1	UNSIGNED32	RO	no	Mapping 0xAAAABBCC A: Index B: Sub-index C: Length in Bit *Input Mapping Area x*
		0x1AFF	RECORD	*Device Status PDO*	0	UNSIGNED8	RO	no	*Number of entries*
					1	UNSIGNED32	RO	no	0xF1000101 *FC Status, K-Bus Cycle Overrun Flag*
					2	UNSIGNED32	RO	no	0xF1000201 *FC Status, Input Process Data Hold Ack.*
					3	UNSIGNED32	RO	no	0xF1000301 *FC Status, Ouput Process Data Hold Ack.*
					4	UNSIGNED32	RO	no	0xF1000401 *FC Status, Output Process Data Clear Ack.*
					5	UNSIGNED32	RO	no	0x0000000B *Gap*
					6	UNSIGNED32	RO	no	0x10F30401 Diagnosis History, New Message Available
					7	UNSIGNED32	RO	no	0xF1000510 *Diagnostics Status Word*
		0x1B01	RECORD	*TxPDO Gap after Digital Modules*	0	UNSIGNED8	RO	no	*Number of Entries*
					1	UNSIGNED32	RO	no	Mapping 0xAAAABBCC A: Index B: Sub-index C: Length in Bit *TxPDO Gap after Digital Modules*

Table 185: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object Description Name	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/Object Description Name
CoE Communication Area 0x1000-0x1FFF		0x1C00	ARRAY	*Sync Manager Communication Type*	0	UNSIGNED8	RO	no	4 *Number of Entries*
					1	UNSIGNED8	RO	no	Mbx Receive 1
					2	UNSIGNED8	RO	no	Mbx Send 2
					3	UNSIGNED8	RO	no	Process Data Output 3
					4	UNSIGNED8	RO	no	Process Data Input 4
		0x1C12	ARRAY	*Sync Manager Channel 2 RxPDO Assign*	0	UNSIGNED8	RO	no	*Number of Entries*
					1..67	UNSIGNED16	RO	no	Indices of RxPDOs
		0x1C13	ARRAY	*Sync Manager Channel 3 TxPDO Assign*	0	UNSIGNED8	RO	no	*Number of Entries*
					1..67	UNSIGNED16	RO	no	Indices of TxPDOs
		0x1C32	RECORD	*Sync Manager 2 Synchronization*	0	UNSIGNED8	RO	no	11 *Number of Entries*
					1	UNSIGNED16	RO	no	0x01 (Synchronous with SM Event) *Synchronization Type*
					4	UNSIGNED16	RO	no	0x02 (Synchronous Supported) *Synchronization Types Supported*
					5	UNSIGNED32	RO	no	*Minimum Cycle Time*
					11	UNSIGNED32	RO	no	*Cycle Time too Small*
		0x1C33	RECORD	*Sync Manager 3 Synchronization*	0	UNSIGNED8	RO	no	11 *Number of Entries*
					1	UNSIGNED16	RO	no	0x22 (Synchronous with SM2 Event) *Synchronization Type*
					4	UNSIGNED16	RO	no	0x02 (Synchronous Supported) *Synchronization Types Supported*
					5	UNSIGNED32	RO	no	*Minimum Cycle Time*
					11	UNSIGNED32	RO	no	*Cycle Time too Small*

Table 185: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object *Description Name*	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/Object *Description Name*
Manufacturer Specific Area 0x2000-0x5FFF		0x2000	ARRAY	*Register Table Fill Command Object*	0 1 2 3	UNSIGNED8 OCTECT-STRING UNSIGNED8 OCTECT-STRING	ROx RW RO RO	no no no no	3 *Number of Entries* *Write Terminal Number into This to Fill the Table* *Status* *Reply*
		0x2001	VARIABLE	*Terminal Number*	0	UNSIGNED8	RO	no	*Terminal Number*
		0x2002	ARRAY	*Terminal Channel 1*	0 1..64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2003	ARRAY	*Terminal Channel 2*	0 1..64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2004	ARRAY	*Terminal Channel 3*	0 1..64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2005	ARRAY	*Terminal Channel 4*	0 1..64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2006	ARRAY	*Terminal Channel 5*	0 1...64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2007	ARRAY	*Terminal Channel 6*	0 1...64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2008	ARRAY	*Terminal Channel 7*	0 1...64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2009	ARRAY	*Terminal Channel 8*	0 1...64	UNSIGNED8 UNSIGNED16	RO RO	no no	*Number of Entries* Content of Register x
		0x2010	RECORD	*Single Register Access Data*	0 1 2 3 4	UNSIGNED8 UNSIGNED8 UNSIGNED8 UNSIGNED8 UNSIGNED16	RO RW RW RW RW	no no no no no	4 *Number of Entries* *Terminal Number (1..64), Coupler (0)* *Table Number (0..x)* *Register Number (0..x)* *Data*
		0x2011	RECORD	*Single Register Read/Write Command Object*	0 1 2 3	UNSIGNED8 OCTECT-STRING UNSIGNED8 OCTECT-STRING	RO RW RO RO	no no no no	3 *Number of Entries* *R (0x52) = Read, W (0x57) = Write* *Status* *Reply*
		0x2100	RECORD	*PDO Index Assignment*	0 1 2	UNSIGNED8 BOOLEAN BOOLEAN	RO RW RO	no no no	2 *Number of Entries* *Use Linear Assignment after next Coldstart* *Current Assignment (TRUE = Linear, FALSE = Conforming to Modular Device Profile)*
		0x2134	VARIABLE	Explicite Device ID (Upper 8 bits)	0	UNSIGNED8	RW	no	Explicite Device ID (Upper 8 bits)

Table 186: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object *Describtion Name*	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/*Object Description Name*
Profile Area 0x6000-0xFFFF	Input Area 0x6000-0x6FFF	0x6000.. 0x63F0 (Steps of 0x0010)	RECORD	Input Data (Examples for object name: - *750-1234* - *750-4xx*)	0 1..X-1	UNSIGNED8 depends on IOM	RO RO	no **TX**	*Number of Entries* Examples of Entry Names: *Status* *Channel x Status* *Channel x Data* *Byte x* *Word x* *DWord x* *Channel x, Byte y* *Channel x, Word y* *Channel x, DWord y*
	Output Area 0x7000-0x7FFF	0x7000.. 0x73F0 (Steps of 0x0010)	RECORD	Output Data (Examples for object name: - *750-1234* - *750-5xx*)	0 1..X-1	UNSIGNED8 depends on IOM	RO RO	no **RX**	Number of Entries Examples of Entry Names: *Control* *Channel x Control* *Channel x Data* *Byte x* *Word x* *DWord x* *Channel x, Byte y* *Channel x, Word y* *Channel x. DWord y*
	Configuration Area 0x8000-0x8FFF	X							
	Information Area 0x9000-0x9FFF	0x9000.. 0x93F0 (Steps of 0x0010)	RECORD	*Module Identi-fication IOM x*	0 9 10	UNSIGNED8 UNSIGNED16 UNSIGNED32	RO RO RO	no no no	10 *Number of Entries* Non Digital: 1 Digital: 2 *Module PDO Group* Terminal Description *Module Ident*
	Diagnosis/Service/Reserved 0xA000-0xEFFF	X							

Table 186: EtherCAT® Object Dictionary

		Index	Object Type	Name/ Object Describtion Name	Sub Index	Data Type	Access	PDO Mapping	Description/ Value/Object Description Name
Profile Area 0x6000-0xFFFF	Device Area 0xF000-0xFFFF	0xF000	RECORD	*Modular Device Profile*	0	UNSIGNED8	RO	no	5 *Number of Entries*
					1	UNSIGNED16	RO	no	0x0010 *Index Distance*
					2	UNSIGNED16	RO	no	64 Maximum Number of Modules
					3	UNSIGNED32	RO	no	0x00000000 *Standard Entries in Object 0x8yy0*
					4	UNSIGNED32	RO	no	0x00000300 *Standard Entries in Object 0x9yy0*
					5	UNSIGNED16	RO	no	Module PDO Group of Device 0x0000 *Module PDO Group of Device*
		0xF00E	ARRAY	*PDO Group Alignment PDO Numbers*	0	UNSIGNED8	RO	no	3 *Number of Entries*
					1	UNSIGNED16	RO	no	PDO Used for Alignment after PDO Group 0 0 (None)
					2	UNSIGNED16	RO	no	PDO Used for Alignment after PDO Group 1 0 (None)
					3	UNSIGNED16	RO	no	PDO Used for Alignment after PDO Group 2 0x101 (PDO 258)
		0xF00F	ARRAY	*Module PDO Group Alignment*	0	UNSIGNED8	RO	no	3 *Number of Entries*
					1	UNSIGNED16	RO	no	Alignment Needed after PDO Group 0 0
					2	UNSIGNED16	RO	no	Alignment Needed after PDO Group 1 0
					3	UNSIGNED16	RO	no	Alignment Needed after PDO Group 2 2
		0xF030	ARRAY	*Configured Module List*	0	UNSIGNED8	RW	no	*Number of Entries*
					1..64	UNSIGNED32	RW	no	Module Ident (Same Value as 0x9kk0:0A)
		0xF040	ARRAY	*Detected Address List*	0	UNSIGNED8	RO	no	*Number of Entries*
					1..64	UNSIGNED16	RO	no	Address of Module (Slot Number)
		0xF050	ARRAY	*Detected Module Ident List*	0	UNSIGNED8	RO	no	*Number of Entries*
					1..64	UNSIGNED32	RO	no	Module Ident (Same Value as 0x9kk0:0A)
		0xF100	RECORD	*Device Status*	1	BOOLEAN	RO	**TX**	*K-Bus Cycle Overrun Flag*
					2	BOOLEAN	RO	**TX**	*Input Process Data Hold Ack.*
					3	BOOLEAN	RO	**TX**	*Output Process Data Hold Ack.*
					4	BOOLEAN	RO	**TX**	*Output Process Data Clear Ack.*
					5	UNSIGNED16	RO	**TX**	*Diagnostics Status Word*

Table 186: EtherCAT® Object Dictionary

		Index	Object Type	*Name/ Object Describtion Name*	Sub Index	Data Type	Access	PDO Mapping	*Description/ Value/Object Description Name*
	Device Area 0xF000-0xFFFF	0xF200	RECORD	*Device Control*	0	UNSIGNED8	RO	no	*Number of Entries*
					1	BOOLEAN	RO	**RX**	*K-Bus Cycle Overrun Flag Disable*
					2	BOOLEAN	RO	**RX**	*Input Process Data Hold Request*
					3	BOOLEAN	RO	**RX**	*Output Process Data Hold Request*
					4	BOOLEAN	RO	**RX**	*Output Process Data Clear Request*
					5	UNSIGNED16	RO	**RX**	*Diagnostics Control Word*

13.2 AL Status Codes

Table 187: AL Status Codes

AL Status Code	Meaning
0x0000	No error
0x0001	General error
0x0011	Invalid status requested (e.g. OP while slave is in PREOP)
0x0012	Undefined status requested
0x0013	BOOT status is not supported
0x0015	Invalid mailbox configuration (BOOT)
0x0016	Invalid mailbox configuration (PREOP)
0x0017	Invalid Sync Manager configuration (SM2/SM3)
0x001B	Sync Manager Watchdog Timeout
0x001D	Setting of Sync Manager 2 (output data) invalid
0x001E	Setting of Sync Manager 3 (input data) invalid
0x001F	Invalid Watchdog settings
0x002B	No valid inputs or outputs
0x002D	No Sync error
0x0030	Invalid DC Sync settings
0x0032	PLL error
0x8000-0x803F	Cannot set process data mapping for I/O module Module number = AL Status Code – 0x8000 + 1 → Contact technical support so that a firmware update can be implemented.
0x8040	Internal error. Too little memory reserved for process data mapping. → Contact technical support so that a firmware update can be implemented.
0x8041	Out of Memory. → Contact technical support so that a firmware update can be implemented.
0x9000	K bus error during first k bus boot. → Ensure that all I/O modules are correctly connected.
0x9001	K bus error in IDLE → Ensure that all I/O modules are correctly connected.
0x9002	K bus error in SM2 triggered operation. → Ensure that all I/O modules are correctly connected.
0x9003	The I.O modules were reconnected during operation and the module configuration no longer corresponds with the original configuration. → Reconnect the I/O modules only when the power supply is switched off! Turn the power supply off and on.

Table 187: AL Status Codes

AL Status Code	Meaning
0x9004	Internal error. → Contact technical support so that a firmware update can be implemented.

13.3 SDO Abort Codes

Table 188: SDO Abort Codes

SDO Abort Code	Meaning
0x05030000	Toggle bit did not toggle
0x05040001	Invalid or unknown command
0x05040005	Not enough memory
0x06010000	Not supported access to an object. (e.g. complete access to a subindex greater than 1)
0x06010001	Attempt to read a write-only object
0x06010002	Attempt to write a read-only object
0x06010004	SDO Complete access not supported for objects of variable length such as ENUM object types
0x06020000	Object does not exist
0x06070010	Length of transmitted data is incompatible with object or subindex
0x06090011	Subindex does not exist
0x06090030	Value range of parameter exceeded (only for write access)
0x06090031	Value to be written is too large.
0x06090032	Value of parameter written too low
0x08000000	General error
0x08000020	The data to be read/written can not be retrieved by the application and/or transmitted to the application. (e.g. writing a command object while this is still engaged in the implementation of the last order)

13.4 EMCY Codes

Table 189: EMCY Codes

error Code	error Reg.	data[0]	data[1]	data[2]	data[3]	data[4]	error description
0xFF00 (device specific)	0x81	0x01	LED error code	LED error argument	0	0	An error condition has been detected by the application software. The cause of the error as described by the LED error code and the LED error argument can be found in the blink code table of the users manual.
0x0000 (error reset)	0x81	0x01	LED error code	LED error argument	0	0	The cause for the error condition as described by the LED error code and the LED error argument is no longer existent.
0xFF00 (device specific)	0x81	0x02	0	0	0	0	Diagnosis History object has overflown or is full now.
0xA000	0x11	0x08	SM2 min. length (low byte	SM2 min. length (high byte)	SM2 max. length (low byte)	SM2 max. length (high byte)	Length set at SM2 by the master is invalid. The EMCY contains the allowed minimum and maximum length for SM2.
0xA000	0x11	0x09	SM2 min. Address (low byte	SM2 min. Address (high byte)	SM2 max. address (low byte)	SM2 max. address (high byte)	Start address set at SM2 by the master is invalid. The EMCY contains the allowed minimum and maximum start address for SM2.
0xA000	0x11	0x0A	SM2 expected settings				Settings configured at SM2 by the master are invalid. The EMCY contains the expected settings. Please refer to SM_Error_EMCY_Data_Structure.xls for details.
0xA000	0x11	0x0C	SM3 min. length (low byte	SM3 min. length (high byte)	SM3 max. length (low byte	SM3 max. length (high byte)	Length set at SM3 by the master is invalid. The EMCY contains the allowed minimum and maximum length for SM3.
0xA000	0x11	0x0D	SM3 min. Address (low byte	SM3 min. Address (high byte)	SM3 max. address (low byte	SM3 max. address (high byte)	Start address set at SM3 by the master is invalid. The EMCY contains the allowed minimum and maximum start address for SM3.
0xA000	0x11	0x0E	SM3 expected settings				Settings configured at SM3 by the master are invalid. The EMCY contains the expected settings. Please refer to SM_Error_EMCY_Data_Structure.xls for details.

List of Figures

Figure 1: Fieldbus Node (Example) 18
Figure 2: Marking Area for Serial Numbers 19
Figure 3: Update Matrix up to 2015 20
Figure 4: Update Matrix from 2016 20
Figure 5: Isolation (Example) 23
Figure 6: System Supply 24
Figure 7: System Voltage (Example) 25
Figure 8: Field Supply (Sensor/Actuator) 29
Figure 9: Supply Module with Fuse Carrier (Example 750-610) 31
Figure 10: Removing the Fuse Carrier 31
Figure 11: Opening the Fuse Carrier 32
Figure 12: Changing the Fuse 32
Figure 13: Fuse Modules for Automotive Fuses, Series 282 32
Figure 14: Fuse Modules for Automotive Fuses, Series 2006 33
Figure 15: Fuse Modules with Pivotable Fuse Carrier, Series 281 33
Figure 16: Fuse Modules with Pivotable Fuse Carrier, Series 2002 33
Figure 17: Supply Example 34
Figure 18: Carrier Rail Contact (Example) 38
Figure 19: Cable Shield at Ground Potential 39
Figure 20: Examples of the WAGO Shield Connecting System 40
Figure 21: Application of the WAGO Shield Connecting System 40
Figure 22: View EtherCAT® fieldbus coupler 42
Figure 23: Device Supply 44
Figure 24: RJ-45 Connector 45
Figure 25: Display Elements 46
Figure 26: Service Interface (Closed and Opened Flap) 47
Figure 27: ID Selection Switch (for Example Setting “7“) 48
Figure 28: Spacing 57
Figure 29: Release Tab 60
Figure 30: Insert I/O Module (Example) 61
Figure 31: Snap the I/O Module into Place (Example) 61
Figure 32: Removing the I/O Module (Example) 62
Figure 33: Data Contacts 63
Figure 34: Example for the Arrangement of Power Contacts 64
Figure 35: Connecting a Conductor to a CAGE CLAMP® 65
Figure 36: Start Behavior 68
Figure 37: Mapping the Process Data 70
Figure 38: Example of the Principle Mapping of Process Data Using EtherCAT® 71
Figure 39: Number of Free Slots 100
Figure 40: Acknowledgement of a Diagnostic Message 100
Figure 41: Overrun Counter in Subindex 11 102
Figure 42: Memory Areas and Data Exchange of the Fieldbus Coupler 109
Figure 43: Display Elements 121
Figure 44: Node Status – I/O LED Signaling 124
Figure 45: Error Message Coding 124
Figure 46: Behavior During a Local Bus Error 134

Figure 47: Error Behavior of the Process Data Watchdog....138
Figure 48: Status Diagram of an EtherCAT® Slave 144
Figure 49: Marking Example According to ATEX and IECEx 190
Figure 50: Text Detail – Marking Example According to ATEX and IECEx 190
Figure 51: Marking Example for Approved Ex i I/O Module According to ATEX and IECEx....192
Figure 52: Text Detail – Marking Example for Approved Ex i I/O Module According to ATEX and IECEx....192
Figure 53: Marking Example According to NEC 194
Figure 54: Text Detail – Marking Example According to NEC 500 194
Figure 55: Text Detail – Marking Example for Approved Ex i I/O Module According to NEC 505....195
Figure 56: Text Detail – Marking Example for Approved Ex i I/O Module According to NEC 506....195
Figure 57: Text Detail – Marking Example for Approved Ex i I/O Modules According to CEC 18 attachment J 196

List of Tables

Table 1: Number Notation 11
Table 2: Font Conventions 11
Table 3: Legend for Figure “Update Matrix from 2016” 20
Table 4: Legend for Figure “System Supply” 24
Table 5: Alignment 26
Table 6: Legend for Figure “Field Supply (Sensor/Actuator) for ECO Fieldbus Coupler” 29
Table 7: Power Supply Modules 30
Table 8: Legend for Figure “Supply Example for Fieldbus Coupler/Controller” 35
Table 9: WAGO Ground Wire Terminals 37
Table 10: Legend for Figure “View EtherCAT® Fieldbus Coupler” 43
Table 11: RJ-45 Connector and RJ-45 Connector Configuration 45
Table 12: Display Elements Fieldbus Status 46
Table 13: Display Elements Node Status 46
Table 14: Legend for Figure “Service Interface (Closed and Opened Flap)” 47
Table 15: EDI Address Structure (“Explicit Device ID”) 48
Table 16: Technical Data – Device Data 49
Table 17: Technical Data – Safe Electrical Isolation 49
Table 18: Technical Data – Supply 49
Table 19: Technical Data – Communication 50
Table 20: Technical Data – Accessories 50
Table 21: Technical Data – Field Wiring 50
Table 22: Technical Data – Power Jumper Contacts 50
Table 23: Technical Data – Data Contacts 50
Table 24: Technical Data – Climatic Environmental Conditions 51
Table 25: Technical Data – Mechanical Strength 51
Table 26: WAGO DIN Rails 57
Table 27: Generic Structure of Object 0x6000 72
Table 28: Example Structure for Object 0x6000 for 750-433 – 4 DI 73
Table 29: Example Structure for Object 0x6000 for 750-467 – 2 AI 73
Table 30: Generic Structure of Object 0x7000 74
Table 31: Example Structure for Object 0x7000 for 750-504 – 4 DO 74
Table 32: Example Structure for Object 0x7000 for 750-550 – 2 AO 74
Table 33: Structure for Object 0xF100 75
Table 34: Structure of Object 0xF200 75
Table 35: Generic Structure of Rx/TxPDO (0x16xx/0x1Axx) 76
Table 36: Recommendations for Setting the 0x2100:01 77
Table 37: Example Output Process Image 79
Table 38: Example Input Process Image 79
Table 39: Object 0xF100 (Fieldbus Coupler Status Information) 80
Table 40:Object 0xF200 (Fieldbus Coupler Control Information) 80
Table 41: Object 0x16FF (Fieldbus Coupler Control PDO) 80
Table 42: Object 0x1AFF (Fieldbus Coupler Status PDO) 81
Table 43: Object 0x6000 (Input Data 750-670) 81
Table 44: Object 0x7000 (Output Data 750-670) 82
Table 45: Object 0x1600 (RxPDO 750-670) 82
Table 46: Object 0x1A00 (TxPDO 750-670) 83

Table 47: Object 0x6010 (Input Data 750-433) 84
Table 48: Object 0x1A01 (TxPDO 750-433)....84
Table 49: Object 0x7020 (Output Data 750-504)....84
Table 50: Object 0x1602 (RxPDO 750-504)....84
Table 51: Object 0x7030 (Output Data 750-550)....85
Table 52: Object 0x1603 (RxPDO 750-550)....85
Table 53: Object 0x6040 (Input Data 750-467) 85
Table 54: Object 0x1A04 (TxPDO 750-467)....85
Table 55: Object 0x1701 (RxPDO Gap at the End of the Output Process Image, Behind the DO´s)....86
Table 56: Object 0x1B01 ((TxPDO Gap at the End of the Input Process Image, Behind the DI´s)....86
Table 57: Object 0x1C12 (RxPDO Assignment List for Sync Manager 2)....86
Table 58: Object 0x1C13 (TxPDO Assignment List for Sync Manager 3) 86
Table 59: Example Output Process Image 88
Table 60: Example Input Process Image 88
Table 61: Object 0xF100 (Fieldbus Coupler Status Information)....89
Table 62:Object 0xF200 (Fieldbus Coupler Control Information) 89
Table 63: Object 0x1600 (Fieldbus Coupler Control PDO)....89
Table 64: Object 0x1A00 (Fieldbus Coupler Status PDO)....90
Table 65: Object 0x6000 (Input Data 750-670) 90
Table 66: Object 0x7000 (Output Data 750-670)....91
Table 67: Object 0x1601 (RxPDO 750-670)....91
Table 68: Object 0x1A01 (TxPDO 750-670)....92
Table 69: Object 0x6010 (Input Data 750-433) 92
Table 70: Object 0x1A03 (TxPDO 750-433)....92
Table 71: Object 0x7020 (Output Data 750-504)....93
Table 72: Object 0x1603 (RxPDO 750-504)....93
Table 73: Object 0x7030 (Output Data 750-550)....93
Table 74: Object 0x1602 (RxPDO 750-550)....93
Table 75: Object 0x6040 (Input Data 750-467) 94
Table 76: Object 0x1A04 (TxPDO 750-467)....94
Table 77: Object 0x1701 (RxPDO Gap at the End of the Output Process Image) 94
Table 78: Object 0x1B01 (TxPDO Gap at the End of the Input Process Image) 94
Table 79: Object 0x1C12 (RxPDO Assignment List for Sync Manager 2)....94
Table 80: Object 0x1C13 (TxPDO Assignment List for Sync Manager 3) 95
Table 81: Structure of “Module Ident” for Digital I/O Modules....96
Table 82: Structure of “Module Ident” for Complex I/O Modules 97
Table 83: Structure of a Diagnostic Entry in Object 0x10F3 (Diagnosis History) 98
Table 84: Structure of the Diagnosis History Object 0x10F3. 99
Table 85: Structure of Sync Manager 2 Synchronization Object 0x1C32....103
Table 86: Structure of the Sync Manager 3 Synchronization Object 0x1C33....103
Table 87: Register List Objects 0x2000 … 0x2009 104
Table 88: Single Register Access Objects 0x2010, 0x2011 106
Table 89: PDO Index Assignment Object 0x2100 107
Table 90: Explicit Device ID (Upper 8 Bits) Object 0x2134 108
Table 91: Process Data Architecture -- Example....111

Table 92: Process Data Objects ... 112
Table 93: Fieldbus Coupler Control Word ... 113
Table 94: Fieldbus Coupler Status Word ... 114
Table 95: LED Assignment for Diagnostics ... 121
Table 96: LED L/A IN, L/A OUT ... 122
Table 97: LED RUN ... 122
Table 98: LED ERR ... 122
Table 99: Node Status Diagnostics – Solution in Event of Error ... 123
Table 100: Blink Code Table for the I/O LED Signaling, Error Code 1 ... 126
Table 101: Blink Code Table for the I/O LED Signaling, Error Code 2 ... 127
Table 102: Blink Code Table for the I/O LED Signaling, Error Code 3 ... 127
Table 103: Blink Code Table for the I/O LED Signaling, Error Code 4 ... 128
Table 104: Blink Code Table for the I/O LED Signaling, Error Code 5 ... 129
Table 105: Blink Code Table for the I/O LED Signaling, Error Code 6 ... 129
Table 106: Diagnostic Control Word ... 131
Table 107: Diagnostic Status Word ... 132
Table 108: Settings for the Sync Managers SM0 and SM1 ... 136
Table 109: Settings for the Sync Managers SM2 and SM3 ... 137
Table 110: Sync Manager Length Error EMCY Data ... 137
Table 111: Sync Manager Address Error EMCY Data ... 137
Table 112: Sync Manager Settings Error EMCY Data ... 137
Table 113: EtherCAT® State Machine (ESM) ... 146
Table 114: CoE Components ... 150
Table 115: Command Object Structure ... 152
Table 116: 1 Channel Digital Input Module with Diagnostics ... 155
Table 117: 2 Channel Digital Input Modules ... 155
Table 118: 2 Channel Digital Input Module with Diagnostics ... 155
Table 119: Diagnostic data of 2 Channel Digital Input Module with Diagnostics in the Diagnostic Status Word ... 156
Table 120: 2 Channel Digital Input Module with Diagnostics and Output Process Data ... 156
Table 121: Diagnostic data 2 Channel Digital Input Module with Diagnostics in the Diagnostic Status Word and Output Process Data ... 156
Table 122: 4 Channel Digital Input Modules ... 157
Table 123: 8 Channel Digital Input Modules ... 157
Table 124: 16 Channel Digital Input Modules ... 157
Table 125: 1 Channel Digital Output Module with Input Process Data ... 158
Table 126: 2 Channel Digital Output Modules ... 158
Table 127: 2 Channel Digital Input Modules with Diagnostics and Input Process Data ... 159
Table 128: Diagnostic data 2 Channel Digital Input Modules with Diagnostics in the Diagnostic Status Word ... 159
Table 129: 2 Channel Digital Input Modules with Diagnostics and Input Process Data 75x-506 ... 159
Table 130: Diagnostic data 2 Channel Digital Input Modules with Diagnostics in the Diagnostic Status Word ... 159
Table 131: 4 Channel Digital Output Modules ... 160
Table 132: 4 Channel Digital Output Modules with Diagnostics and Input Process Data ... 160

Table 133: 4 Channel Digital Output Modules with Diagnostics in the Diagnostic Status Word 160
Table 134: 8 Channel Digital Output Module 161
Table 135: 8 Channel Digital Output Modules with Diagnostics and Input Process Data 162
Table 136: 8 Channel Digital Output Modules with Diagnostics in the Diagnostic Status Word 162
Table 137: 16 Channel Digital Output Modules 162
Table 138: 8 Channel Digital Input/Output Modules 163
Table 139: 1 Channel Analog Input Modules 163
Table 140: 2 Channel Analog Input Modules 164
Table 141: 2-Channel Analog Input Modules HART 165
Table 142:: 2 Channel Analog Input Modules HART + 6 bytes Mailbox 165
Table 143: 4 Channel Analog Input Modules 165
Table 144: 8 Channel Analog Input Modules 166
Table 145: 3-Phase Power Measurement Module 167
Table 146: 3-Phase Power Measurement Modules 750-494, -495, (and all variations) 168
Table 147: 2 Channel Analog Output Modules 169
Table 148: 4 Channel Analog Output Modules 169
Table 149: 8 Channel Analog Output Modules 170
Table 150: Counter Modules 750-404, (and all variations except of /000-005), 753-404, -404/000-003) 171
Table 151: Counter Modules 750-404/000-005, 753-404/000-005 171
Table 152: Counter Modules 750-633 172
Table 153: Counter Modules 750-638, 753-638 172
Table 154: Pulse Width Modules 750-511, (and all variations), 753-511 173
Table 155: Serial Interface Modules with Alternative Data Format 174
Table 156: Serial Interface Modules with Standard Data Format 174
Table 157: Serial Interface Modules 750-652, 753-652 175
Table 158: Data Exchange Module 750-654, -654/000-001 175
Table 159: SSI Transmitter Interface Modules 176
Table 160: Incremental Encoder Interface Modules 750-631/000-004, -010, -011 176
Table 161: Incremental Encoder Interface Modules 750-634 177
Table 162: Incremental Encoder Interface Modules 750-637, (and all variations) 177
Table 163: Digital Pulse Interface Modules 750-635, 753-635 178
Table 164: DC-Drive Controller 750-636, -636/000-700, -636/000-800 178
Table 165: Stepper Controller 750-670, -671, -672 179
Table 166: RTC Module 750-640 180
Table 167: DALI/DSI Master Module 750-641 180
Table 168: DALI Multi-Master Module 753-647 in the “Easy” Mode 181
Table 169: DALI Multi-Master Module 753-647 in the “Full” Mode 182
Table 170: EnOcean Radio Receiver 750-642 183
Table 171: MP Bus Master Module 750-643 183
Table 172: *Bluetooth*® RF-Transceiver 750-644 184
Table 173: Vibration Velocity/Bearing Condition Monitoring VIB I/O 750-645 ... 185
Table 174: AS-interface Master Module 750-655, 753-655 186
Table 175: System Modules with Diagnostics 750-606 187

Table 176: System Modules with Diagnostics 750-610, -611 187
Table 177: Filter Modules 750-624/020-002, 750-626/020-002 187
Table 178: Binary Space Module 750-622 (with Behavior like 2 Channel Digital Input) 188
Table 179: Description of Marking Example According to ATEX and IECEx..... 191
Table 180: Description of Marking Example for Approved Ex i I/O Module According to ATEX and IECEx 193
Table 181: Description of Marking Example According to NEC 500.................. 194
Table 182: Description of Marking Example for Approved Ex i I/O Module According to NEC 505 195
Table 183: Description of Marking Example for Approved Ex i I/O Modules According to NEC 506 195
Table 184: Description of Marking Example for Approved Ex i I/O Modules According to CEC 18 attachment J 196
Table 185: EtherCAT® Object Dictionary 200
Table 186: EtherCAT® Object Dictionary 205
Table 187: AL Status Codes 208
Table 188: SDO Abort Codes 209
Table 189: EMCY Codes 210

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D - 32385 Minden
Hansastraße 27 • D - 32423 Minden
Phone: +49 571 887 – 0
Fax: +49 571 887 – 844169
E-Mail: info@wago.com
Internet: www.wago.com