WAGO-I/O-SYSTEM 750

Manual

750-337

CANopen Fieldbus Coupler, MCS
10 kBaud ... 1 MBaud; digital and analog signals

Version 4.1.0

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: http://www.wago.com

Technical Support

Phone: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 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

Table of Contents

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 "CANopen Fieldbus Coupler, MCS" (750-337) and the variants listed in the table below.

Table 1: Variants

Item Number/Variant	Designation
750-337	CANopen Fieldbus Coupler, MCS
750-337/025-000	CANopen Fieldbus Coupler, MCS/T

Note

Documentation Validity for Variants
Unless otherwise indicated, the information given in this documentation applies to listed variants.

The product "CANopen Fieldbus Coupler, MCS" (750-337) 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

This Manual, including all figures and illustrations, is copyright-protected. Any further use of this Manual by third parties that violate pertinent copyright provisions is prohibited. Reproduction, translation, electronic and phototechnical filing/archiving (e.g., photocopying) as well as any amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden, Germany. Non-observance will involve the right to assert damage claims.

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 2: 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 3: 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 that serve to increase the efficiency of technical progress. 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 WAGO-I/O-SYSTEM 750 in Compliance with Underlying Provisions

Fieldbus couplers, fieldbus 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 programmable 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/controller.

Appropriate housing (per 94/9/EG) 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.

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. WAGO Kontakttechnik GmbH & Co. KG will be exempted from any liability in case of changes in hardware or software as well as to non-compliant usage of devices.

Please send your request for modified and new hardware or software configurations directly to WAGO Kontakttechnik GmbH & Co. KG.

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.

NOTICE

Replace defective or damaged devices!
Replace defective or damaged device/module (e.g., in the event of deformed contacts), since the long-term functionality of device/module involved can no longer be ensured.

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 soiled contacts using oil-free compressed air or with ethyl alcohol and leather cloths.

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.

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.

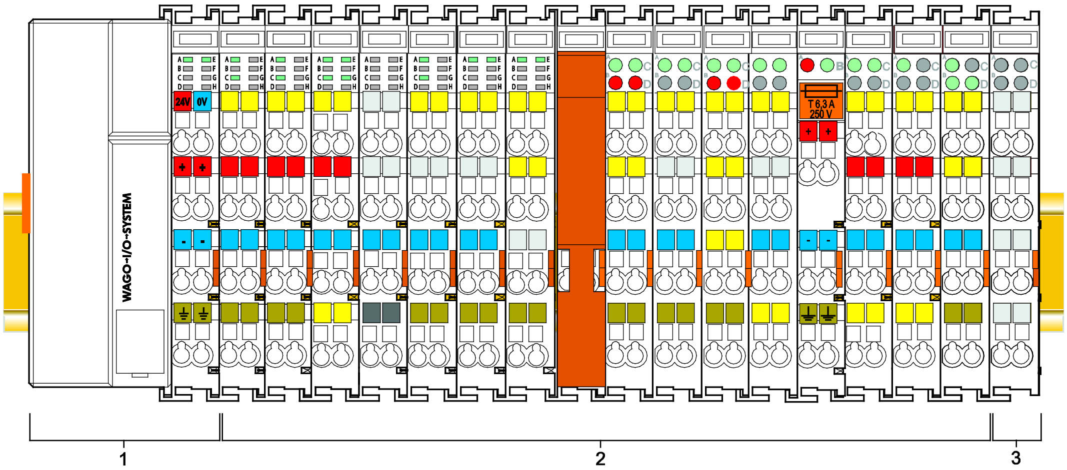


Figure 1: Fieldbus Node (Example)

Fieldbus couplers/controllers are available for different fieldbus systems.

The standard fieldbus couplers/controllers contain the fieldbus interface, electronics and a power supply terminal. The fieldbus interface forms the physical interface to the relevant fieldbus. The electronics process the data of the I/O modules and make it available for the fieldbus communication. The 24 V system supply and the 24 V field supply are fed in via the integrated power supply terminal.

The fieldbus coupler/controller exchanges process data with the respective control via the respective fieldbus. The programmable fieldbus controllers (PFC) allow implementation of additional PLC functions. WAGO-I/O-*PRO* is used to program the fieldbus controllers according to IEC 61131-3.

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 an internal 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.
In addition, the serial number is printed on the cover cap of the configuration and programming interface of the fieldbus coupler/controller, so that it can also be read when installed.

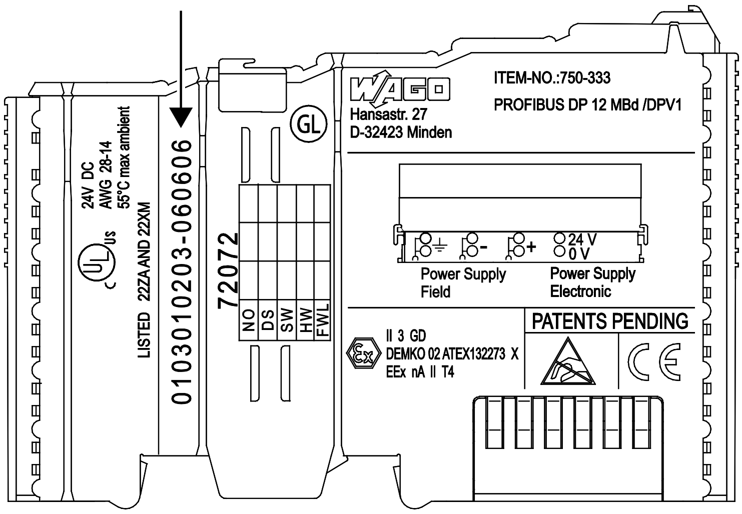


Figure 2: Labeling on the Side of a Component (Example)

Manufacturing number					
01	**03**	**01**	**02**	**03**	**-B060606**
Calendar week	Year	Software version	Hardware version	Firmware loader version	Internal number

Figure 3: Example of a Manufacturing Number

The manufacturing number consists of the production week and year, the software version (if available), the hardware version of the component, the firmware loader (if available) and further internal information for WAGO Kontakttechnik GmbH & Co. KG.

3.2 Component Update

For the case of an update of one component, the lateral marking on each component contains a prepared matrix.

This matrix makes columns available for altogether three updates to the entry of the current update data, like production order number (NO; starting from calendar week 13/2004), date stamp (DS), software version (SW), hardware version (HW) and the firmware loader version (FWL, if available).

Current version data for		1. Update	2. Update	3. Update	
Production order no.	**NO**				← only starting from calendar week 13/2004
Date stamp	**DS**				
Software version	**SW**				
Hardware version	**HW**				
Firmware loader vers.	**FWL**				← only for fieldbus couplers/controllers

If the update of a component took place, the current version data are registered into the columns of the matrix.

Additionally with the update of a fieldbus coupler or controller also the cover of the configuration and programming interface of the fieldbus coupler or controller is imprinted with the current production order number.

The original manufacturing information on the device's 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 equipping of machines
- DIN EN 50178 Equipping of high-voltage systems with electronic components (replacement for VDE 0160)
- EN 60439 Low-voltage switchgear and controlgear assemblies

3.5 Power Supply

3.5.1 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 (internal bus)
- All I/O modules have an electrical isolation between the electronics (internal bus, logic) and the field electronics. Some digital and analog input modules have each channel electrically isolated, please see catalog.

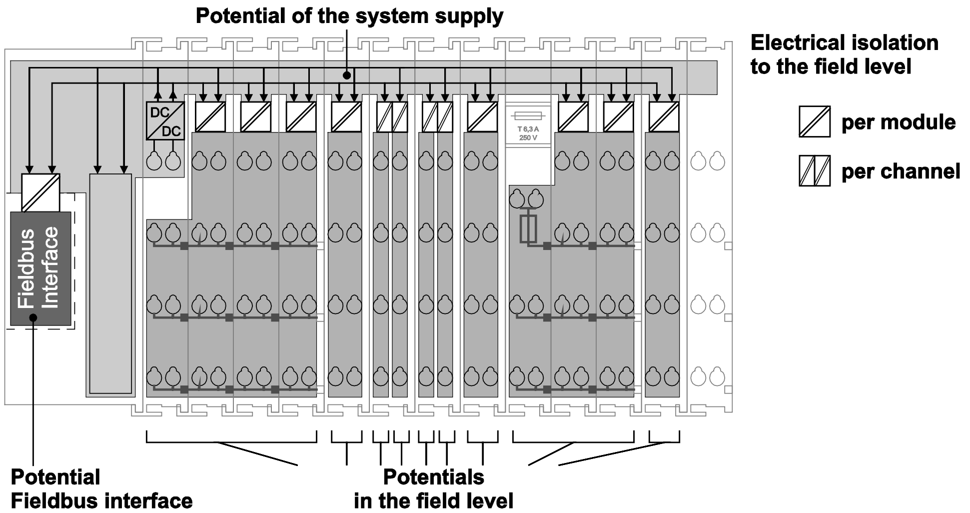


Figure 4: Isolation for Fieldbus Couplers/Controllers (Example)

3.5.2 System Supply

3.5.2.1 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.

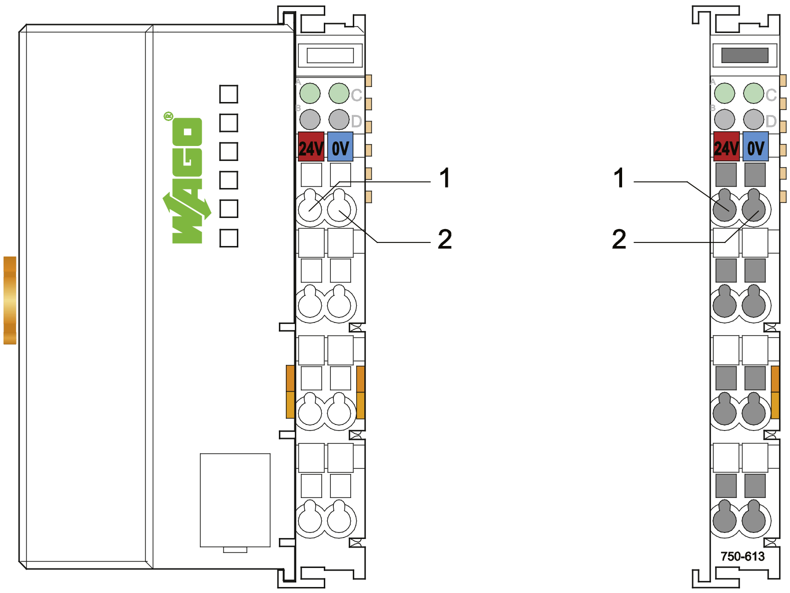


Figure 5: System Supply via Fieldbus Coupler/Controller (left) and via Internal System Supply Module (right)

Table 4: Legend for Figure “System Supply via Fieldbus Coupler/Controller (left) and via Internal System Supply Module (right)”

Position	Description
1	System supply DC 24 V (-25 % … +30 %)
2	System supply 0 V

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

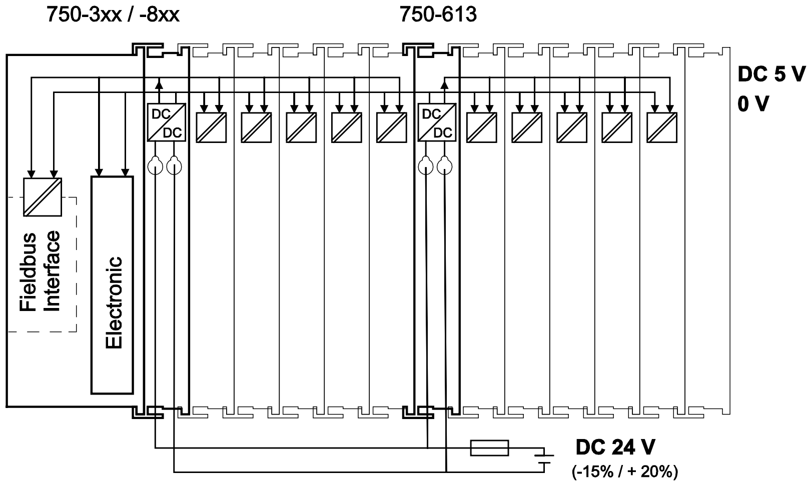


Figure 6: System Voltage for Standard Couplers/Controllers and Extended ECO Couplers

Note

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

3.5.2.2 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 the fieldbus coupler:

Internal current consumption of the coupler	350 mA at 5 V
Total current for I/O modules	1650 mA at 5 V
Sum $I_{(5\ V)\ total}$	**2000 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 a standard fieldbus coupler/controller:

A node configuration with 20 relay modules (750-517) and 30 digital input modules (750-405) should be attached to a fieldbus coupler/controller:

Internal current consumptions	20 × 90 mA = 1800 mA at 5 V
	+ 30 × 2 mA = 60 mA at 5 V
Sum of internal current consumptions	**1860 mA at 5 V**

However, the fieldbus coupler can only provide 1650 mA for the 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

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

$\eta = 0.87$
(87 % 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.

3.5.3 Field Supply

3.5.3.1 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 fieldbus coupler/controller provides field side power (DC 24 V). In this case it is a passive power supply without protection equipment.

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

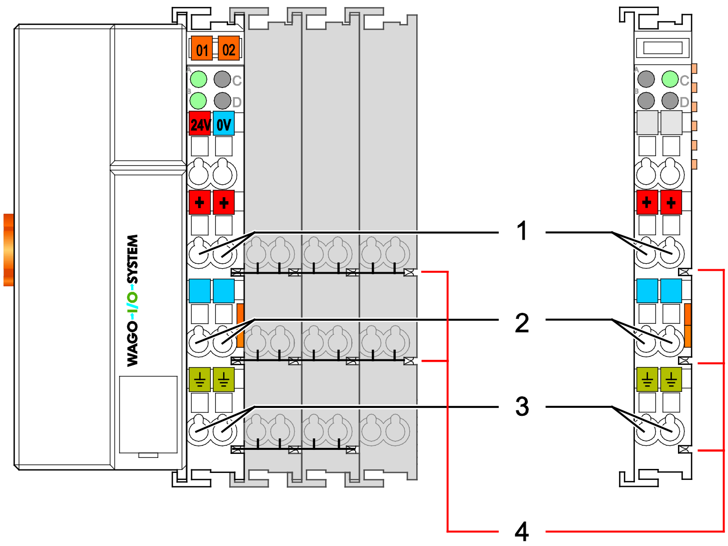


Figure 7: Field Supply for Standard Couplers/Controllers and Extended ECO Couplers

Table 6: Legend for Figure “Field Supply for Standard Couplers/Controllers and Extended ECO Couplers”

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

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 DC 24 V to AC 230 V, 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.

3.5.3.2 Fusing

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 V DC, Supply/Fuse
750-609	230 V AC, Supply/Fuse
750-615	120 V AC, Supply/Fuse
750-617	24 V AC, Supply/Fuse
750-610	24 V DC, Supply/Fuse/Diagnosis
750-611	230 V AC, Supply/Fuse/Diagnosis
750-606	Supply Module 24 V DC, 1,0 A, Ex i
750-625/000-001	Supply Module 24 V DC, 1,0 A, Ex i (without diagnostics)

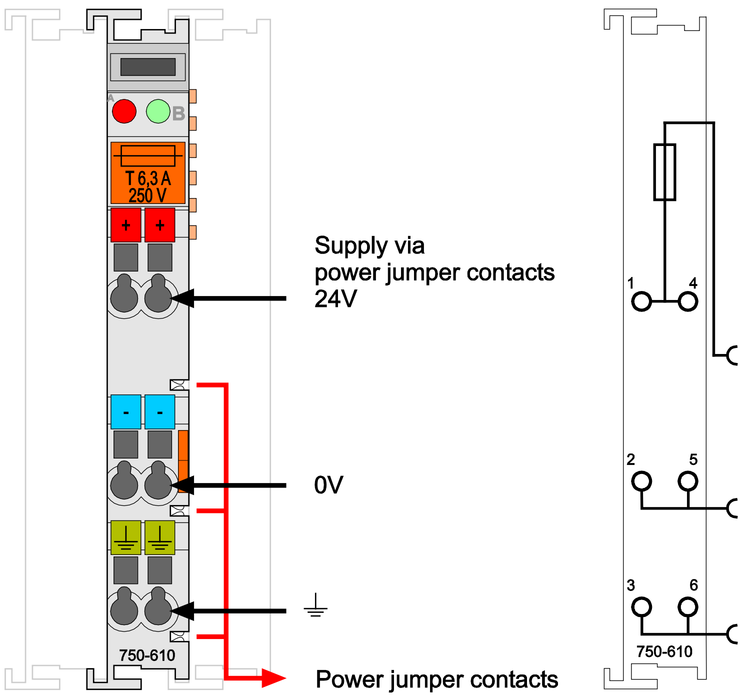


Figure 8: Supply Module with Fuse Carrier (Example 750-610)

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 9: Removing the Fuse Carrier

Lifting the cover to the side opens the fuse carrier.

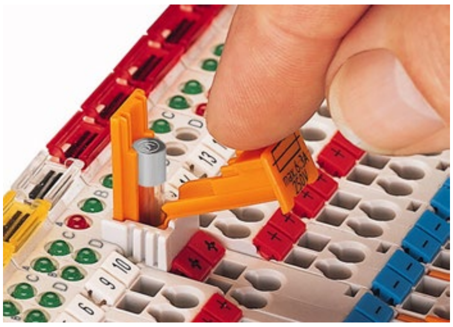


Figure 10: Opening the Fuse Carrier

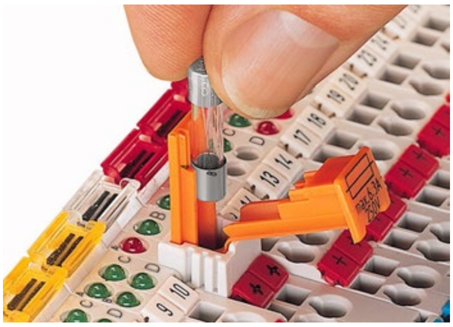


Figure 11: Changing the Fuse

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

Alternatively, fusing can be done externally. The fuse modules of the WAGO series 281 and 282 are suitable for this purpose.

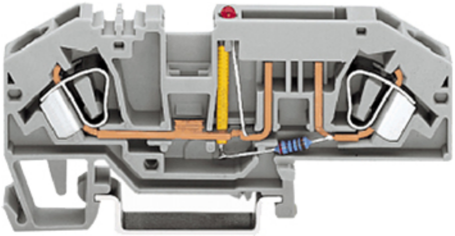


Figure 12: Fuse Modules for Automotive Fuses, Series 282

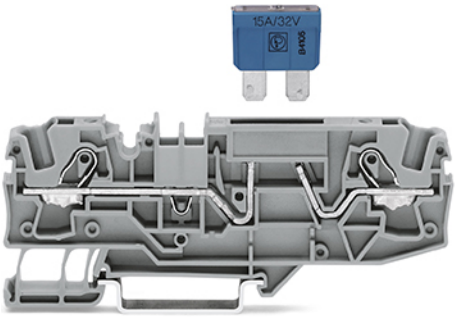


Figure 13: Fuse Modules for Automotive Fuses, Series 2006

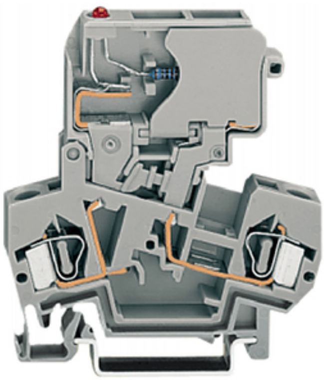


Figure 14: Fuse Modules with Pivotable Fuse Carrier, Series 281

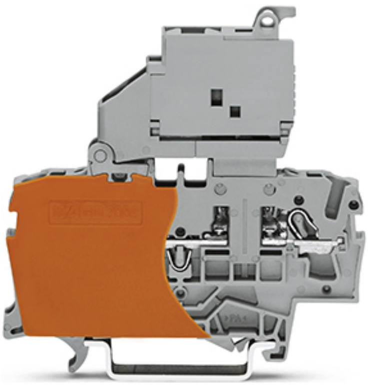


Figure 15: Fuse Modules with Pivotable Fuse Carrier, Series 2002

3.5.4 Supplementary Power Supply Regulations

The WAGO-I/O-SYSTEM 750 can also be used in shipbuilding or offshore and onshore areas of work (e. g. working platforms, loading plants). This is demonstrated by complying with the standards of influential classification companies such as Germanischer Lloyd and Lloyds Register.

Filter modules for 24 V supply are required for the certified operation of the system.

Table 8: Filter Modules for 24 V Supply

Order No.	Name	Description
750-626	Supply Filter	Filter module for system supply and field supply (24 V, 0 V), i. e. for fieldbus coupler/controller and bus power supply (750-613)
750-624	Supply Filter	Filter module for the 24 V field supply (750-602, 750-601, 750-610)

Therefore, the following power supply concept must be absolutely complied with.

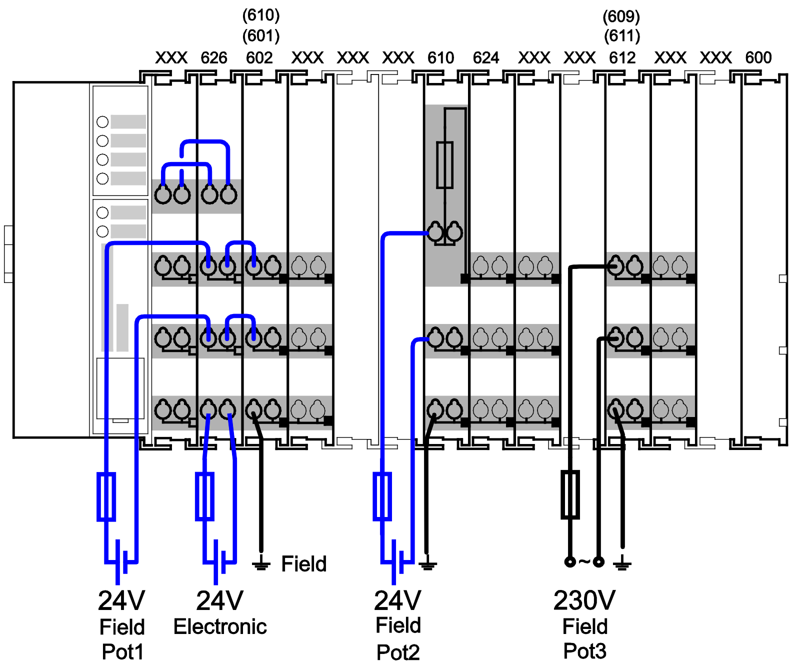


Figure 16: Power Supply Concept

Note

Use a supply module for equipotential bonding!
Use an additional 750-601/ 602/ 610 Supply Module behind the 750-626 Filter Module if you want to use the lower power jumper contact for equipotential bonding, e.g., between shielded connections and require an additional tap for this potential.

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.

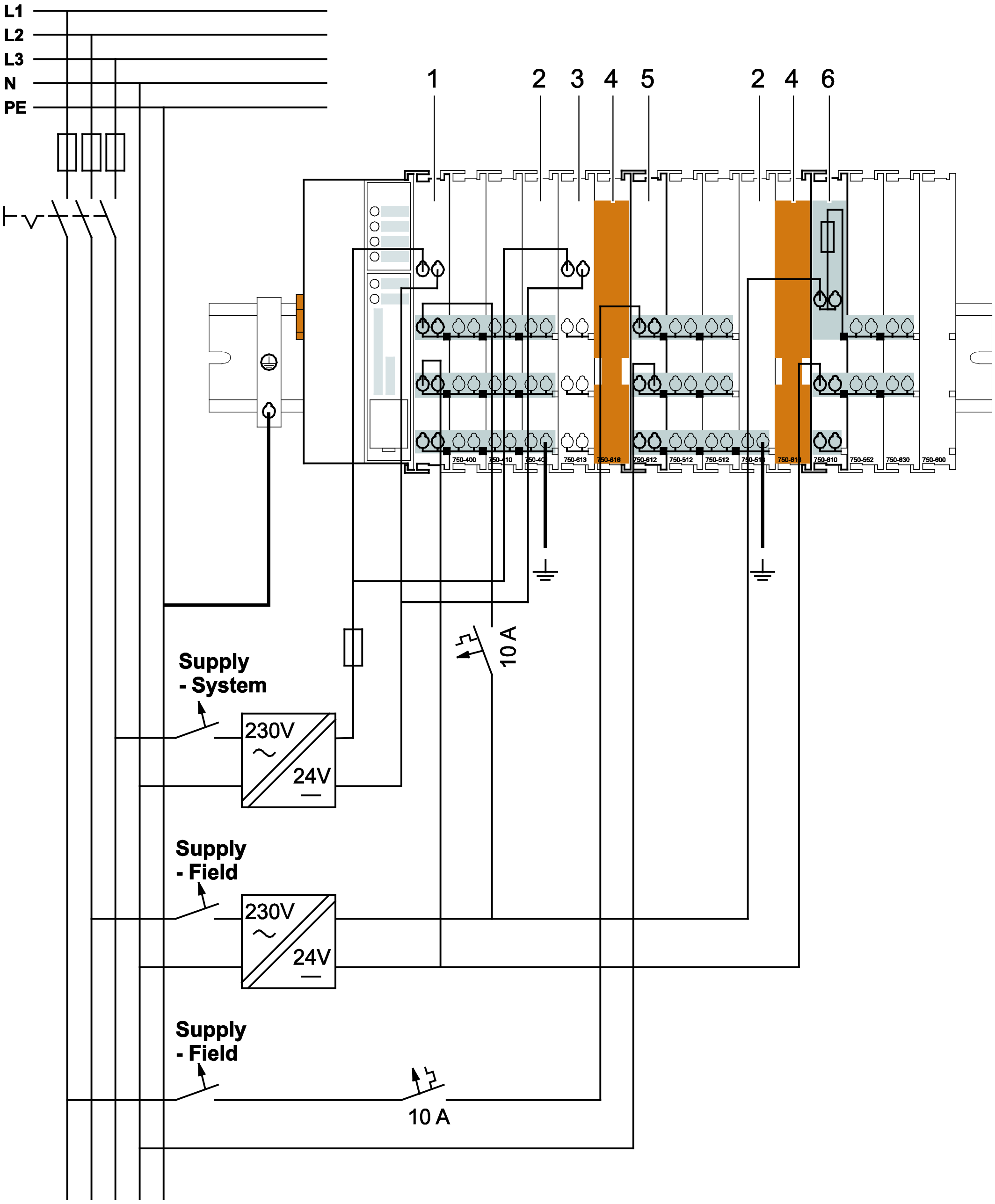


Figure 17: Supply Example for Standard Couplers/Controllers

Table 9: Legend for Figure “Supply Example for Fieldbus Coupler/Controller”

Pos.	Description
1	Power Supply on coupler 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 (see also table "WAGO power supply units").

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

Note

Power failure time not acc. IEC 61131-2!
Note that the power failure time of 10 ms acc. IEC 61131-2 is not maintained in a maximum configuration.

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.

Table 10: WAGO Power Supply Units (Selection)

WAGO Power Supply Unit	Description
787-612	Primary switched mode; DC 24 V; 2,5 A Input nominal voltage AC 230 V
787-622	Primary switched mode; DC 24 V; 5 A Input nominal voltage AC 230 V
787-632	Primary switched mode; DC 24 V; 10 A Input nominal voltage AC 230/115 V
 288-809 288-810 288-812 288-813	Rail-mounted modules with universal mounting carrier AC 115 V/DC 24 V; 0,5 A AC 230 V/DC 24 V; 0,5 A AC 230 V/DC 24 V; 2 A AC 115 V/DC 24 V; 2 A

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 11: 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.

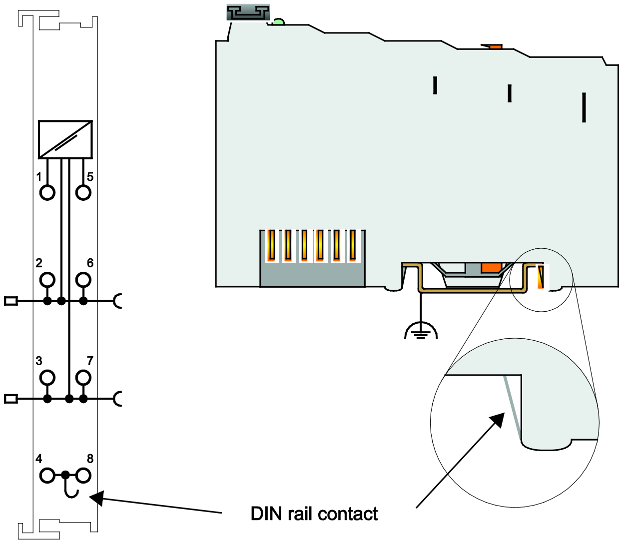


Figure 18: Carrier Rail Contact (Example)

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.

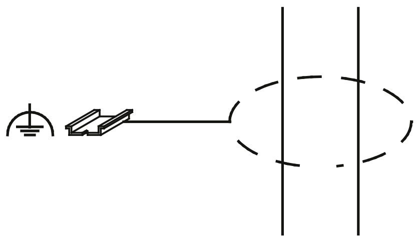


Figure 19: Cable Shield at Ground Potential

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 the bus line is described in the respective configuration guidelines and standards of the bus system.

3.7.3 Signal Lines

I/O modules for analog signals and some interface I/O modules are equipped with shield clamps.

Note

Use shielded signal lines!
Only use shielded signal lines for analog signals and I/O modules which are equipped with shield clamps. Only then can you 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.

3.7.4 WAGO Shield Connecting System

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

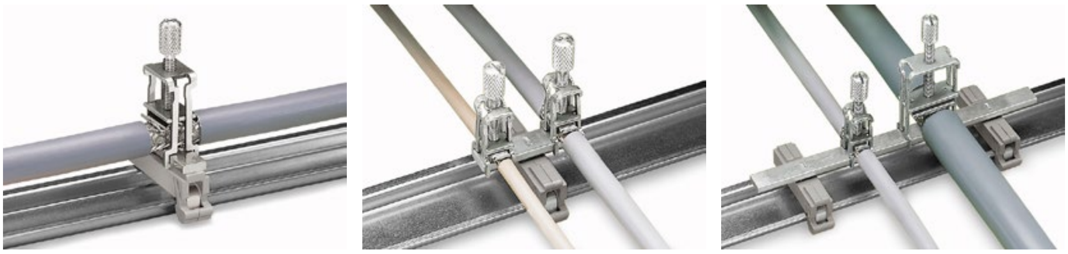


Figure 20: Examples of the WAGO Shield Connecting System

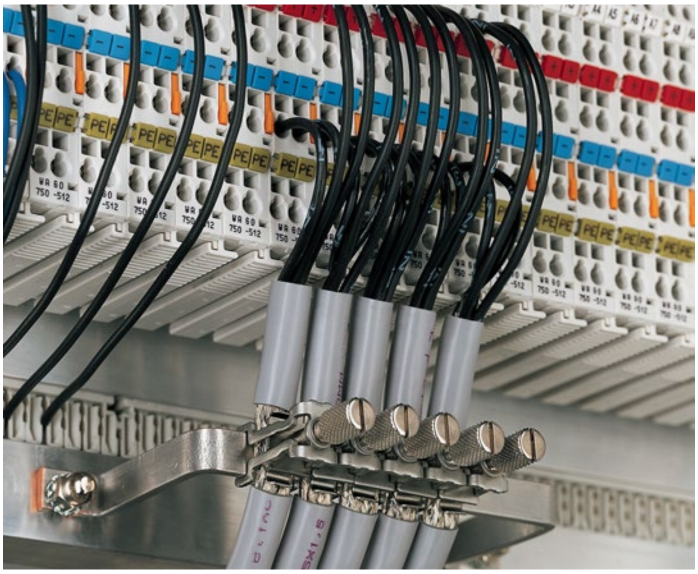


Figure 21: Application of the WAGO Shield Connecting System

4 Device Description

The CANopen Fieldbus Coupler, MCS 750-337 connects the WAGO-I/O-SYSTEM 750 to the CANopen fieldbus system.

Data is transmitted via PDOs and SDOs.

The CANopen Fieldbus Coupler, MCS recognizes all connected I/O modules and creates a local process image on this basis. Analog and specialty module data is sent via words and/or bytes; digital data is sent bit by bit.

The local process image is divided into two data zones containing the data received and the data to be sent. The process data can be sent via the CANopen bus to a control system for further processing. The process output data can be sent via the CANopen bus. The data of the analog modules is stored in the PDOs according to the order in which the modules are connected to the CANopen Fieldbus Coupler, MCS. The bits of the digital modules are sent byte by byte and also mapped in the PDOs. If the amount of digital information exceeds 8 bits, the CANopen Fieldbus Coupler, MCS automatically starts with a new byte.

All entries of the object directory can be mapped as required in the 32 Rx PDOs and 32 Tx PDOs. The complete input and output process image can be transmitted via SDOs.

Software can be used to set virtual “spacer modules” for possible future extensions (see section “Fieldbus Communication” > … > “Object 0x4500 – Empty Module Configuration”).

4.1 View

The view below shows the three parts of the device:

- The fieldbus connection is on the left side.
- LEDs for operation status, bus communication, error messages and diagnostics, as well as the service interface are in the middle area.
- The right side shows the power supply unit for the system supply and for the field supply of the attached I/O modules via power jumper contacts.
  LEDs show the status of the operating voltage for the system and field supply (jumper contacts).

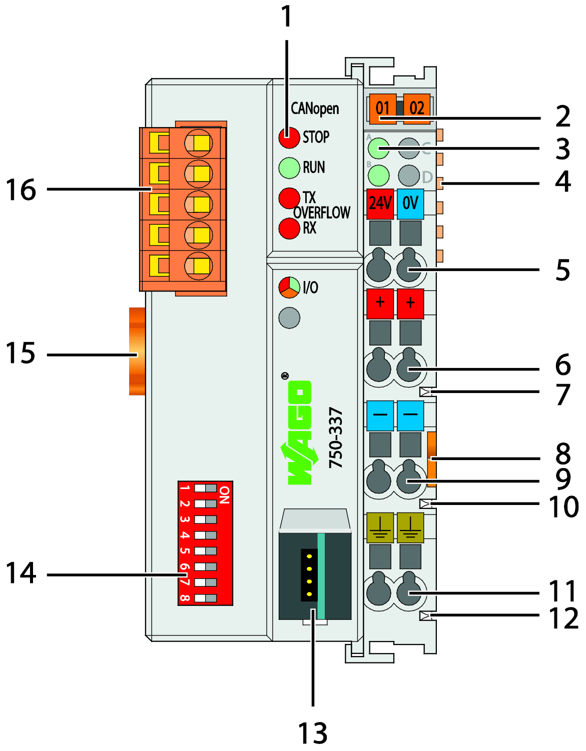


Figure 22: View CANopen Fieldbus Coupler

Table 12: Legend for Figure “View CANopen Fieldbus Coupler”

Pos.	Desig-nation	Meaning	Details see Section
1	STOP, RUN, TX, RX	Status LEDs Fieldbus	“Device Description” > “Display Elements”
2	---	Group marking carrier (retractable) with additional marking possibility on two miniature WSB markers	---
3	A, B or C	Status LED’s System/Field Supply	“Device Description” > “Display Elements”
4	---	Data Contacts	“Connect Devices” > “Data Contacts/Internal Bus”
5	24 V, 0 V	CAGE CLAMP® Connections System Supply	“Connect Devices” > “Connecting a conductor to the CAGE CLAMP®”
6	+	CAGE CLAMP® Connections Field Supply 24 VDC	“Connect Devices” > “Connecting a conductor to the CAGE CLAMP®”
7	---	Power Jumper Contact 24 VDC	“Connect Devices” > “Power Contacts/ Field Supply”
8	---	Unlocking Lug	“Mounting” > “Inserting and Removing Devices”
9	-	CAGE CLAMP® Connections Field Supply 0 V	“Connect Devices” > “Connecting a conductor to the CAGE CLAMP®”
10	---	Power Jumper Contact 0 V	“Connect Devices” > “Power Contacts/ Field Supply”
11	(Ground)	CAGE CLAMP® Connections Field Supply (Ground)	“Connect Devices” > “Connecting a conductor to the CAGE CLAMP®”
12	---	Power Jumper Contact (Ground)	“Connect Devices” > “Power Contacts/ Field Supply”
13	---	Service Interface (open flap)	“Device Description” > “Operating Elements”
14	---	Address Selection Switch	“Device Description” > “Operating Elements”
15	---	Locking Disc	“Mounting” > “Inserting and Removing Devices”
16	---	Fieldbus Data contacts connection, 231 Series (MCS)	“Device Description“ > “Connectors“

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.

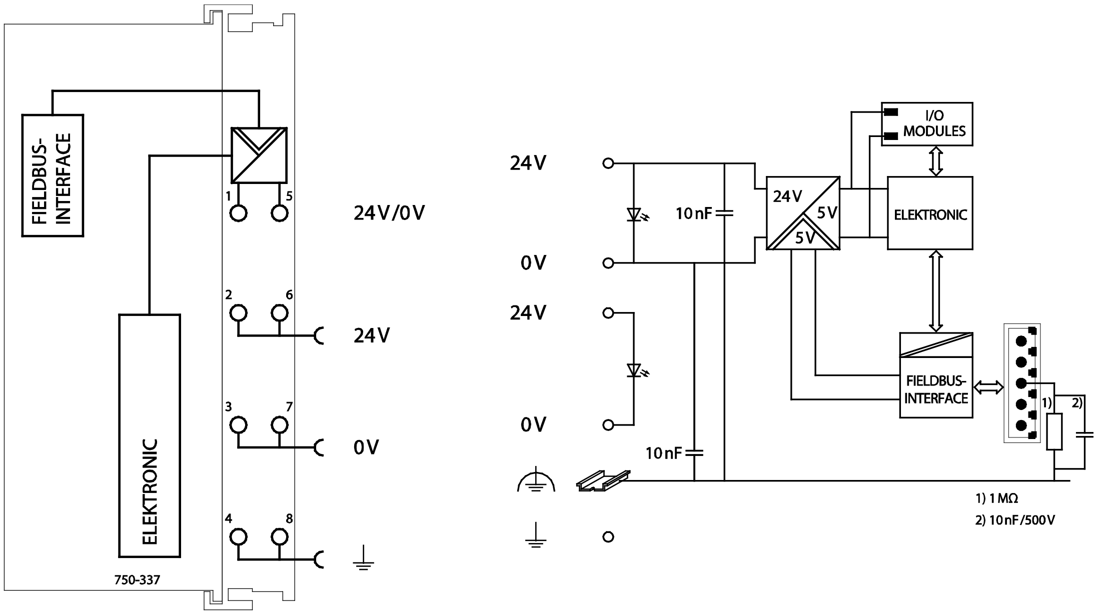


Figure 23: Device Supply

4.2.2 Fieldbus Connection

The CAN interface of the fieldbus coupler/controller 750-337 is designed as an open-style connection, *MCS* (*MULTI CONNECTION SYSTEM* Series 231).

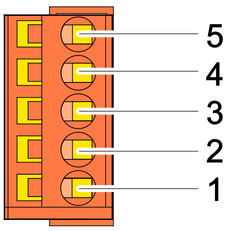


Figure 24: Fieldbus Connections, Series 231 *(MCS)*

Table 13: Pin assignment for the fieldbus connection, series 231 (*MCS*)

Pin	**Signal**	**Description**
5	N.C.	not used
4	CAN_H	CAN Signal $_{High}$
3	Drain Shield	Shield termination
2	CAN_L	CAN Signal $_{Low}$
1	GND	Ground

The connection point of the fieldbus coupler is lowered for mounting in an 80 mm-high switchgear cabinet after connector attachment.
DC/DC converters and optocouplers in the fieldbus interface provide electrical isolation between the fieldbus system and the electronics. The cable shield must be applied to CAN-shield for both fieldbus couplers. This is terminated to PE in devices with 1 MΩ (DIN rail contact). A low-impedance connection of the shielding to PE is possible only from the outside (e.g., by a supply module). The aim is a central PE contact for the entire CANopen bus cable shield.

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).

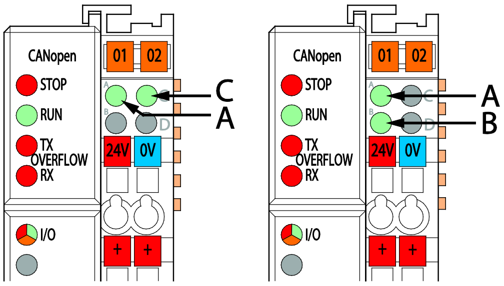


Figure 25: Display Elements

For the diagnostics of the different domains fieldbus, node and supply voltage, the LEDs can be divided into three groups:

Table 14: Display Elements Fieldbus Status

LED	Color	Meaning
STOP	red	Fieldbus coupler is in the STOP state
RUN	green	Shows the user whether the fieldbus coupler/controller is functioning correctly.
TX OVERFLOW	red	CAN transmit buffer has overflowed
RX OVERFLOW	red	CAN input buffer has overflowed

Table 15: 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.

Table 16: Display Elements Supply Voltage

LED	Color	Meaning
A	green	Indicates the status of the operating voltage – system
B or C	green	Indicates the status of the operating voltage – power jumper contacts (LED is manufacturing dependent either on position B or C)

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.

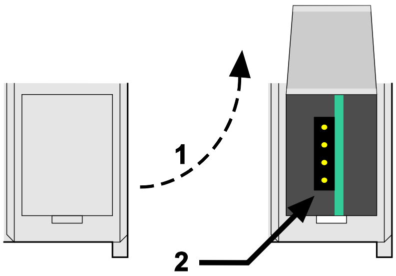


Figure 26: Service Interface (Closed and Opened Flap)

Table 17: 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.

4.4.2 Address Selection Switch

The DIP switch on the fieldbus coupler/controller is used to parameterize the fieldbus coupler/controller (baud rate setting) and to set the module ID. This module ID is used to calculate standard COB-IDs (e.g., PDOs 1...4, 1. Server SDO, etc.).

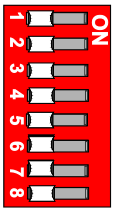


Figure 27: Setting the Station Address (Setting 0 Here)

The binary significance of the individual DIP switches increases in the direction of the switch number. Module ID 1 is set by DIP1 = ON, module ID 8 by DIP4 = ON, etc.

For the nodes of the WAGO-I/O-SYSTEM, module IDs can be set from 1 to 127.

4.4.2.1 Baud Rate Setting

The DIP switch is used to set the baud rate. 9 different baud rates are supported.

When the module ID = 0 (all slide switches OFF) and the supply voltage is ON, the fieldbus coupler/controller is put in configuration mode.
In this state, the baud rate currently set is displayed.

The top LED group (STOP, RUN, TX OVERFLOW and RX OVERFLOW) displays the baud rate, where STOP = switch 1, RUN = switch 2, TX OVERFLOW = switch 3 and RX OVERFLOW = switch 4.

The baud rate currently set is represented by slow flashing of the respective LEDs. In this state, the DIP switch can be used to set the new baud rate. The required slide switch is moved to the ON position.
The configuration is saved by moving sliding DIP8 to ON. After saving the configuration, the respective LED lights up continuously to indicate the new baud rate. The baud rate of 1 Mbaud is an exception. This baud rate is indicated by all 4 LEDs flashing/illuminated.

Example: 125 kBd: Tx overflow LED flashes/lights up
250 kBd: STOP and RUN LEDs flash/light up

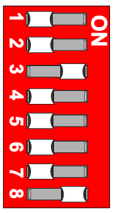


Figure 28: Example: Saving the Baud Rate 125 kBd

Data cannot be exchanged via CAN in this state.

Table 18: Baud Rate Setting

Switch	Function	1 MBd	800 kBd	500 kBd	250 kBd	125 kBd *)	100 kBd	50 kBd	20 kBd	10 kBd	indicated by the LEDs
1 (LSB)	Baud rate	0	1	0	1	0	1	0	1	0	STOP
2	Baud rate	0	0	1	1	0	0	1	1	0	RUN
3	Baud rate	0	0	0	0	1	1	1	1	0	TX OVERFLOW
4 (MSB)	Baud rate	0	0	0	0	0	0	0	0	1	RX OVERFLOW
5											
6											
7											
8	Apply	'off' -> 'on' : Apply the configuration set									

*) Factory setting

After setting / checking the baud rate, switch off the operating voltage because the DIP value set is only used as the module ID when calculating IDs when the power is ON. When OFF, the required module ID (1 as delivered) can be set on the DIP switch.

4.5 Technical Data

4.5.1 Device Data

Table 19: Technical Data – Device Data

Width	51 mm
Height (from upper edge of DIN 35 rail)	65* mm
Length	100 mm
Weight	approx. 195 g
Protection type	IP20

* from upper-edge of DIN 35 rail

4.5.2 System Data

Table 20: Technical Data – System Data

Number of subscribers within the CAN network (not including repeaters)	110
Number of I/O modules	64
Number of supported modules with mailbox functionality	max. 3 (with firmware version 10 to 15) max. 8 (starting from firmware version 16)
Transmission medium	shielded Cu cable based on EN 50170
Fieldbus segment length	30 m … 1000 m (depending on baud rate/cable)
Baud rate	10 kBaud … 1 MBaud
Bus connection	5-pin plug connector; series 231 (*MCS*); connector 231-305/010-000 included
Number of I/O modules that can be arranged side-by-side	63
Max. input process image	512 bytes
Max. output process image	512 bytes
Configuration	via PC or PLC
Number of PDOs	32 Tx / 32 Rx
Number of SDOs	2 Server SDOs
Communication profile	DS-301 V4.01
Device profile	DS-401 V2.0 Limit monitoring, flank-triggered PDOs, configurable response in the event of an error, DSP 405, NMT Master programmable using function blocks
COB-ID distribution	SDO, Standard
Node ID distribution	DIP switch

Table 20: Technical Data – System Data

Other CANopen features	NMT Slave, Minimum Boot-up, variables PDO mapping, emergency message, life guarding, empty module configuration

4.5.3 Power Supply

Table 21: Technical Data – Power Supply

Power supply	24 VDC (-15% … +20%)
Input current $_{max.}$	500mA at 24V
Efficiency of the power supply	87%
Internal current consumption	350mA at 5V
Total current for I/O modules	1650mA at 5V
Isolation	500V system/supply
Voltage via power jumper contacts	24 VDC (-15% … +20%)
Current via power jumper contacts	max. 10 ADC

4.5.4 Accessories

Table 22: Technical Data – Accessories

EDS Files	Download: *www.wago.com*
Miniature WSB Quick marking system	

4.5.5 Electrical Safety

Table 23: Technical Data - Electrical Safety

Clearance/Creepage distances	based on IEC 60664-1
Degree of contamination based on IEC 61131-2	2

4.5.6 Connection Type

Table 24: Technical Data – Field Wiring

Wire connection	CAGE CLAMP®
Cross section	0.08 mm² … 2.5 mm², AWG 28 … 14
Stripped lengths	8 mm … 9 mm / 0.33 in

Table 25: Technical Data – Power Jumper Contacts

Power jumper contacts	Spring contact, self-cleaning
Voltage drop at I $_{max.}$	< 1 V/64 modules

Table 26: Technical Data – Data Contacts

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

4.5.7 Climatic Environmental Conditions

Table 27: Technical Data – Climatic Environmental Conditions

Operating temperature range	0 °C … 55 °C
Operating temperature range for components with extended temperature range (750-xxx/025-xxx)	−20 °C … +60 °C
Storage temperature range	−25 °C … +85 °C
Storage temperature range for components with extended temperature range (750-xxx/025-xxx)	−40 °C … +85 °C
Relative humidity	Max. 5 % … 95 % without condensation
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 acc. to IEC 61131-2

Table 28: Technical Data – Mechanical Strength acc. to IEC 61131-2

Test specification	**Frequency range**	**Limit value**
IEC 60068-2-6 vibration	5 Hz ≤ f < 9 Hz	1.75 mm amplitude (permanent) 3.5 mm amplitude (short term)
	9 Hz ≤ f < 150 Hz	0.5 g (permanent) 1 g (short term)
	Note on vibration test: a) Frequency change: max. 1 octave/minute b) Vibration direction: 3 axes	
IEC 60068-2-27 shock		15 g
	Note on shock test: a) A Type of shock: half sine b) Shock duration: 11 ms c) Shock direction: 3x in positive and 3x in negative direction for each of the three mutually perpendicular axes of the test specimen	
IEC 60068-2-32 free fall	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 the basic version and all variations of 750-337 fieldbus couplers/controllers:

cULus UL508

Conformity Marking

Korea Certification MSIP-REM-W43-FBC750

The following Ex approvals have been granted to the basic version and all variations of 750-337 fieldbus couplers/controllers:

TÜV 07 ATEX 554086 X

I M2 Ex d I Mb
II 3 G Ex nA IIC T4 Gc
II 3 D Ex tc IIIC T135°C Dc

IECEx TUN 09.0001 X

Ex d I Mb
Ex nA IIC T4 Gc
Ex tc IIIC T135°C Dc

cULus ANSI/ISA 12.12.01
Class I, Div2 ABCD T4

The following Ex approvals have been granted to the basic version of 750-337 fieldbus coupler/controller:

Brasilian-Ex TUEV 12.1297 X
Ex nA IIC T4 Gc

The following ship approvals have been granted to 750-337 fieldbus coupler/controller:

ABS (American Bureau of Shipping)	
Federal Maritime and Hydrographic Agency	
BV (Bureau Veritas)	
DNV (Det Norske Veritas)	Class B
GL (Germanischer Lloyd)	Cat. A, B, C, D (EMC 1)
KR (Korean Register of Shipping)	
LR (Lloyd’s Register)	Env. 1, 2, 3, 4
NKK (Nippon Kaiji Kyokai)	
PRS (Polski Rejestr Statków)	
RINA (Registro Italiano Navale)	

The following ship approvals have been granted to 750-337/025-000 I/O module variations:

Federal Maritime and Hydrographic Agency	
GL (Germanischer Lloyd)	Cat. A, B, C, D (EMC 1)

Information

For more information about the ship approvals:
Note the “Supplementary Power Supply Regulations” section for the ship approvals.

4.7 Standards and Guidelines

750-337 meets the following requirements on emission and immunity of interference:

EMC CE-Immunity to interference	acc. to EN 61000-6-2
EMC CE-Emission of interference	acc. to EN 61000-6-4

750-337 meets the following requirements on emission and immunity of interference:

EMC marine applications-Immunity to interference	acc. to Germanischer Lloyd
EMC marine applications-Emission of interference	acc. to Germanischer Lloyd

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 50022 (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 medal 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 Rail

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

Table 29: WAGO DIN Rail

Order number	Description
210-113 /-112	35 x 7.5; 1 mm; steel yellow chromated; slotted/unslotted
210-114 /-197	35 x 15; 1.5 mm; steel yellow chromated; slotted/unslotted
210-118	35 x 15; 2.3 mm; steel yellow chromated; unslotted
210-198	35 x 15; 2.3 mm; copper; unslotted
210-196	35 x 7.5; 1 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.

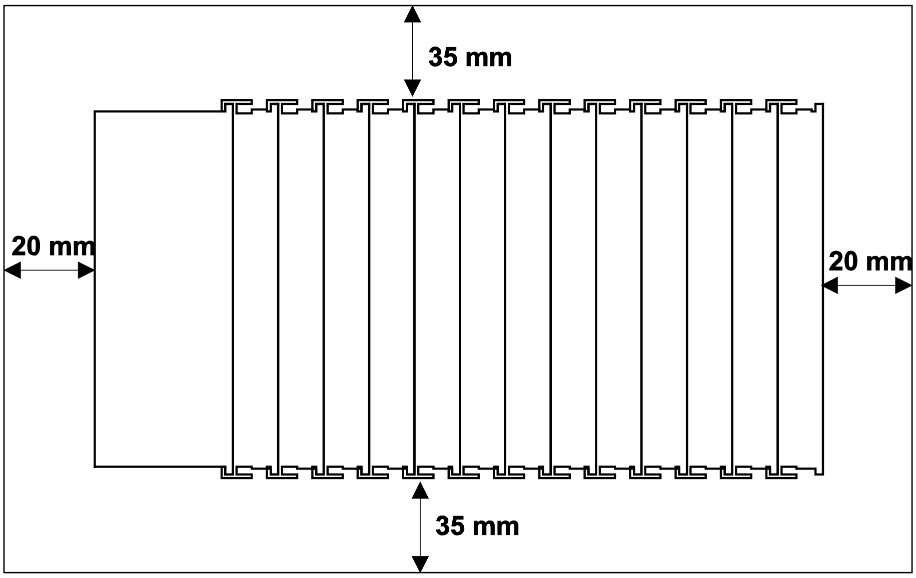


Figure 29: Spacing

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 750/753 are snapped directly on a carrier rail in accordance with the European standard EN 50022 (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/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.

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/controllers to guarantee proper data transfer.

5.6 Inserting and Removing Devices

NOTICE

Perform work on devices only if they are de-energized!
Working on energized devices can damage them. Therefore, turn off the power supply before working on the devices.

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 30: Release Tab Standard Fieldbus Coupler/Controller (Example)

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/controller or to the previous or possibly subsequent I/O module are engaged.

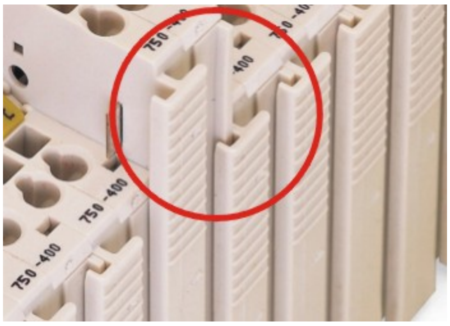


Figure 31: Insert I/O Module (Example)

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

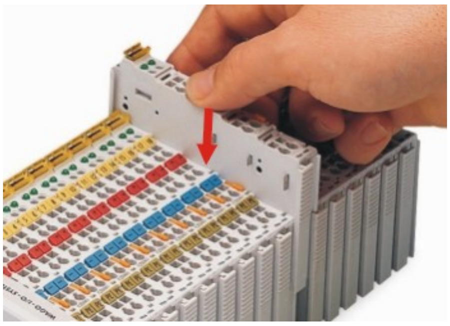


Figure 32: Snap the I/O Module into Place (Example)

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/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.

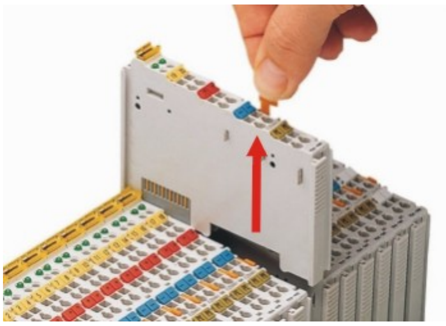


Figure 33: Removing the I/O Module (Example)

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

6 Connect Devices

6.1 Data Contacts/Internal 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 internal bus. It is comprised of 6 data contacts, which are available as self-cleaning gold spring contacts.

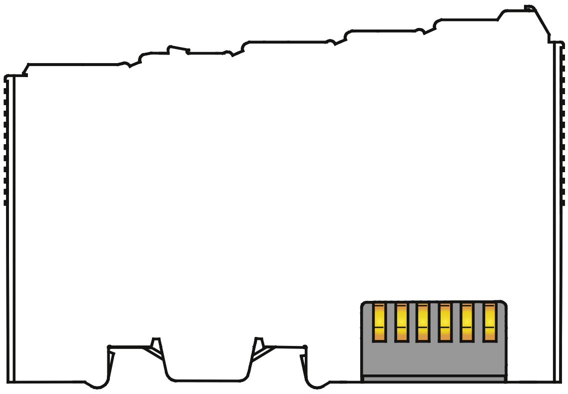


Figure 34: Data Contacts

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.

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.

Power jumper contacts

Blade	0	0	3	2
Spring	0	3	3	2

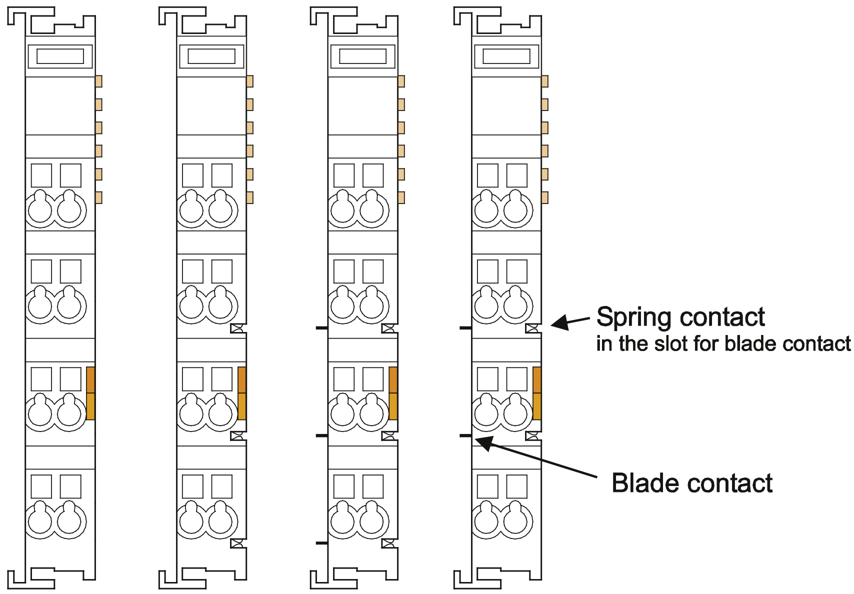


Figure 35: Example for the Arrangement of Power Contacts

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.

Exception:
If it is unavoidable to jointly connect 2 conductors, then you must use a ferrule to join the wires together. The following ferrules can be used:

Length:	8 mm
Nominal cross section $_{max.}$:	1 mm^2 for 2 conductors with 0.5 mm^2 each
WAGO product:	216-103 or products with comparable properties

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.

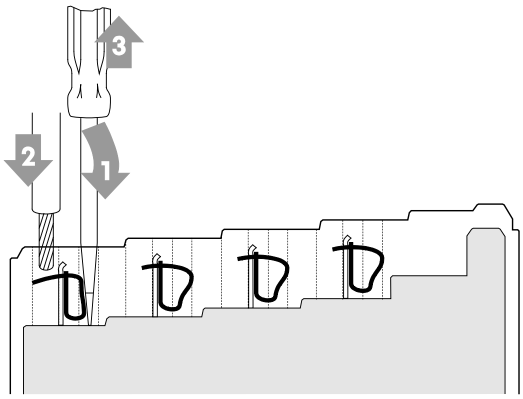


Figure 36: Connecting a Conductor to a CAGE CLAMP®

7 Function Description

7.1 Operating System

After master configuration and electrical installation of the fieldbus station, the system is operative.

The fieldbus coupler begins running up after switching on the power supply or after a reset.

Upon initialization, the fieldbus coupler determines the connected I/O modules and the current configuration. During this phase, the I/O LED flashes red.
After trouble-free startup, the fieldbus coupler enters “Fieldbus start” mode and the RUN LED flashes in green.
After bus data exchange, the I/O LED lights up green.
If an error occurs during start-up, the I/O LED flashes red and a flash code indicates the corresponding error message.

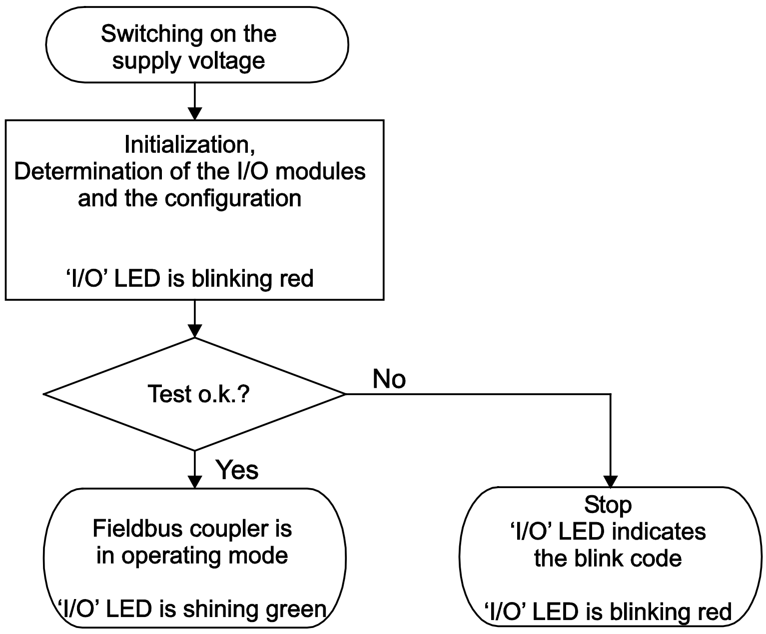


Figure 37: Operating System

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 Process Data Architecture

After turning on the supply voltage, the fieldbus coupler recognizes the I/O modules in the fieldbus node that have a data width > 0 bit.

In the maximum total length of a fieldbus node, up to 64 I/O modules can be connected to the fieldbus coupler. Any number of analog input/output modules and digital input/output modules can be arranged.

The data of the digital input/output modules is bit-oriented, i.e., data is exchanged bit by bit. The data of the analog input/output modules is byte-oriented, i.e., data is exchanged byte by byte.
The term “analog input/output module” stands for all I/O modules that are byte-oriented. This group includes counter modules, angle and distance measurement modules and communication modules.

Table 30: Data Width of the I/O Modules

Data width ≥ 1 byte/channel	**Data width = 1 bit/channel**
Analog input modules	Digital input modules
Analog output modules	Digital output modules
Input modules for thermocouples	Digital output modules with diagnostics (2 bit/channel)
Input modules for resistor sensors	Supply modules with fuse carrier/diagnostics
Pulse width output modules	Solid-state load relays
Interface modules	Relay output modules
Up/down counter	
I/O modules for angle and distance measurement	

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 populated by the fieldbus master with the process output data. The PII is populated by the fieldbus coupler with the process input data.

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

The data of the byte-oriented I/O modules is entered in the process image before the data of the bit-oriented I/O modules. The bits of the digital input/output 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.

Note

Expandable with Module Bus Extension Coupler and End Module!
With the use of the WAGO Module Bus Extension Coupler Module 750-628 and the End Module 750-627, it is possible to operate up to 250 modules on the device.

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 Data Exchange

Process data is exchanged via the communication objects with the CANopen fieldbus coupler.

Each object consists of a CAN telegram with a maximum of 8 bytes process data and a COB (Communication Object Identifier) ID that is unique within the network. These communication objects are used for transmission of data, triggering of events, signaling of error statuses, etc.

The parameters required for the communication objects as well as CANopen device parameters and data are stored in an object directory.

7.3.1 Fieldbus Coupler Communication Objects

The fieldbus coupler 750-337 supports the following communication objects:

- 32 Tx PDOs,
  for process data exchange of input data of the fieldbus node
- 32 Rx-PDOs,
  for process data exchange of output data of the field fieldbus node
- 2 Server SDOs,
  for the exchange of configuration data and for information on node state
- Synchronization object (SYNC),
  for network synchronization
- Emergency object (EMCY)
- Network management objects
  - Module control protocols
  - Error control protocols
  - Bootup protocol

7.3.2 Communication Interfaces

The CANopen fieldbus coupler has two interfaces for exchanging data:

- Interface to the fieldbus (master)
- Interface to the I/O modules

Data is exchanged between the fieldbus and the I/O modules. How the data is accessed from the fieldbus side depends on the fieldbus.

7.3.3 Memory Space

For the physical input/output data, memory space of 256 words (word 0 ... 255) is available in the fieldbus coupler.

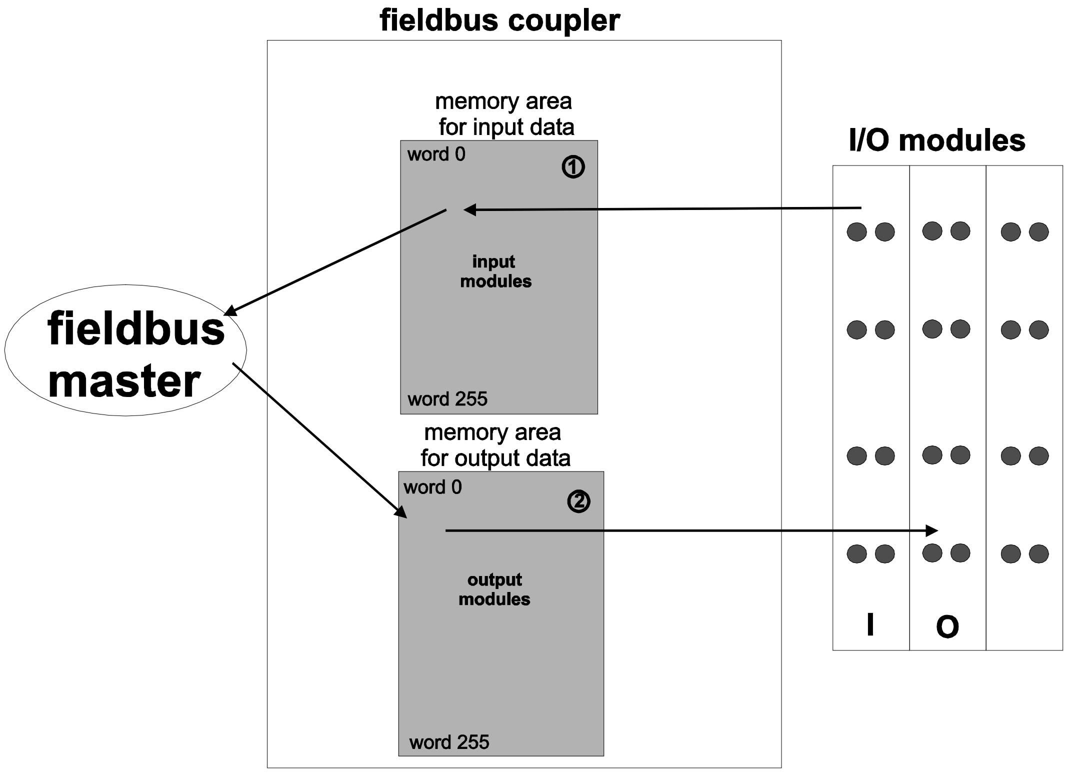


Figure 38: Memory Areas and Data Exchange

The fieldbus coupler process image contains the physical data for the bus modules.
These have a value of 0 ... 255 and word 512 ... 1275.

1 The input module data can be read by the CPU and by the fieldbus side.

2 Likewise, data can be written to the output modules from the CPU and the fieldbus side.

7.3.4 Addressing

After turning on the supply voltage, the data is mapped from the process image in an object directory. A CANopen fieldbus master uses the 16-bit indices and 8-bit sub-indices of the object directory to address data via PDOs or SDOs and to access the data.

The position of the data in the process image is therefore not directly significant for the CANopen user.

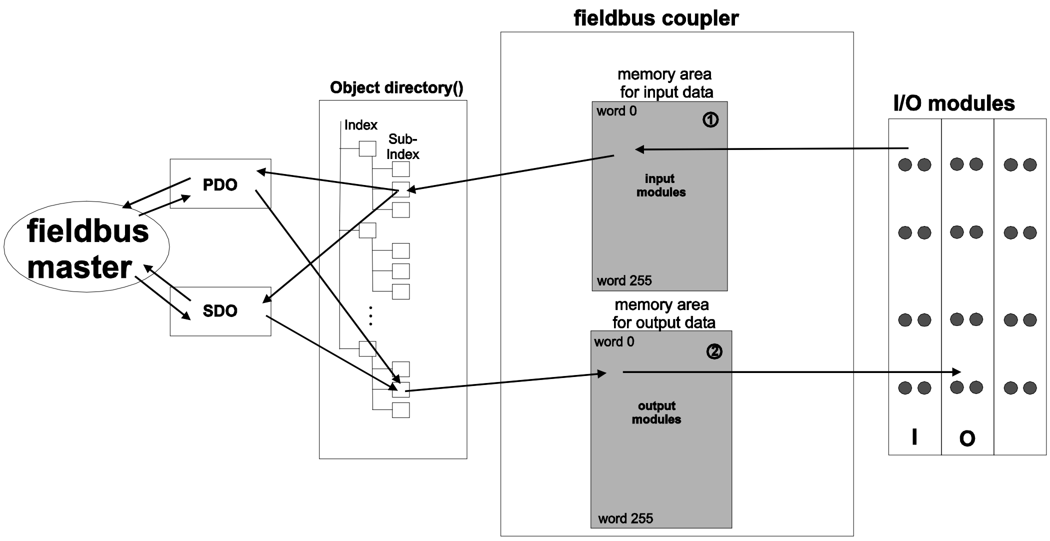


Figure 39: Fieldbus-Specific Data Exchange for a CANopen Fieldbus Coupler

7.3.4.1 Indexing of the I/O Module Data

If a customized configuration has been saved before initialization and the current connected I/O module configuration matches the last saved, this configuration is used when initializing.

Information

Additional information

An example for initialization of a customized configuration is explained in the section “Commissioning”.

In any other case, the object directory is occupied by the default configuration according to device profile DS-401 during initialization.

Entry in the object directory is carried out separately by data width (1 bit, 1 byte, 2 bytes, 3 bytes, etc.) and input and output.
The physical arrangement of the I/O modules on a node is arbitrary.

Table 31: Data Width of the I/O Modules

Data width ≥ 1 byte/channel	Data width = 1 bit/channel
Analog input modules	Digital input modules
Analog output modules	Digital output modules
Input modules for thermocouples	Digital output modules with diagnostics (2 bit/channel)
Input modules for resistor sensors	Supply modules with fuse carrier/diagnostics
Pulse width output modules	Solid-state load relays
Interface modules	Relay output modules
Up/down counter	
I/O modules for angle and distance measurement	

Note

Attention

For the number of input and output bits or bytes of the individual I/O modules, please refer to the corresponding descriptions of the I/O modules.

The digital I/O module data is considered first.

As CANopen does not transmit the data bit by bit, the digital input/output data of the I/O modules is combined in bytes and assigned to the respective index, digital input data to index 0x2000 (0x6000), digital output data to index 0x2100 (0x6200).

I/O module data that has a width of 1 byte or more is assigned to the corresponding indices in a similar manner.

This table provides an overview of the indices of the I/O module data.

Table 32: Indexing of the I/O Modules in the Object Directory

Data width	Input modules	Output modules
	Index	
1 bit digital	0x2000 (0x6000)	0x2100 (0x6200)
1-byte modules	0x2200	0x2300
2-byte modules	0x2400 (0x6401)	0x2500 (0x6411)
3-byte modules	0x2600	0x2700
4-byte modules	0x2800	0x2900
5-byte modules	0x3000	0x3100
6-byte modules	0x3200	0x3300
7-byte modules	0x3400	0x3500
8-byte modules	0x3600	0x3700
9+ byte modules	0x3800	0x3900

Each index has a maximum of 256 sub-indices (sub-index 0-255).

The number of data entries is given in sub-index 0, and the data is stored in blocks in the subsequent sub-indices.
The size of the blocks depends on the data width of the associated I/O module.

Table 33: Sub-Indexing of the I/O Module Data in the Object Directory

Sub-Index	Contents
0	Number of data blocks
1	First data block with data width of corresponding I/O module
2	Second data block with data width of the corresponding I/O module
…	…

Information

Additional information

Please refer to the “Object Directory” for a detailed description for setting the default configuration.

Note

Attention

If a node is changed or expanded, this may result in a new process image structure. In this case, the process data addresses also change. In case of an expansion, the process data of all previous modules has to be taken into account.

Example:

The I/O module configuration contains:

1. Five 2-channel digital input modules (e.g., 750-400)
2. One 4-channel digital output modules (e.g., 750-504)
3. Two 2-channel analog output modules with 2 bytes per channel (e.g., 750-552)

Table 34: For 1. Indexing the Data from the Five 2-Channel Digital Input Modules

Index	Sub-Index	Contents								Description
0x2000 (0x6000)	0	2								No. of dig. 8-bit input blocks
	1	D4.2	D4.1	D3.2	D3.1	D2.2	D2.1	D1.2	D1.1 *)	1. dig. input block
	2	0	0	0	0	0	0	D5.2	D5.1 *)	2. dig. input block

*) D1.1 = Data bit module 1 channel 1, D1.2 = Data bit module 1 channel 2, etc.

Table 35: For 2. Indexing the Data from the 4-Channel Digital Output Module

Index	Sub-Index	Contents								Description
0x2100 (0x6200)	0	1								No. of dig. 8-bit output blocks
	1	0	0	0	0	D1.4	D1.3	D1.2	D1.1 *)	dig. output block

*) D1.1 = Data bit module 1 channel 1, D1.2 = Data bit module 1 channel 2, etc.

Table 36: For 3. Indexing the Data from the two 2-Channel analog output modules

Index	**Sub-Index**	**Contents**	**Description**
0x2500 (0x6411)	0	4	Number of 2-byte special channels
	1	D1.1 *)	1. Output channel
	2	D1.2 *)	2. Output channel
	3	D2.1 *)	3. Output channel
	4	D2.2 *)	4. Output channel

*) D1.1 = Data word from module 1 channel 1, D1.2 = Data word from module 1 channel 2, etc.

8 Commissioning

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

Note

Good example!
This description is just an example and only serves to describe the procedure for a local start-up of a single CANopen fieldbus node.

The description includes the following steps:

1. Connecting PC and fieldbus nodes
2. Checking and setting the baud rate
3. Setting the module ID
4. Switching to the OPERATIONAL state
5. Enabling analog input data
6. User-specific mapping

8.1 Connecting PC and Fieldbus Nodes

Use a fieldbus cable to connect the installed CANopen fieldbus node to the CANopen fieldbus card in your PC and boot the PC.

8.2 Checking and Setting the Baud Rate

First, move all slide switches to the “OFF” position (module ID = 0) and then switch On the supply voltage (24 VDC power supply) on the fieldbus coupler/ controller.

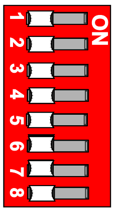


Figure 40: All Slide Switches in the “OFF” Position for Checking and Setting the Baud Rate

The baud rate currently set is then checked and is indicated by the respective LEDs flashing in the top LED group.

Note

Important note!
If not all slide switches are not in the “OFF” position when the voltage is switched on, the existing settings are applied as the module ID.

Now move the respective slide switch for the required baud rate to ON, e.g., slide switch 3 for baud rate 125 kBd.

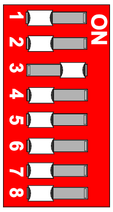


Figure 41: Setting Baud Rate 125 kBd

To save the new settings. also move slide switch 8 to ON.
Then switch off the supply voltage of the fieldbus coupler/controller.

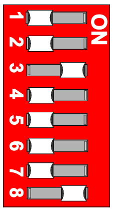


Figure 42: Saving the Baud Rate 125 kBd

8.3 Setting the Module ID

The module ID is set when the supply voltage is OFF. Move all slide switches again to the “OFF” position. Then move the respective slide switches for the required module ID to ON, e.g., slide switch 1 for module ID 1.

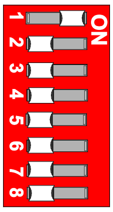


Figure 43: Setting Module ID 1

The fieldbus coupler/controller is in the INITIALIZATION state when the supply voltage is switched on.
The process image is created based on the I/O modules connected and the object directory initialized based on the default mapping if no customized configuration is saved.

Once the initialization phase has completed successfully, the fieldbus coupler/controller automatically switches to the PRE-OPERATIONAL state.

In this state, communication is possible via SDOs with which various settings can be made using your CAN Master software:

- You can set the fieldbus coupler/controller in the OPERATIONAL state directly.

Note

Note

Because PDO transmission of the analog input data is OFF by default, the analog input data is not taken into account.

- You can enable transmission of the analog input data, or
- You can select application-specific mapping.

8.4 Switching to the OPERATIONAL State

Use the **Start_Remote_Node** command from the network management objects to change the fieldbus coupler/controller from the PRE-OPERATIONAL to OPERATIONAL state. The fieldbus nodes for PDOs are ready for communication (see section “Start Remote Node”).

Note

Enable analog input data!

PDO transmission of analog input data is disabled by default. Therefore, the data is read only once and no longer updated. The analog input data must be enabled, so that it can be used via PDOs. Access via SDOs is possible at any time.

If not other settings are made, the fieldbus coupler/controller is ready for operation and communication is possible based on the default mapping (see section “Fieldbus Communication” > … > “Initialization”.

8.5 Enabling Analog Input Data

To prevent a CAN bus overflow with CAN messages, transmission of analog input data via PDOs is disabled by default. That means the object 0x6423 “Analog input global interrupt enable” has the default value 'FALSE' (0) (see section “Object 0x6423, Analog Input Global Interrupt Enable”). In general, you can enable transmission by setting object 0x6423 to 'TRUE' (1).
If necessary, use the “**Start Remote Node**” command to change the fieldbus coupler/controller from the PRE-OPERATIONAL to OPERATIONAL state. Transmission of analog input data via PDOs is then possible.

If not other settings are made, the fieldbus coupler/controller is ready for operation and communication is possible based on the default mapping (see section “Fieldbus Communication” > … > “Initialization”).

8.6 Application-Specific Mapping

In contrast to the default mapping, application-specific PDO mapping gives you the option to specify which data is to be transferred by the PDOs. The fieldbus coupler/controller should be in or put in the PRE-OPERATIONAL state using the "**Enter Pre-Operational**" NMT service.

The procedure for application-specific mapping is specifically explained in the example below.

Example:
Using Tx-PDO 2, the 3rd and 5th analog input channel with the data width of 2 bytes and the first 8-bit digital input group should be read. The CAN identifier 0x432 should be used for the transmission, which should be synchronous with every third SYNC object.
The standard CAN IDs are used for the SDOs. The setting is made at node 8.
xx is not evaluated.

1. First disable PDO mapping by setting the number of mapping objects in index 0x1A01, sub-index 0 ("Transmit PDO Mapping Parameter") to 0.

Table 37: Deactivate PDO Mapping

	CAN ID	**Data**
Send	608	0x2F 01 1A 00 00 xx xx xx
Receive	588	0x60 01 1A 00 xx xx xx xx

2. Enter the index, sub-index and object length of the application object into the Tx-PDO mapping parameter structure (index 0x1A01). A maximum of 8 bytes of data can be assigned per PDO.

Table 38: Writing the Mapping Parameter Structure

Application Object	**Index**	**Sub-Index**
3rd analog input channel	0x6401	3
5th analog input channel	0x6401	5
1st digital input group	0x6000	1

3. This requires the following structure in the mapping parameters of the 2nd Tx-PDO.

Table 39: Tx-PDO Mapping Parameter Structure, index 0x1A01

	Application Object		
Sub-Index	**Index**	**Sub-Index**	**Object Length (in bits)**
0	3		
1	0x6401	3	0x10
2	0x6401	5	0x10
3	0x6000	1	0x08

Note

Enter mapping parameter sub-index 1 … 8 first!
Enter mapping parameter sub-index 1 … 8 first and then the number of valid sub-indices in sub-index 0!

These objects are stored using SDO transmissions:

Table 40: Mapping the 3rd Analog Input Channel

	CAN ID	Data	
Send	0x608	0x23	01 1A 01 10 03 01 64
		23	0 data bytes invalid
		011 A	Index (low byte first)
		01	Sub-Index
		10	Data width of the analog channel
		03	Sub-index where the 3rd analog channel is in the Manufacturer Device Profile
		01 64	Index (low byte first) where the 3rd analog channel is in the Manufacturer Device Profile
Receive	0x588	0x60	01 1A 01 xx xx xx xx
		60	OK
		011 A	Index (low byte first)
		01	Sub-Index

Table 41: Mapping the 5th Analog Input Channel

	CAN ID	Data
Send	0x608	0x23 01 1A 02 10 05 01 64
Receive	0x588	0x60 01 1A 02 xx xx xx xx

Table 42: Mapping the 1st Digital Input Group

	CAN ID	Data
Send	0x608	0x23 01 1A 03 08 01 00 60
Receive	0x588	0x60 01 1A 03 xx xx xx xx

Table 43: Number of Mapping Objects = 3, Enter in Sub-Index 0

	CAN ID	Data
Send	0x608	0x2F 0x2F 01 1A 00 03 xx xx xx
Receive	0x588	0x60 0x60 01 1A 00 xx xx xx xx

4. To change the communication parameters, disable the PDO that you want to map.
   In this example, that is Tx-PDO2.
   Write the value 0x80000000 into the object with index 0x1801, sub-index 01 ("Transmit PDO Communication Parameter").

Table 44: Disable PDO

	CAN ID	Data
Send	608	0x23 01 18 01 00 00 00 80
Receive	588	0x60 01 18 01 xx xx xx xx

5. Write the communication parameters into the object with index 0x1801, sub-index 1 to 3 ("Transmit PDO Communication Parameter").

The transmission type is 3 (synchronous transmission with every third SYNC object).

Table 45: Entering the Communication Parameters

Tx-PDO-Communication Parameter, Index 0x1801		
Sub-Index	**Value**	**Explanation**
0	3	Number of supported entries in the record
1	0x432	COB-ID used by PDO
2	3	Transmission type
3	0	Inhibit time

Table 46: Sub-Index 3: Inhibit Time = 0

	CAN ID	**Data**
Send	0x608	0x2B 01 18 03 00 00 xx xx
Receive	0x588	0x60 01 18 03 xx xx xx xx

Table 47: Sub-Index 2: Transmission Type = 3

	CAN ID	**Data**
Send	0x608	0x2F 01 18 02 03 xx xx xx
Receive	0x588	0x60 01 18 02 xx xx xx xx

Table 48: Sub-Index 1: Set COB-ID = 432 of the PDO and set PDO from Invalid to Valid

	CAN ID	**Data**
Send	0x608	0x23 01 18 01 32 04 00 00
Receive	0x588	0x60 01 18 01 xx xx xx xx

6. After putting the fieldbus coupler/controller in the OPERATIONAL state using the “Start Remote Node” command, the PDOs are active and the Tx-PDO object can be used to transmit data.

9 Diagnostics

9.1 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).

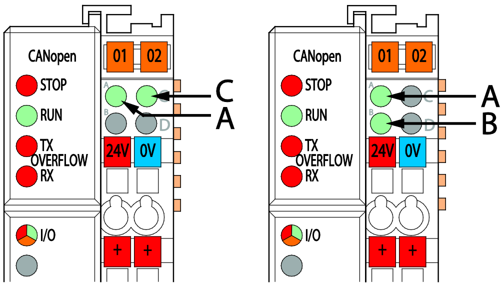


Figure 44: Display Elements

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 49: LED Assignment for Diagnostics

Diagnostics area	LEDs
Fieldbus status	• STOP • RUN • TX OVERFLOW • RX OVERFLOW
Node status	• I/O
Supply voltage status	• A • B or C (LED position depends on manufacturing)

9.1.1 Evaluating Fieldbus Status

Communication status via fieldbus is indicated by the upper LED group, STOP, RUN, TX OVERFLOW and RX OVERFLOW.

Table 50: Fieldbus Diagnostics – Solution in Event of Error

STOP	RUN	TX OVERFLOW	RX OVERFLOW	Explanation	Remedy
OFF	OFF	OFF	OFF	No function or self-test	Check supply voltage (24 V and 0 V), wait for self-test
OFF	Slow flashing	X	X	Module is in the PRE-OPERATIONAL state	
OFF	Illuminated	X	X	Module is in the OPERATIONAL state	
Illuminated	OFF	X	X	Module is in the STOPPED state or there are serious fieldbus-independent errors (e.g., an I/O module was pulled), incorrect configuration	In the event of fieldbus-independent errors: Check the I/O modules, reset the node, check the same in the event of a configuration error
X	X	X	Illuminated	The CAN input buffer has overflowed, data loss should be expected	Increase time between two telegrams
X	X	Illuminated	X	The CAN transmit buffer has overflowed, data loss should be expected	Check the bus system configuration, increase the sending priority of the module
X	X	Rapid flashing alternating with RX OVERFLOW	Rapid flashing alternating with TX OVERFLOW	CAN fieldbus coupler/controller has exceeded the warning level, too many error telegrams	Check baud rate, check bus connection, connect min. 2 CAN modules to the network
OFF	Rapid flashing	X	X	Module is in the PRE-OPERATIONAL state, Sync/Guard message / heartbeat has failed	Switch to the OPERATIONAL state and continue Sync/Guard message / heartbeat
Rapid flashing	Rapid flashing	X	X	Module is in the OPERATIONAL state, Sync/Guard message / heartbeat has failed	Continue Sync/Guard message / heartbeat
Rapid flashing	OFF	X	X	Module is in the STOPPED state, Sync/Guard message / heartbeat has failed	Switch to the OPERATIONAL state and continue Sync/Guard message / heartbeat

9.1.2 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 51: 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 indicate 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 internal bus or of a internal bus error.	Note the following flashing sequence.
red cyclical flashing	Up to three successive flashing sequences indicate internal data 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 internal 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 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.

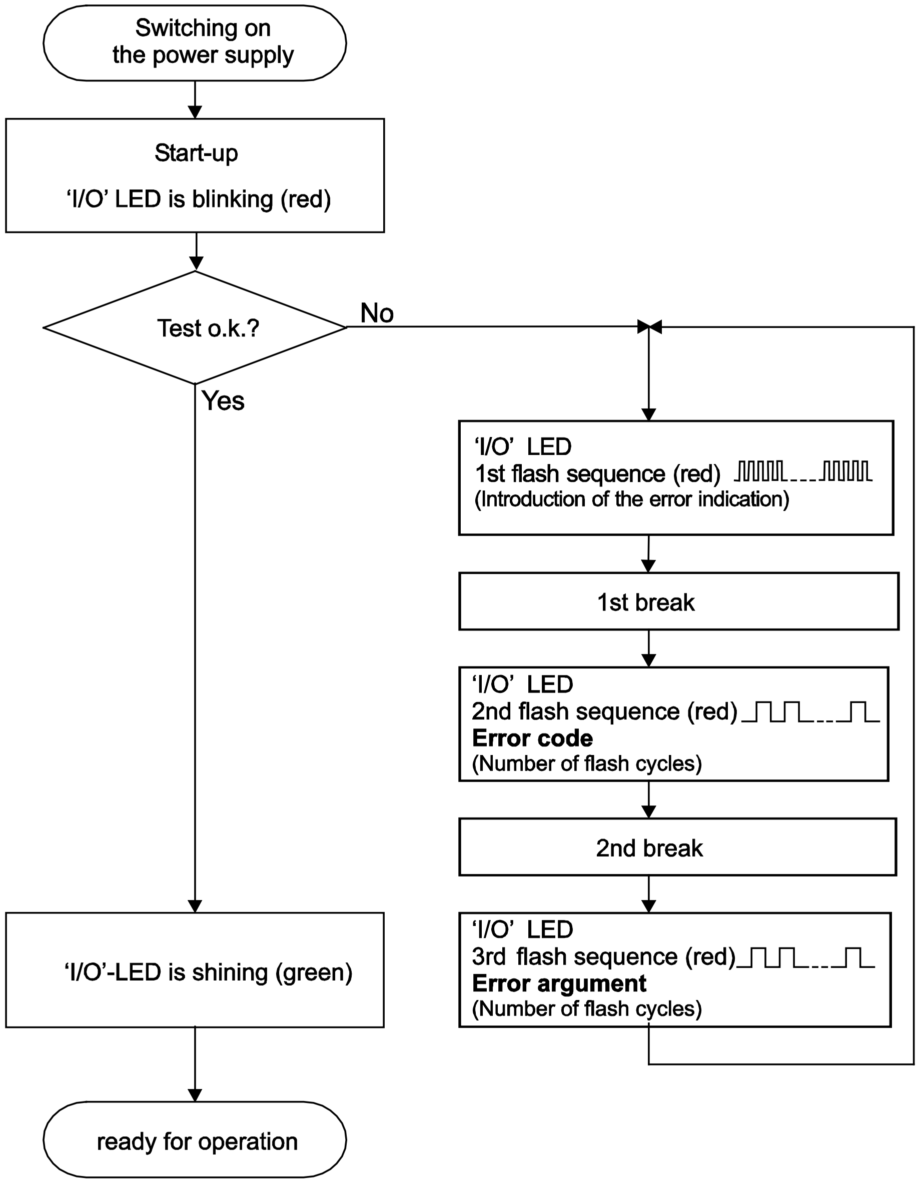


Figure 45: Node Status – I/O LED Signaling

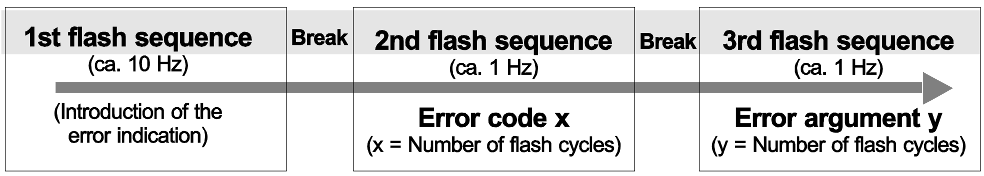


Figure 46: Error Message Coding

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 internal data 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 52: 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 attached I/O modules.	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 persists, 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 controller). - 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 controller). 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 supply for the node. 2. Replace the fieldbus coupler and turn the power supply on again.
4	Fault when writing in the serial EEPROM.	1. Turn off the power supply for the node. 2. Replace the fieldbus coupler and turn the power supply on again.
5	Fault when reading the serial EEPROM	1. Turn off the power supply for the node. 2. Replace the fieldbus coupler and turn the power supply on again.

Table 52: Blink Code Table for the 'I/O' LED Signaling, Error Code 1

Error code 1: "Hardware and configuration error"		
Error Argument	**Error Description**	**Solution**
6	The I/O module configuration after AUTORESET differs from the configuration determined the last time the fieldbus coupler was powered up.	1. Restart the fieldbus coupler by turning the power supply off and on.
7	Invalid hardware-firmware combination.	1. Turn off the power supply for the node. 2. Replace the fieldbus coupler and turn the power supply on again.
8	Timeout during serial EEPROM access.	1. Turn off the power supply for the node. 2. Replace the fieldbus coupler and turn the power supply on again.
9	Bus controller initialization error	1. Turn off the power supply for the node. 2. Replace the fieldbus coupler and turn the power supply on again.
10 ... 13	not used	
14	Maximum number of gateway or mailbox modules exceeded	1. Turn off the power supply for the node. 2. Reduce the number of corresponding modules to a valid number.
15	Permitted number of modules with min. 9-byte data length per channel exceeded or max. PA size exceeded by modules with min. 9-byte data length per channel	1. Turn off the power supply for the node. 2. Reduce the number of corresponding modules to a valid number.
	Note **Note process image size limit!** Please note the process image size limit when configuring a node with analog input/output modules and I/O modules that have mailbox functionality. Depending on the overall configuration of all I/O modules of a node, the maximum number of object directory entries may be exceeded in some cases.	

Table 53: Blink Code Table for the I/O LED Signaling, Error Code 2

Error code 2: -not used-		
Error Argument	**Error Description**	**Remedy**
-	Not used	-

Table 54: Blink Code Table for the I/O LED Signaling, Error Code 3

Error code 3: "Protocol error, internal bus"		
Error Argument	**Error Description**	**Remedy**
-	Internal bus communication interrupted, faulty module cannot be identified	--- Are there power supply modules with bus power supply (750-613) in the node? --- 1. Check that these I/O modules are supplied correctly with power. 2. Determine this by the state of the associated status LEDs. --- Are all I/O modules connected correctly or are there any 750-613 I/O modules in the node? --- 1. Identify 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 supply 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. 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.

Table 55: Blink Code Table for the I/O LED Signaling, Error Code 4

Error code 4: "Physical error, internal bus"		
Error Argument	**Error Description**	**Remedy**
-	Internal bus data transmission error or interruption of the internal bus at the fieldbus coupler	1. Turn off the power supply for the node. 2. Plug in an internal bus with process data behind the fieldbus controller. 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 supply and plug the end module into the middle of the first half of the node (toward the fieldbus controller). - 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 controller). 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 internal bus behind the nth I/O module with process data.	1. Turn off the power supply for the node. 2. Replace the (n+1) I/O module with 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 56: Blink Code Table for the I/O LED Signaling, Error Code 5

Error code 5: "Initialization error, internal bus"		
Error Argument	**Error Description**	**Remedy**
n*	Error during register communication during internal bus initialization	1. Turn off the power supply for the node. 2. Replace the (n+1) I/O module with 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 57: Blink Code Table for the I/O LED Signaling, Error Code 6

Error code 6: "Node configuration error"		
Error Argument	**Error Description**	**Remedy**
n	I/O module configured incorrectly in the empty module mapping	1. Check the empty module mapping

Table 58: Blink Code Table for the I/O LED Signaling, Error Code 9

Error code 9: "CPU Trap error"		
Error Argument	**Error Description**	**Solution**
1	Illegal Opcode	Fault in the program sequence. 1. Please contact the I/O Support.
2	Stack overflow	
3	Stack underflow	
4	NMI	

Table 59: Blink Code Table for the I/O LED Signaling, Error Code 11

<table>
<tr><th colspan="3">Error code 11: “Error in I/O modules with mailbox functionality”</th></tr>
<tr><th>Error Argument</th><th>Error Description</th><th>Remedy</th></tr>
<tr><td rowspan="2">1</td><td>Too many I/O modules with mailbox functionality are plugged in</td><td>1. Reduce the number of mailbox modules.</td></tr>
<tr><td colspan="2">Note
Please note the firmware version.
The fieldbus controller with firmware version 11 supports 3 I/O modules with mailbox functionality.
Firmware version 12 supports 8 I/O modules with mailbox functionality.

Note
Note process image size limit!
Please note the process image size limit when configuring a node with analog input/output modules and I/O modules that have mailbox functionality.
Depending on the overall configuration of all I/O modules of a node, the maximum number of object directory entries may be exceeded in some cases.</td></tr>
<tr><td>2</td><td>Maximum mailbox size exceeded</td><td>1. Reduce the size of the mailbox.</td></tr>
<tr><td>3</td><td>Maximum PA size exceeded due to connected I/O modules with mailbox functionality</td><td>1. Reduce the data width of I/O modules with mailbox functionality.</td></tr>
</table>

9.2 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.

9.2.1 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”.

9.2.2 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”.

9.2.3 Internal Data Bus Error

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

In addition, an internal data 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 an internal data bus error by sending a blink code. To send the blink code, the fieldbus coupler uses the I/O LED.

10 Fieldbus Communication

10.1 CANopen

10.1.1 Description

CAN (Controller Area Network) was developed in the mid-1980s for data transmission in automobiles. The “Data Link Layer” is defined in the CAN specification. This is the physical layer and data link layer. The telegram structure is described, but nothing is said about the application layer.
In contrast, CAL describes the application layer or the significance of the transmitted data. CAL is a general description language for CAN networks and makes a number of communication services available.

CANopen is a network concept based on CAN serial bus system. As a standard application level, CANopen is defined by specification DS 301 of CiA (CAN in Automation).

Network management powers up the network in a simplified manner. This network can be extended by the user as required.

CAN is a multi-master bus system. Contrary to other fieldbus systems, the modules connected to the bus are not addressed, but the messages are identified. The devices are allowed to send messages whenever the bus is available. Bus conflicts are solved by assigning messages a specific priority. This priority is determined by the COB-ID (Communication Object Identifier). It is uniquely assigned to a communication object. The smaller the identifier is, the higher the priority. Thus, communication without bus master module is also possible.

Each bus node decides by itself when it wants to send data. However, it is also possible to request other bus nodes to send data. This request happens via the so-called “Remote Frame”.

The CANopen specificiation (DS 301) defines the technical and functional characteristics with which the distributed field automation devices can be networked.

Information

More information about CAN

The CiA organization (CAN in Automation) makes other documents on the INTERNET available to its members at www.can-cia.de.

10.1.2 Network Configuration

10.1.2.1 Transmission Medium

10.1.2.1.1 Cable Type

The basis for the physical implementation of CAN a bus medium.

Both the bus connection and bus medium are specified in CAN according to ISO 11898 (CAN High-Speed).

According to the cable specification, twisted pair with an impedance of 108 to 132 Ohm is recommended as the medium.

Twisted-pair is inexpensive, easy to handle and allows simple bus-shaped wiring.

The WAGO CANopen fieldbus nodes are intended for wiring with shielded copper cable (3 × 0.25 mm²).

When dimensioning the electric bus medium, it is important to note the maximum bus length and the required cable cross-section.

10.1.2.1.2 Maximum Bus Length

The bus lengths are mainly limited by the signal propagation delay and must therefore be adjusted to the baud rate:

Table 60: Maximum Bus Lengths Dependent on Preset Baud Rate

Baud rate	**Bus length**
1 MBd	30 m
800 kBd	50 m
500 kBd	100 m
250 kBd	250 m
125 kBd	500 m
≤ 50 kBd	1000 m

10.1.2.1.3 Required Cable Cross-Section

The cross-section of the cable should be selected based on the cable length and the number of connected nodes.

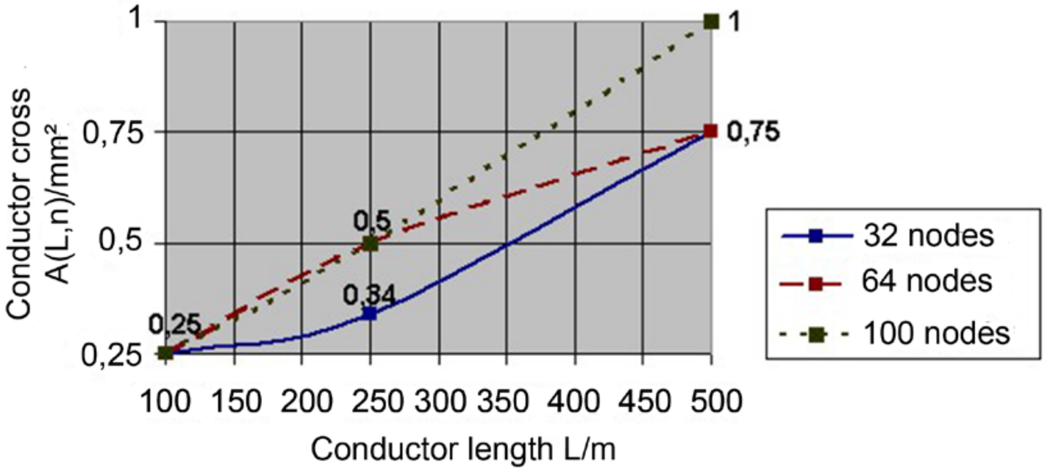


Figure 47: Cable Cross-Section Based on the Cable Length and the Number of Nodes

10.1.2.2 Cabling

A WAGO fieldbus node is connected to the CANopen cable using the respective fieldbus connector (*MCS* or D-Sub).

For wiring with shielded copper cable (3 × 0.25 mm²), the corresponding fieldbus connector with the CAN_High, CAN_Low and CAN_GND connections is used. CAN_High and CAN_Low are two physically different bus signal levels. CAN_GND is the common reference potential.

The cable's shielding can be applied to the "drain" connection.
This is terminated to PE in devices with 1 MΩ (DIN rail contact). A low-impedance connection of the shielding to PE is possible only from the outside (e.g., by a supply module). The aim is a central PE contract for the entire CANopen bus cable shield.

Note

Shield with the WAGO Shield Connecting System!
WAGO offers the shield connecting system (series 790) for optimal connection between fieldbus cable shielding and functional ground.

Each CANopen node forms the U_{Diff} differential voltage from the CAN_High and CAN_Low as follows: $U_{Diff} = U_{CAN\_High} - U_{CAN\_Low}$.
The differential signal transmission offers the benefit of insensitivity to common mode interferences and ground offsets between the nodes.

If the bus signal levels are found to be recessive, a voltage of 2.5 V is present between CAN_Low and CAN_GND, as well as between CAN_High and CAN_GND.

The differential voltage is 0 V.

If the bus signal levels are found to be dominant, a voltage of 1.5 V is present between CAN_Low and CAN_GND, and a voltage of 3.5 V between CAN_High and CAN_GND.
The differential voltage is approx. 2 V.

Information

Do not switch data lines, bus terminating resistor is important!
When connecting nodes, do not switch the data lines. Always use a bus terminating resistor of 120 Ohm to terminate the bus cable on the end of the cable to prevent reflections and transmission problems. This is also required for very short cable lengths.

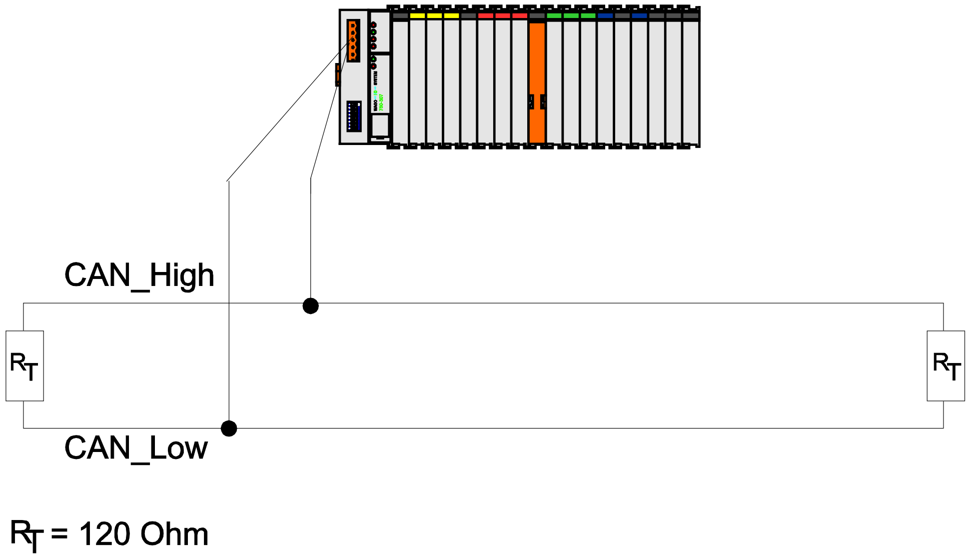


Figure 48: Connection Priciple of a WAGO Fieldbus Node to the CAN Bus

Before connecting the fieldbus node to the network, check the installation. You can test the physical connection at any point in the CAN bus using an ohmmeter. Before taking the measurement, you must disconnect all connected devices up to the terminating resistors.

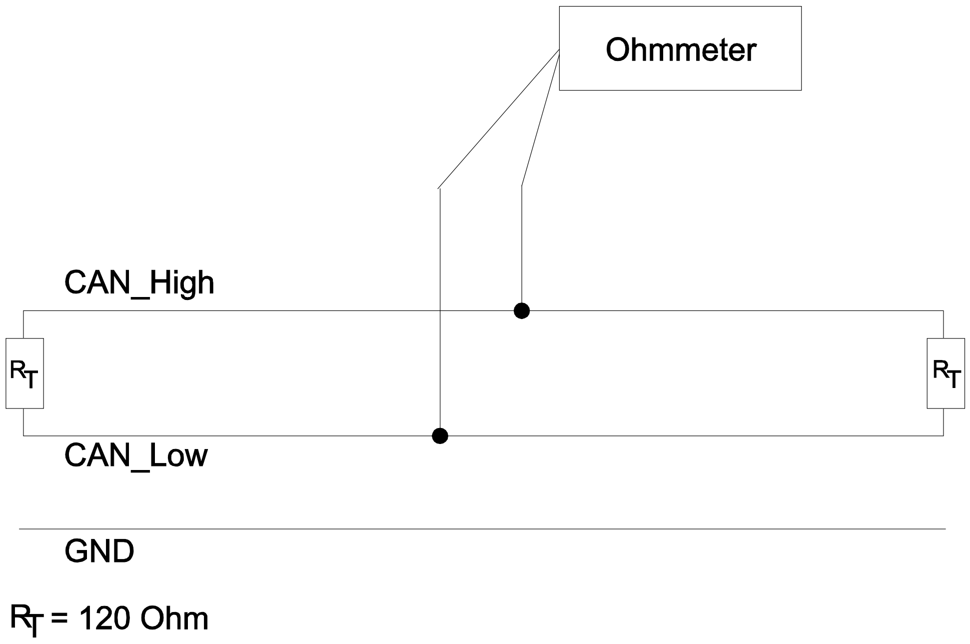


Figure 49: Principle of Measurement for Checking the CAN Bus Before Cabling

Table 61: Measurement

Measurement between:	**Measured Value**	**Explanation**
GND & CAN_L	Infinite	OK
	0	Short circuit between GND & CAN_L
GND & CAN_H	Infinite	OK
	0	Short circuit between GND & CAN_H
CAN_L & CAN_H	approx. 60 Ω	OK, 2 terminating resistors in the bus
	approx. 120 Ω	Only 1 terminating resistor in the bus
	< 50 Ω	More than 2 terminating resistors in the bus

Because the CAN bus can be designed as a 2-wire bus, bus fault management detects a break or short circuit in a line by asymmetric operation.

Information

More information about cable specifications

CiA makes documents about specifications, i.e., cable specifications, available on the INTERNET at http://www.can-cia.de.

10.1.2.3 Topology

To set up a simple CANopen network, you need a master (PC with PC fieldbus card CANopen), a connecting cable and a 24 VDC power supply unit in addition to a CANopen fieldbus node.

The CANopen network is set up as a linear topology with terminating resistors (120 Ohm).

In systems with more than two stations, all nodes are wired in parallel. This requires the bus cable to be looped through without disruption. The maximum length of a line branch should not exceed 0.3 m.

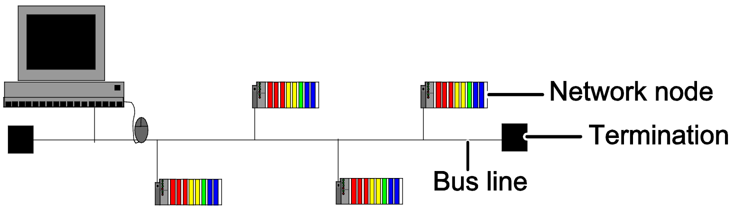


Figure 50: Bus Topology of a CANopen Network

All devices in the network communicate at the same baud rate. The bus structure makes it possible to couple and decouple stations or to start up the system step by step in a non-reactive manner.

Later upgrades have no effect on stations that are already in operation. If a device malfunctions or is added to the network, it is automatically detected.

Router nodes also allow branching from the linear bus and thus, setup of hierarchal network structures.

Repeaters can be used to increase the maximum possible number of 110 nodes and to increase the network reach (bus length). Although the network reach depends on the transmission rate, CAN can also be used for extensive networks. The achievable data rates are of the same order of magnitute as for other bus systems. Due to signal delay, the maximum possible cable length is reduced to 20 to 30 m per repeater.

10.1.2.4 Interface Modules

All WAGO CANopen fieldbus nodes operate as slaves in a network. Master operation is performed by a central control system, such as PLC, NC or RC. The fieldbus devices are linked via interface modules.

Interface modules for programmable logic controllers (PLCs) and PC interface cards for CANopen are available from different manufacturers.

10.1.2.5 Configuration Software

The interface modules must be configured with the specific station data to enable links to be set up between the PLC and the fieldbus devices.

The software for configuration, start-up and diagnostics is part of the delivery of the interface modules or PC cards or is also in the CAN Master software used (e.g., SyCon from Hilscher GmbH).

The data required for the WAGO CANopen fieldbus couplers and controllers is available in the configuration software with integration of the EDS (Electrical Data Sheet) files.

Note

Download the EDS files!

The EDS files for CANopen can be downloaded at http://www.wago.com → Online Catalogs/eShop → Catalogs → WAGO Product Catalog → Automation Components

10.1.3 Network Communication

In CANopen, the communication objects transmit data, trigger events, signal errors, etc. Each communication object is assigned a unique COB-ID (Communication Object Identifier) in the network.
The COB-ID based on device profile DS-401 is assigned according to the following table.

Table 62: COB-ID Assignment

10	9	8	7	6	5	4	3	2	1	0
Function code				Node identifier (0[= all], 1-127)						

Table 63: Communication Objects

Communication Objects		Function Code		Resulting COB-ID		Object Directory Index
		dec	bin	dec	bin	hex
Broadcast messages	NMT commands	0	0000	0		
	Sync message	1	0001	128	80	1005, 1006, 1007
	System time	2	0010	256	100	1012, 1013
Peer-to-Peer messages	Alarm objects	1	0001	129-255	81-FF	1014, 1015
	Tx-PDO1	3	0011	385-511	181-1FF	1800
	Rx-PDO1	4	0100	513-639	201-27F	1400
	Tx-PDO2	5	0101	641-767	281-2FF	1801
	Rx-PDO2	6	0110	769-895	301-37F	1401
	Tx-PDO3	7	0111	897-1023	381-3FF	1802
	Rx-PDO3	8	1000	1025-1151	401-47F	1402
	Tx-PDO4	9	1001	1153-1279	481-4FF	1803
	Rx-PDO4	10	1010	1281-1407	501-57F	1403
	Tx-SDO	11	1011	1409-1535	581-5FF	1200
	Rx-SDO	12	1100	1537-1663	601-67F	1200
	Node monitoring	14	1110	1793-1919	701-77F	100C, 100D 1016, 1017

The parameters required for the communication objects and the parameters/data of the CANopen devices are stored in the object directory.

Which and how many objects are supported by the node depends on the respective fieldbus coupler/controller.

Apart from some specific objects, e.g., for network management (NMT), synchronization (SYNC) or error messages (EMCY), the communication profile contains contains the two object types PDO and SDO.

The PDOs (Process Data Objects) are used to transmit real-time data and the SDOs (Service Data Objects) allow read/write access to the object directory.

10.1.3.1 Communication Objects

10.1.3.1.1 Process Data Object - PDO

PDOs contain, e.g., real-time data with high-priority identifiers. The data telegrams consist of a maximum of 8 bytes. The can be freely exchanged between the individual modules. Data exchange can be event-driven or synchronized.

Event-driven data exchange can reduce the bus load significantly, i.e., high communication performance is achieved at a low baud rate. The different modes can also be operated mixed (see section “Fieldbus Communication” > … > “Object 0x1400 – 0x141F – Receive PDO Communication Parameter”).

10.1.3.1.1.1 PDO Protocol

This protocol is used to transmit data to/from the fieldbus coupler/controller with no protocol overhead. PDOs only consist of the CAN identifier and the data field. A PDO contains no additional protocol information. The data content is determined by the mapping parameters, and the transfer type is determined by the communication parameters.

A distinction is made between Rx-PDO (receive PDO) and Tx-PDO (transmit PDO).

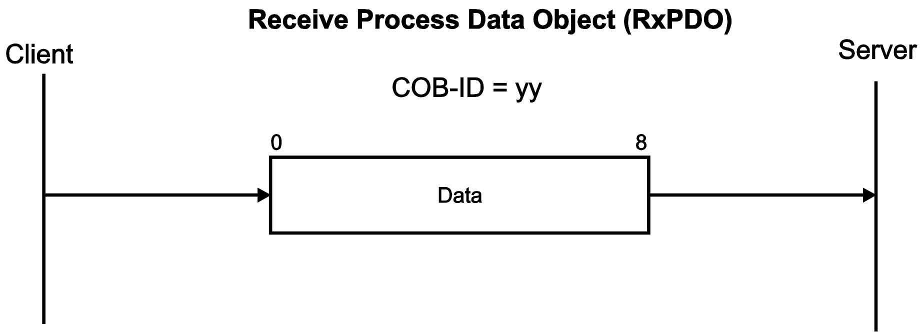


Figure 51: Rx-PDO

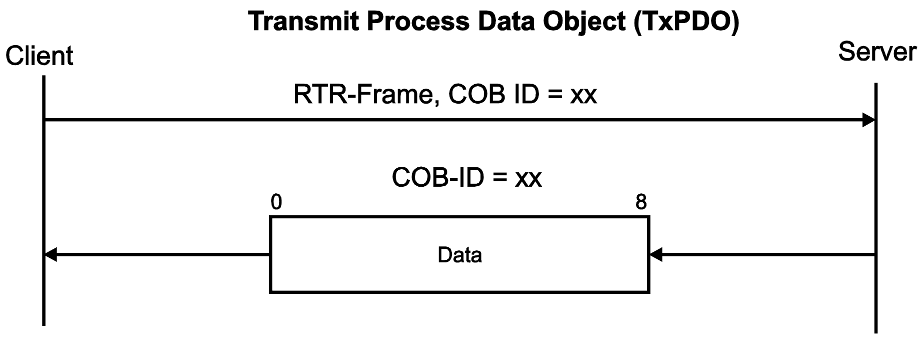


Figure 52: Tx-PDO

10.1.3.1.2 Service Data Object – SDO

Entries in the object directory can be read or written using SDOs. Thus, a CANopen device can be fully configured. The standard SDO is preset a low priority identifier. If the transferred data exceeds 4 bytes, it must be allocated across several telegrams.

10.1.3.1.2.1 SDO Protocol

Protocol overhead is required for the transfer. The protocol overhead must contain the command specifier, index and sub-index of the entry to be read/written.

10.1.3.1.2.1.1 General Structure

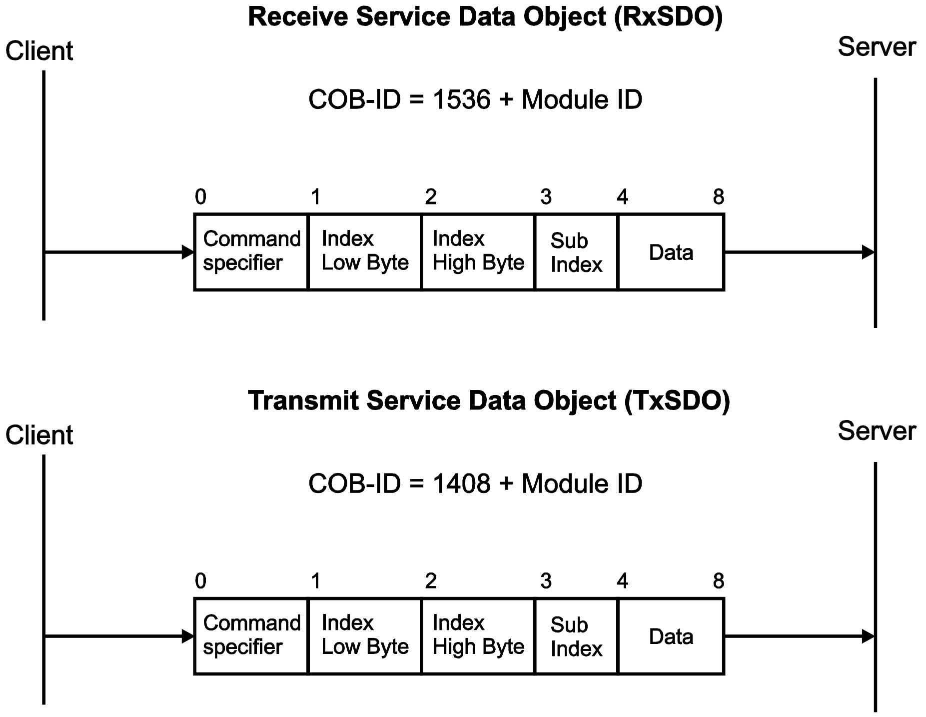


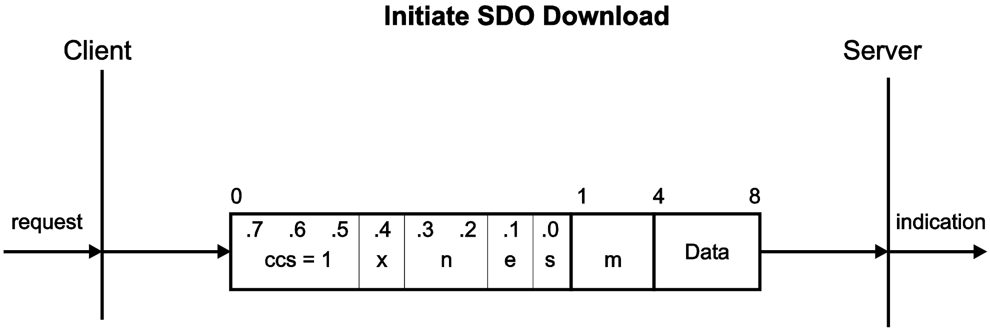
Figure 53: SDO Protocol

10.1.3.1.2.1.2 Download SDO Protocol

This protocol is used to write data from the master to the fieldbus coupler/controller.

10.1.3.1.2.1.2.1 Initiate SDO Download

This protocol is used to initiate data transfer from the the master to the fieldbus boupler/controller. If the transfer contains a maximum of 4 bytes of data, this data is included in the transfer inside the protocol.



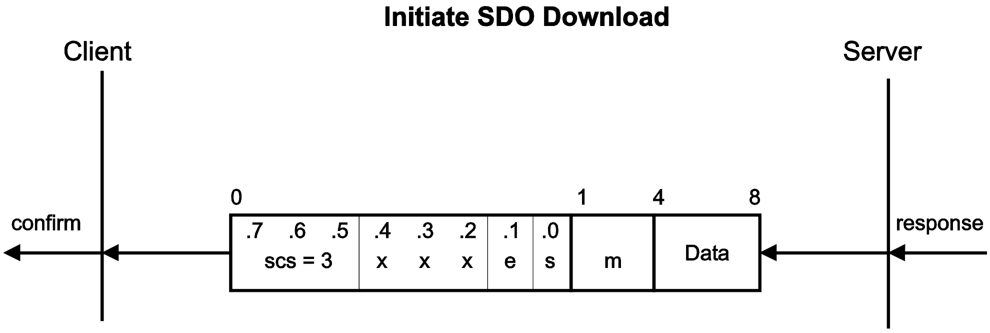


Figure 54: Initiate SDO

Table 64: Initiate SDO

Abbreviation	Data transfer	Description
ccs:	Client command specifier	1: initiate download request
scs:	Server command specifier	3: initiate download response
n:	Only valid when e = 1 and s = 1, otherwise 0.	When n is valid, it indicates the number of bytes that contain no data. Example: 3 data bytes, e = 1 and s = 1, n = 4 - 3 = 1
e:	transfer type	0: normal transfer, number of bytes to be written >= 5 bytes
		1: expedited transfer, number of bytes to be written < 5 bytes
s:	size indicator	0: data set size is not displayed
		1: data set size is displayed s is always 1
m:	multiplexor	Index and sub-index of object directory:
		Index, Low Byte : Byte #1
		Index, High Byte: Byte #2
		Sub-Index: Byte #3
d:	data	e = 0, s = 0: d is reserved for another use by CiA
		e = 0, s = 1: d contains the number of bytes for the download
		Byte 4 contains the LSB; byte 7 contains the MSB.
		e = 1: d contains the data
X:		not used, always 0
reserved:		Reserved for another use by CiA

10.1.3.1.2.1.3 Download SDO Segment

This protocol is used when more than 4 bytes of data are transferred. Data transfer begins after "Initiate SDO Download Protocol" which initiates the transfer, is processed.

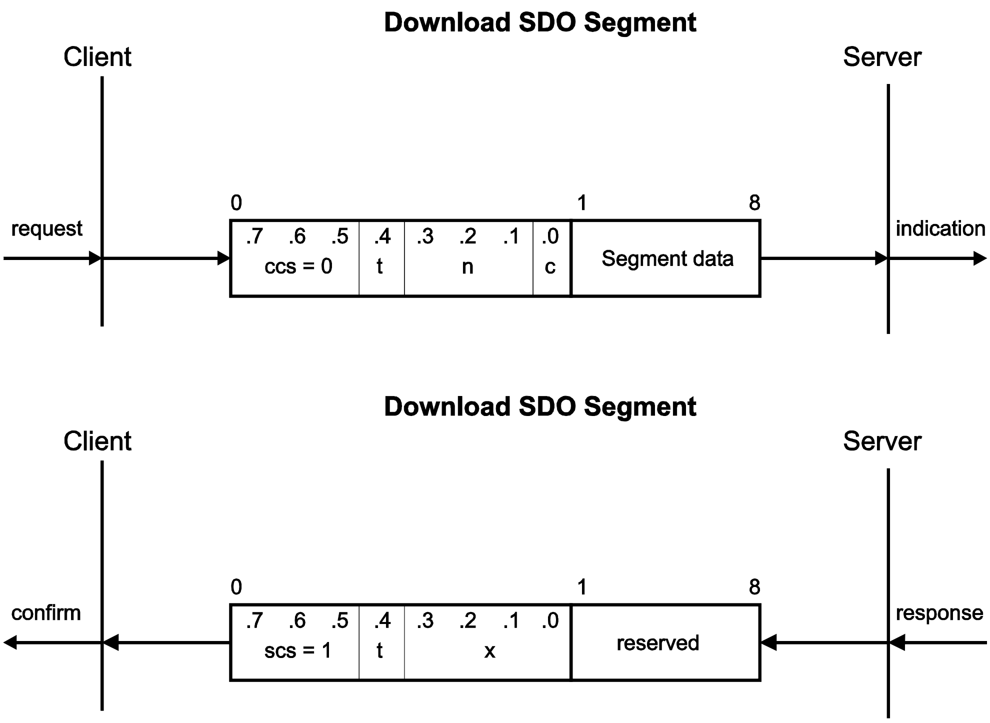


Figure 55: Download SDO Segment

Table 65: Download SDO Segment

Abbreviation	Data transfer	Description
ccs:	Client command specifier	0: download segment request
scs:	Server command specifier	1: download segment response
seg-data	Contains the data to be transferred	The application determines the importance of the data
n:		Indicates the number of bytes that contain no data. No is 0 when no segment size is indicated.
c:	Indicates whether additional data should be downloaded.	0: Data is not yet available for download. 1: No more data is available for download.
t:	Toggle bit	This bit must toggle for each segment for which a download takes place. The first segment sets the toggle bit to 0. The toggle bit is the same for the request and response message.
X:		not used, always 0
reserved:		Reserved for another use by CiA

10.1.3.1.2.1.4 Upload SDO Protocol

This protocol is used to read data from the fieldbus coupler/controller.

10.1.3.1.2.1.4.1 Initiate SDO Upload

This protocol is used to initiate data transfer from the fieldbus coupler/controller to the master. If the transfer contains a maximum of 4 bytes of data, this data is included in the transfer inside the protocol.

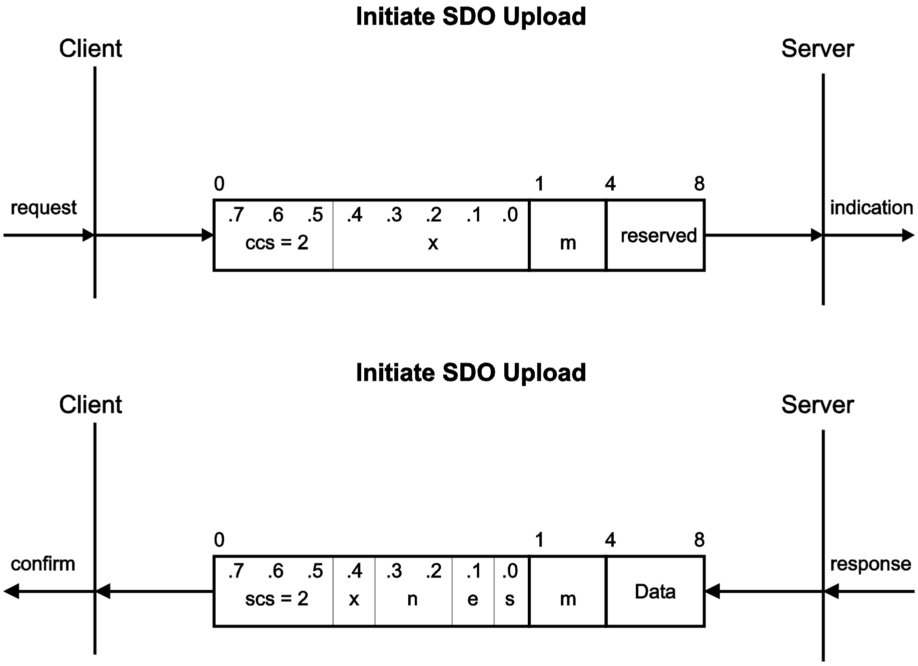


Figure 56: Initiate SDO Upload

Table 66: Initiate SDO Upload

Abbreviation	Data transfer	Description
ccs:	Client command specifier	2: initiate upload request
scs:	Server command specifier	2: initiate upload response
n:	Only valid when e = 1 and s = 1; otherwise 0.	When n is valid, it indicates the number of bytes in d that contain no data. The bytes [8-n, 7] contain no segment data.
e:	transfer type	0: normal transfer, number of bytes to be written >= 5 Byte
		1: expedited transfer, number of bytes to be written < 5 Byte
s:	size indicator	0: The number of bytes that are to be transferred is not indicated
		1: The number of bytes that are to be transferred is indicated (depending on number of bytes)
m:	multiplexor	Index and sub-index of the object directory: Index, Low Byte: Byte #1 Index, High Byte: Byte #2 Sub-index: Byte #3
d:	data	e = 0, s = 0: d is reserved for another use by CiA e = 0, s = 1: d contains the number of bytes available for upload. Byte 4 contains the LSB; byte 7 contains the MSB. e = 1: d contains the data
X:		not used, always 0
reserved:		Reserved for another use by CiA

10.1.3.1.2.1.4.2 Upload SDO Segment

This protocol is used when more than 4 bytes of data are transferred. Data transfer begins after "Initiate Upload Protocol" which initiates the transfer, is processed.

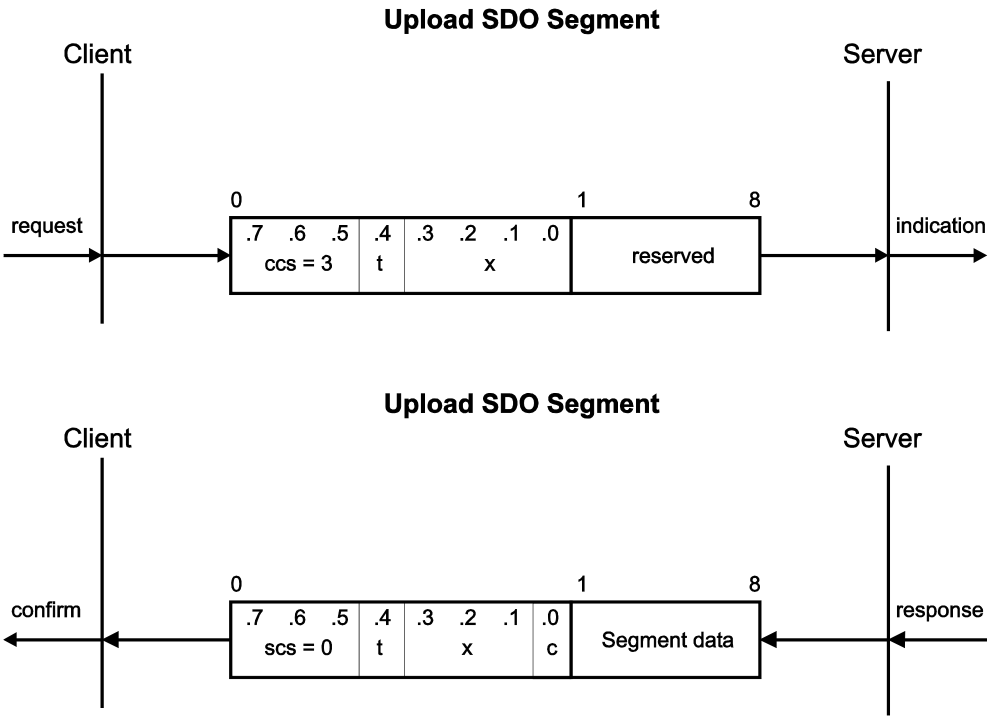


Figure 57: Upload SDO Segment

Table 67: Upload SDO Segment

Abbreviation	Data transfer	Description
ccs:	Client command specifier	3: download segment request
scs:	Server command specifier	0: download segment response
t:	Toggle bit	This bit must change for each segment for which an upload takes place. Foir the first segment, the toggle bit must be set to 0. The toggle bit is the same for the request and response.
c:	Indicates whether additional segments are available for upload	0: Additional segments available for upload 1:No additional segments available for upload
seg-data	Contains the data that is to be transferred.	The application determines the importance of the data.
n:		Indicates the number of bytes that contain no data. Bytes [8-n, 7] contain no data. n is 0 when no segment size is indicated.
X:		not used, always 0
reserved:		Reserved for another use by CiA

10.1.3.1.2.1.5 Abort SDO Transfer

This protocol is used when errors arise during transfer.

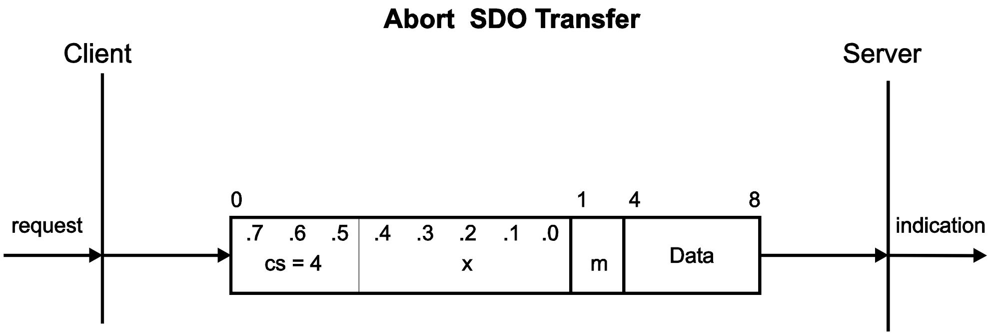


Figure 58: Abort SDO Transfer

Table 68: Abort SDO Transfer

Abbreviation	Data transfer	Description
cs:	command specifier	4: abort domain transfer
m:	multiplexor	Index and sub-index of object directory:
X:		not used, always 0
Data:	4 Byte Error Code	Application-specific data on the transfer abort.

Table 69: Structure of Support Abort Domain Transfer Messages

Byte	Explanation
0	Command specifier; 0x80
1 2	Index
3	SubIdx
4 5	Additional code
6	Error code
7	Error class

The errors listed below are coded as UNSIGNED32 for Additional Code, Error Code and error class:

Table 70: Abort SDO Transfer Error Codes

Byte 4 5 Error Class	Byte 6 Error Code	Byte 7 Additional Code	Designation
05 03	00	00	Toggle bit not alternated
05 04	00	00	SDO protocol timed out
05 04	00	01	Client/server command specifier not valid or unknown
05 04	00	02	Invalid block size (block mode only)
05 04	00	03	Invalid sequence number (block mode only)
05 04	00	04	CRC error (block mode only)
05 04	00	05	Out of memory
06 01	00	00	Unsupported access to an object
06 01	00	01	Attempt to read a write only object
06 01	00	02	Attempt to write a read only object

Table 70: Abort SDO Transfer Error Codes

Byte 4	5	Byte 6	Byte 7	Designation
Error Class		Error Code	Additional Code	
06	02	00	00	Object does not exist in the object dictionary
06	04	00	41	Object cannot be mapped to the PDO
06	04	00	42	The number and length of the objects to be mapped would exceed PDO length
06	06	00	00	Access failed due to an hardware error
06	07	00	10	Data type does not match, length of service parameter does not match
06	07	00	12	Data type does not match, length of service parameter too high
06	07	00	13	Data type does not match, length of service parameter too low
06	09	00	11	Sub-index does not exist
06	09	00	30	Value range of parameter exceeded (only for write access)
06	09	00	31	Value of parameter written too high
06	09	00	32	Value of parameter written too low
06	09	00	36	Maximum value is less than minimum value
08	00	00	00	general error
08	00	00	20	Data cannot be transferred or stored to the application
08	00	00	21	Data cannot be transferred or stored to the application because of local control
08	00	00	22	Data cannot be transferred or stored to the application because of the present device state
08	00	00	23	Object dictionary dynamic generation fails or no object dictionary is present (e.g. object dictionary is generated from file and generation fails because of an file error)

10.1.3.1.2.2 SDO Examples

Four SDO examples are listed below. The data is represented in hexadecimal format. These examples show the operation of SDOs on the CAN telegram level and can prove helpful when the SDO protocol is implemented on a CAN card.

A telegram is divided into 4 columns:

Table 71: Operation of SDOs on the Telegram Level – Telegram Structure

Column	Function	Description
Column 1	Direction	M->BK = Telegram is sent from the master to the fieldbus coupler/controller BK->M = Telegram is sent from the fieldbus coupler/controller to the master
Column 2	CAN identifier	
Column 3	Frame type	D = Data frame R = RTR frame
Column 4	Data	CAN telegram data bytes Up to 8 data bytes can be transferred in a CAN telegram. The individual bytes are separated from each other by blank spaces. Entries with value XX are insignificant, but must be present. The values should be set to 0 for better understanding. Entries in the response from the fieldbus coupler/controller with the value DD contain data that is dependent on the configuration.

10.1.3.1.2.2.1 Example 1: Read Index 0x1000 Sub-Index 0; Device Type

Index 0x1000 returns 4 bytes. Expedited Transfer Mode is used for transfer.

Table 72: Read Index 0x1000 Sub-Index 0; Device Type

Direction	CAN ID	Frame Type	Data Bytes 0-7
M->BK	0x601	D	0x40 00 10 00 XX XX XX XX
BK->M	0x581	D	0x43 00 10 00 91 01 DD 00

Result:

Data bytes 4 and 5: 91 01 sequence low byte, high byte turning:
0x0191 = 401 device profile number

Data bytes 6 and 7: DD 00 sequence low-byte, high-byte turning

10.1.3.1.2.2.2 Example 2: Read Index 0x1008 Sub-Index 0; Manufacturer Device Name

Index 0x1008 returns more than 4 bytes. Normal Transfer Mode is used for transfer. In this case, 2 telegrams are sent per node.

Table 73: Read Index 0x1008 Sub-Index 0; Manufacturer Device Name

Direction	CAN ID	Frame Type	Data Bytes 0-7
M->BK	0x601	D	0x40 08 10 00 XX XX XX XX
BK->M	0x581	D	0x60 XX XX XX XX XX XX XX
M->BK	0x601	D	0x60 XX XX XX XX XX XX XX
BK->M	0x581	D	0x01 37 35 30 2D 33 33 37

Result:

In the first response from the fieldbus coupler/controller, the quantity of data to be transferred is shared with the master (0x00000007 bytes). In the second telegram, the fieldbus coupler/controller returns the item number in ASCII format (hex format).

10.1.3.1.2.2.3 Example 3: Read Index 0x6000 Sub-Index 1; First 8-Bit Digital Input Block

In index 0x6000, the signals of the digital input modules are saved. 8 bits each are assigned to a group and can be read from sub-index 1. In this example, the input value of the first 8-bit group is read via an SDO telegram.

Table 74: Read Index 0x6000 Sub-Index 1; First 8-Bit Digital Input Block

Direction	CAN ID	Frame Type	Data Bytes 0-7
M->BK	0x601	D	0x40 00 60 01 XX XX XX XX
BK->M	0x581	D	0x4F 00 60 01 02 XX XX XX

Result:

The fieldbus coupler/controller returns the status of the first 8-bit group in the 5th byte of the CAN telegram. In this case, bit 2 is set. Bytes 5 - 7 have no significance.

10.1.3.1.2.2.4 Example 4: Write Index 0x6200 Sub-Index 1; First 8-Bit Digital Output Block

In index 0x6200, the output values of the digital output modules are saved. 8 bit each are assigned to a group and can be read and written from sub-index 1. In this example, the value 0xFF is written to the outputs of the first 8-bit digital output group.

Table 75: Write Index 0x6200 Sub-Index 1; First 8-Bit Digital Output Block

Direction	CAN ID	Frame Type	Data Bytes 0-7
M->BK	0x601	D	0x2F 00 62 01 FF XX XX XX
BK->M	0x581	D	0x60 00 62 01 XX XX XX XX

Result:
The outputs of the first 8-bit digital output modules are set.

10.1.3.1.3 Synchronization Object – SYNC

This object enables the synchronization of all network devices. Through the corresponding PDO configuration, you can cause the network device to process its input data or to update its outputs upon arrival of a SYNC object.

Sending the SYNC object in a cyclical manner ensures that all network devices process data simultaneously.

10.1.3.1.3.1 SYNC Protocol

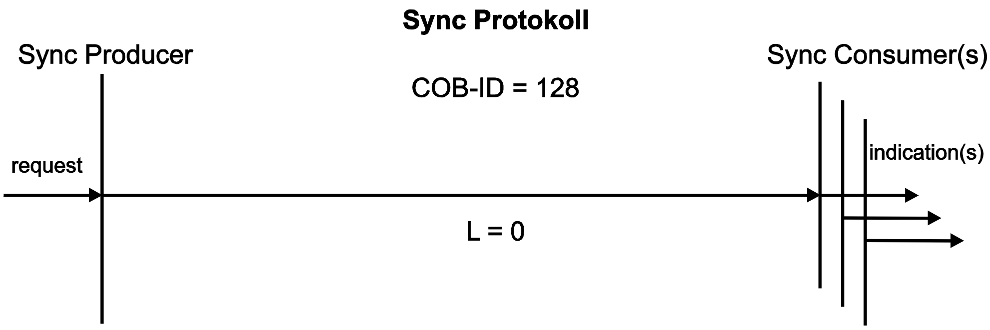


Figure 59: SYNC Protocol

10.1.3.1.4 Emergency Object – EMCY

Emergency objects are triggered by an internal error situation, e.g., an I/O module is pulled during operation or an I/O module reports an error. An emergency object is sent (broadcast) from the fieldbus coupler/controller to all connected devices to notify each device of the error. The bus nodes can then take appropriate error correction measures.

10.1.3.1.4.1 EMCY Protocol

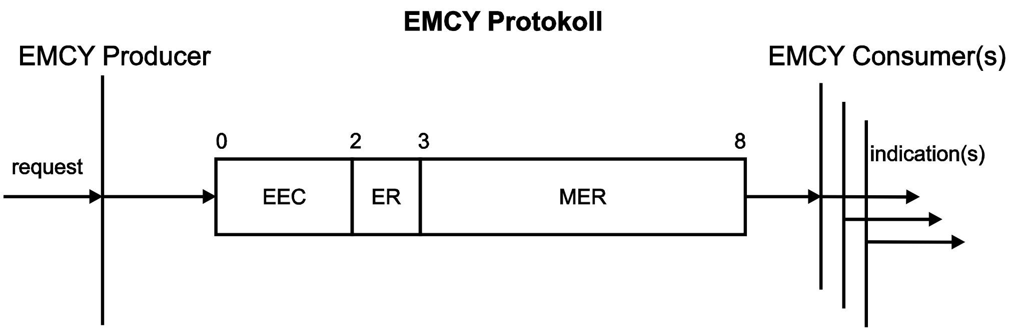


Figure 60: EMCY Protocol

Information

More information about Emergency messages
A detailed description of Emergency messages is available in section “Fieldbus Communication” > … > “Error Messages (Emergency)”.

10.1.3.2 Communication States of a CANopen Fieldbus Coupler/Controller

10.1.3.2.1 CANopen State Diagram

The state diagram described below illustrates the individual communication states and possible transitions relating to CAN communication.

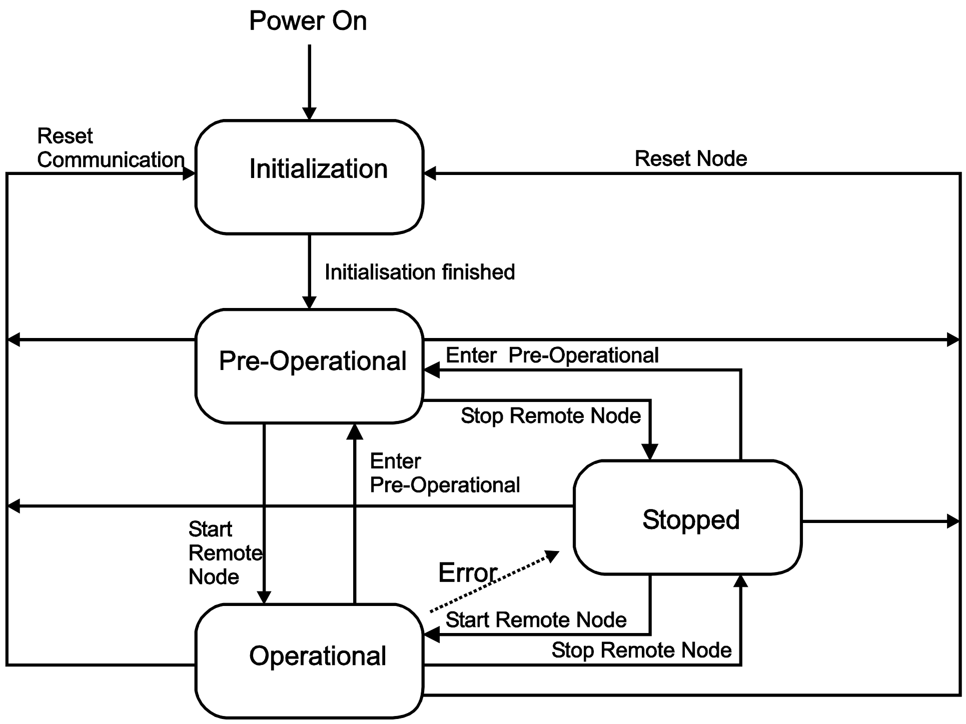


Figure 61: State Diagram of the Fieldbus Coupler/Controller

10.1.3.2.2 INITIALIZATION

After a power-on or restart (module ID does not equal 0), the fieldbus coupler/controller is automatically in the INITIALIZATION state. In this state, the fieldbus coupler/controller checks all functions of its components and communication interface in a self-test. The process image is created and the object directory initialized based on the connected I/O modules and a saved configuration (if applicable). If no errors are detected during the initialization phase, the fieldbus coupler/controller automatically switches to the PRE-OPERATIONAL state. If errors occur, the fieldbus coupler/controller switches to the STOPPED state.

During initialization, the I/O LED flashes orange first and then red with increasing frequency. If initialization and the switch to the PRE-OPERATIONAL state were completed successfully, the I/O LED lights up green and the RUN LED flashes. If errors occur (e.g., no end module), the I/O LED indicates the type of error by a red flashing sequences (see LED status display). In this case, the STOP LED lights up red.

10.1.3.2.3 PRE-OPERATIONAL

In this state, communication can take place via SDOs. Communication via PDOs is not possible. The SDOs can be used to read and write entries in the object direcotyr. In this way, for example, the fieldbus coupler/controller can be reconfigured using a configuration tool. The mapping, fieldbus coupler/controller parameters, IDs, etc. can be adapted to the required conditions. The new configuration can be saved to the flash memory.

The NMT service Start_Remote_Node is used to switch from the PRE-OPERATIONAL to OPERATIONAL state.

In the PRE-OPERATIONAL state, the I/O LED lights up green and the RUN LED flashes.

10.1.3.2.4 OPERATIONAL

In this state, communication can take place via SDOs and PDOs. Different configurations are not possible in this state. For example, changing the COB-ID of a valid PDO is not allowed. A detailed description can be found in the corresponding entries in the object directory.

The NMT service Enter_Pre_Operational_State is used to switch from the OPERATIONAL to PRE-OPERATIONAL state.

In the OPERATIONAL state, the I/O LED and RUN LED light up.

10.1.3.2.5 STOPPED

The STOPPED state reflects a fault condition. It is reached when NMT service Stop_Remote_Node has been received or when a serious internal error has occurred (e.g., I/O module was pulled during operation).

In this state, communication via SDOs or PDOs is not possible. Only the NMT services and Node-Guarding/Heartbeat (if enabled) are executed.

The STOPPED state can be exited via the NMT services Start_Remote_Node_Indication, Enter_Pre_Operational_State and Reset_Node.

In the STOPPED state, the STOP LED lights up.

10.1.3.3 Network Management Objects

10.1.3.3.1 Module Control Protocols

The NMT master can use these protocols to control the state of the NMT slave. The INITIALIZATION, PRE-OPERATIONAL, OPERATIONAL and STOPPED states are defined. It is possible to change the state of all nodes or each node individually to another state.

10.1.3.3.1.1 Start Remote Node

This service is used to change the NMT slave (fieldbus coupler/controller) to the OPERATIONAL state.

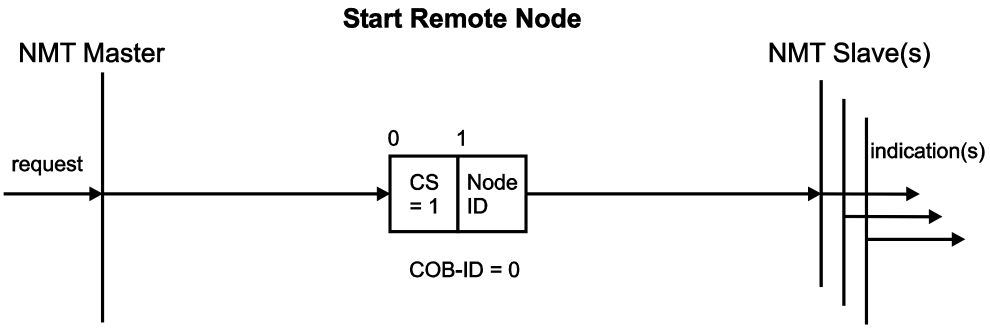


Figure 62: Start Remote Node

Node ID = 0: All available nodes are switched to the OPERATIONAL state.

10.1.3.3.1.2 Stop Remote Node

This service is used to change the NMT slave (fieldbus coupler/controller) to the STOPPED state.

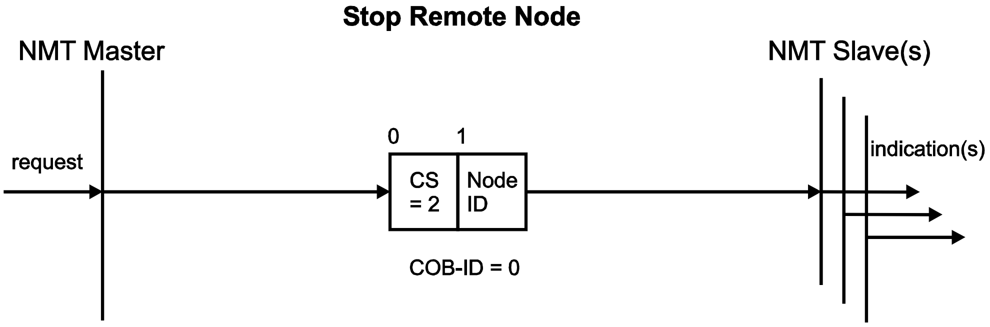


Figure 63: Stop Remote Node

Node ID = 0: All available nodes are switched to the STOPPED state.

10.1.3.3.1.3 Enter Pre-Operational

This service is used to change the NMT slave (fieldbus coupler/controller) to the PRE-OPERATIONAL state.

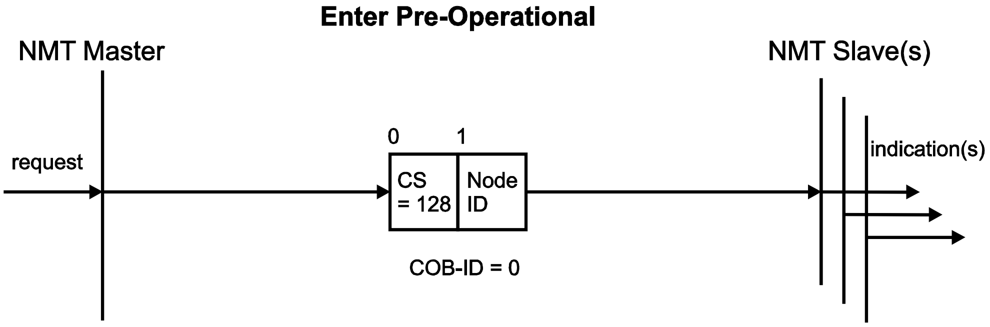


Figure 64: Enter PRE-OPERATIONAL

Node ID = 0: All available nodes are switched to the PRE-OPERATIONAL state.

10.1.3.3.1.4 Reset Node

This service is used to reset the NMT slave (fieldbus coupler/controller).

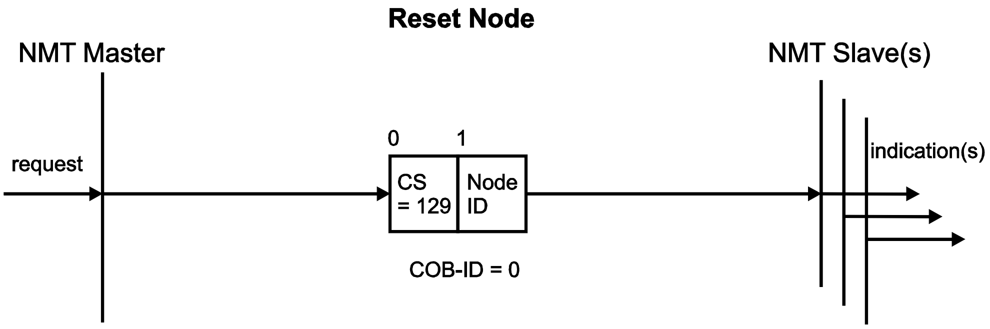


Figure 65: Reset Node

Node ID = 0: All available nodes are reset.

10.1.3.3.2 Error Control Protocols

These protocols can be used to identify errors in the network. In this way, a master can check if a node is still in the designated state or if it has been switched to a different state (e.g., by a reset).

10.1.3.3.3 Node Guarding Protocol

With Node Guarding, the NMT slave is prompted cyclically via an RTR frame to send its current state. An additional bit switch determines whether the NMT slave is still functioning correctly.

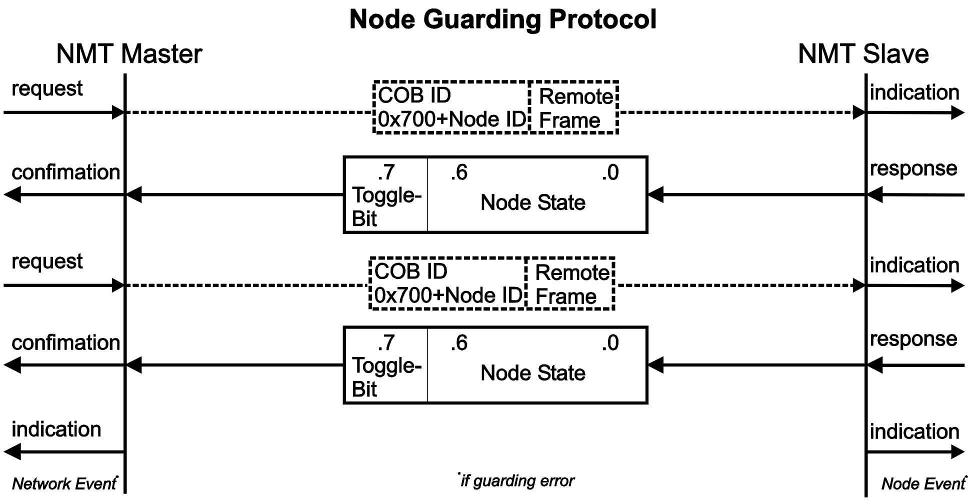


Figure 66: Node Guarding Protocol

10.1.3.3.4 Heartbeat Protocol

This protocol enables monitoring without RTR frames. A heartbeat generator generates a hearbeat message cyclically and n nodes receive it. In the heart message, the current state of the generator is encoded.

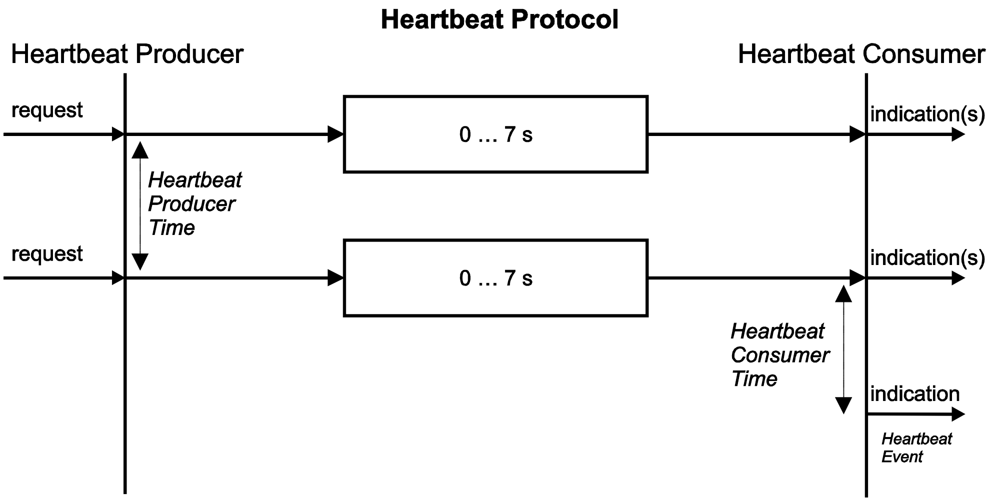


Figure 67: Heartbeat Protocol

10.1.3.3.5 Boot-up Protocol

This protocol indicates that the NMT slave has switched from the INITIALIZATION state to the PRE-OPERATIONAL state. It is executed following a hardware/software reset or following the “Reset Node” service.

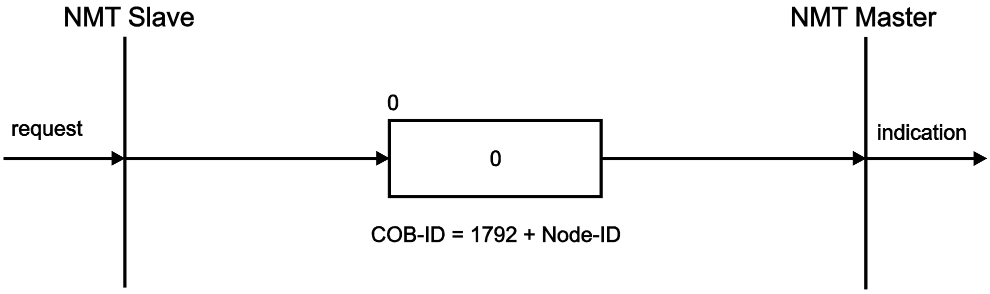


Figure 68: Bootup Protocol

10.1.3.4 Object Directory

The object directory is the central location of a CANopen device where all configuration information is stored and accessed. The directory is organized in a table and includes three different types of CANopen objects:

- Communication Profile Area (Index 0x1000 – 0x1FFF)
  This area includes all parameters relevant for CANopen communication. This area is the same for all CANopen devices.

- Manufacturer-Specific Profile Area (Index 0x2000 – 0x5FFF)
  In the area, manufacturers cna implement their own company-specific objects.

- Standardized Device Profile Area (Index 0x6000 – 0x9FFF)
  This area includes all objects that are supported by a specific device profile. The fieldbus coupler/controller supports device profile DS-401 (Device Profile for Generic I/O Modules).

A logical addressing structure in the object directory is used to access communication and device parameters as well as data and functions. Every entry in the directory is identified by a 16-bit index. This index indicates the row address in the table. The maximum number of entries is 65536.

If an object is composed of several components, the components are identified by an 8-bit sub-index. The sub-index indicates the column address in the table, the maximum number of entries being 256.

Every entry consists of:

- An object name that describes the function of the object
- A data type attribute that determines the data type of the entry
- An access attribute that indicates whether the entry can be read, written or both

The sub index 0 indicates the maximum number of the following subiindexes. The data is coded in the subsequent sub-indices.

Table 76: Structure of CANopen Object Directory

Index (hexadecimal)	**Object**
0x0000	Not used
0x0001 – 0x001F	Static data types
0x0020 – 0x003F	Complex data types
0x0040 – 0x005F	Manufacturer-specific data types
0x0060 – 0x007F	Profile-specific static data types
0x0080 – 0x009F	Profile-specific complex data types
0x00A0 – 0x0FFF	Reserved
0x1000 – 0x1FFF	Communication profile (DS 301)
0x2000 – 0x5FFF	Manufacturer-specific parameters
0x6000 – 0x9FFF	Parameters from the standardized device profiles
0xA000 – 0xFFFF	Reserved

The size of the object directory can cope with the worst case. Object entries that are not used due to the current I/O module configuration are disabled.

10.1.3.4.1 Initialization

After power-on, the connected I/O module configuration is determined.

If a customized configuration was stored and the current I/O module configuration matches this configuration, then it is used to initialize the object directory.

Otherwise, the object directory is initialized with a default configuration.

10.1.3.4.1.1 Default Configuration

10.1.3.4.1.1.1 Initialization of the Communication Profile Area

All objects of this profile supported by the fieldbus coupler/controller are initialized according to the DS 301 default values (CANopen Application Layer and Communication Profile).

Entering the default mapping parameters:

The mapping parameters are preset based on the device profile used. The fieldbus coupler/controller supports the DS 401 profile and the procedure described there is used.

The first 4 Rx-/Tx-PDOs are defined as the default PDOs. If there are more inputs/outputs on the fieldbus coupler/controller than can be covered with the default PDOs, all remaining inputs/outputs are entered from Rx/Tx-PDO 5. All digital inputs/outputs are entered first and then all analog inputs/outputs. Even if no analog I/O modules are available, it only continues with PDO 5 with more than 64 digital I/Os per input/output. PDO 2 to 4 then remain unused. In addition, a PDO is only preset with one data type, i.e., if there is a 3-byte and 4-byte I/O module, 2 PDOs are occupied with one entry.

1st Rx-PDO:

The first Rx-PDO contains the first 8x8 digital outputs at the most. If there are no digital outputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 77: 1. Rx-PDO:

Idx	S-Idx	Comment	Default Value
0x1600	0	Number of mapped objects	None, possible values: 0: No digital output block 1..8: 1..8 digital output blocks
	1	1st mapped digital output block	0x6200 01 08
	2	2nd mapped digital output block	0x6200 02 08
	:		
	8	8th mapped digital output block	0x6200 08 08

2nd Rx-PDO:

The second Rx-PDO contains the 16-bit analog outputs 1 - 4 at the most. If there are no 16-bit analog outputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 78: 2nd Rx-PDO

Idx	S-Idx	Comment	Default Value
0x1601	0	Number of mapped objects	None, possible values: 0: No analog output 1..4: 1 … 4 analog output
	1	1. Mapped 16-bit analog output	0x6411 01 10
	2	2nd mapped 16-bit analog output	0x6411 02 10
	3	3rd mapped 16-bit analog output	0x6411 03 10
	4	4th mapped 16-bit analog output	0x6411 04 10

3rd Rx-PDO:

The third Rx-PDO contains 16-bit analog outputs 5 - 8 at the most. If there are more than 4 16-bit analog outputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 79: 3rd Rx-PDO:

Idx	S-Idx	Comment	Default Value
0x1602	0	Number of mapped objects	None, possible values: 0: No analog output 1..4: 1 … 4 analog output
	1	5th mapped 16-bit analog output	0x6411 05 10
	2	6th mapped 16-bit analog output	0x6411 06 10
	3	7th mapped 16-bit analog output	0x6411 07 10
	4	8th mapped 16-bit analog output	0x6411 08 10

4th Rx-PDO:

The fourth Rx-PDO contains 16-bit analog outputs 9 - 12 at the most. If there are more than 8 16-bit analog outputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 80: 4th Rx-PDO

Idx	S-Idx	Comment	Default Value
0x1603	0	Number of mapped objects	No, possible values: 0: No analog output 1..4: 1 …4 analog outputs
	1	9th mapped 16-bit analog output	0x6411 09 10
	2	10th mapped 16-bit analog output	0x6411 0A 10
	3	11th mapped 16-bit analog output	0x6411 0B 10
	4	12th mapped 16-bit analog output	0x6411 0C 10

1st Tx-PDO:

The first Tx-PDO contains the first 8x8 digital inputs at the most. If there are no digital inputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 81: 1st Tx-PDO:

Idx	S-Idx	Comment	Default Value
0x1A00	0	Number of mapped objects	None, possible values: 0: No digital input block 1..8: 1..8 digital input blocks
	1	1st mapped digital input block	0x6000 01 08
	2	2nd mapped digital input block	0x6000 02 08
	:	:	:
	8	8th mapped digital input block	0x6000 08 08

2nd Tx-PDO:

The second Tx-PDO contains 16-bit analog input 1 - 4 at the most. If there are no 16-bit analog inputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 82: 2nd Tx-PDO

Idx	S-Idx	Comment	Default Value
0x1A01	0	Number of mapped objects	None, possible values: 0: No analog input 1..4: 1..4 analog inputs
	1	1st mapped 16-bit analog input	0x6401 01 10
	2	2nd mapped 16-bit analog input	0x6401 02 10
	3	3rd mapped 16-bit analog input	0x6401 03 10
	4	4th mapped 16-bit analog input	0x6401 04 10

3rd Tx-PDO:

The third Tx-PDO contains 16-bit analog inputs 5 - 8 at the most. If there are more than 4 16-bit analog inputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 83: 3rd Tx-PDO

Idx	S-Idx	Comment	Default value
0x1A02	0	Number of mapped objects	None, possible values: 0: No analog input 1..4: 1..4 analog inputs
	1	5th mapped 16-bit analog input	0x6401 05 10
	2	6th mapped 16-bit analog input	0x6401 06 10
	3	7th mapped 16-bit analog input	0x6401 07 10
	4	8th mapped 16-bit analog input	0x6401 08 10

4th Tx-PDO:

The fourth Tx-PDO contains 16-bit analog inputs 9 - 12 at the most. If there are more than 4 16-bit analog inputs, sub-index 0 has the value 0 and this PDO is not used by default.

Table 84: 4th Tx-PDO

Idx	S-Idx	Comment	Default Value
0x1A03	0	Number of mapped objects	None, possible values: 0: No analog input 1..4: 1..4 analog inputs
	1	9th mapped 16-bit analog input	0x6401 09 10
	2	10th mapped 16-bit analog input	0x6401 0A 10
	3	11th mapped 16-bit analog input	0x6401 0B 10
	4	12th mapped 16-bit analog input	0x6401 0C 10

Initialization of Manufacturer-Specific Profile Area
This area is initialized as described in the object directory.

Initialization of Device Profile Area
All supported objects are initialized as defined in the standard DS 401.

10.1.3.4.2 Communication Profile Area

The following table lists all the communication profile objects supported by the fieldbus coupler/controller.

Idx	Name	Type	Explanation
0x1000	Device Type	Unsigned32	Device profile
0x1001	Error register	Unsigned8	Internal error
0x1002	Manufacturer Status Register	Unsigned32	Status Register (relevant for manufacturers)
0x1003	Predefined error field	Array Unsigned 32	Save the last 20 errors that have occurred
0x1005	COB-ID SYNC message	Unsigned32	COB-ID for the synchronization object
0x1006	Communication cycle period	Unsigned32	Max. time between 2 SYNC messages
0x1008	Manufacturer device name	Visible string	Device name
0x1009	Manufacturer hardware version	Visible string	Hardware version
0x100A	Manufacturer software version	Visible string	Software version
0x100C	Guard time	Unsigned16	Monitoring time for the “Life Guarding Protocol”
0x100D	Lifetime factor	Unsigned8	Lifetime factor
0x1010	Store parameters	Array Unsigned 32	Parameters to store the configuration
0x1011	Restore default parameters	Array Unsigned 32	Parameter to restore the default configuration
0x1014	COB-ID emergency object	Unsigned32	COB-ID for the emergency object
0x1015	Inhibit time EMCY	Unsigned32	Min. time between two EMCY messages
0x1016	Consumer heartbeat time	Array Unsigned 32	Heartbeat monitoring time
0x1017	Producer heartbeat time	Unsigned16	Time between two generated heartbeat messages
0x1018	Identity object	Record identity	Device information
0x1200 - 0x1201	Server SDO parameter	Record SDO parameter	Parameter for server SDO
0x1400 - 0x141F	Receive PDO communication parameter	Record PDO parameter	Communication parameter for receive PDO
0x1600 - 0x161F	Receive PDO mapping parameter	Record PDO mapping	Mapping parameter for the receive PDO
0x1800 - 0x181F	Transmit PDO communication parameter	Record PDO parameter	Communication parameter for the transmit PDO
0x1A00 - 0x1A1F	Transmit PDO mapping parameter	Record PDO mapping	Mapping parameter for the transmit PDO

10.1.3.4.2.1 Object 0x1000 – Device Type

Table 85: Object 0x1000 – Device Type

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1000	0	Device Type	Unsigned32	RO	-

This object indicates the implemented device profile. The CANopen fieldbus coupler/controller implemented the “Device Profile for Generic I/O Modules” (device profile no. 401). In addition, the value in index 0x1000 provides information about what type of I/O modules are plugged in.

Table 86: Object 0x1000 – Structure, Device Type

MSB			**LSB**
0000.0000	0000.4321	Device Profile Number 0x01 (High Byte)	Device Profile Number 0x91 (Low Byte)

With bit 1 = 1, if at least on digital input is present.

2 = 1, if at least one digital output is present.

3 = 1, if at least one analog input is present.

4 = 1, if at least one analog output is present.

10.1.3.4.2.2 Object 0x1001 – Error Register

Table 87: Object 0x1001 – Error Register

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1001	0	Error register	Unsigned8	RO	-

Internal errors are represented in this register. This register is also part of the emergency message.

Table 88: Object 0x1001 – Structure, Error Register

Bit	Explanation
0	General error
1	Electricity
2	Voltage
3	Temperature
4	Communication
5	Device profile specific
6	Reserved
7	Manufacturer-specific

If an error has arisen, bit 0 is always set. Additional bits that are set specify the error more precisely.

10.1.3.4.2.3 Object 0x1002 – Manufacturer Status Register

Table 89: Object 0x1002 – Manufacturer Status Register

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1002	0	Manufacurer Status Register	Unsigned32	RO	-

Note

Control of object 0x1002 by object 5200 sub-index 6-8!
Object 0x1002 can be activated and its content set by object 5200 sub-index 6-8 (see section “Fieldbus Communication”> … > “Object 0x5200 – Fieldbus Coupler/Controller Configuration”).

Object 0x1002 is deactivated by default!

10.1.3.4.2.4 Object 0x1003 – Predefined Error Field

Table 90: Object 0x1003 – Predefined Error Field

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1003	0	Number of errors	Unsigned8	RW	0
	1	Standard error field	Unsigned32	RO	-
	:	:	:	:	:
	20	Standard error field	Unsigned32	RO	-

Sub-index 0 contains the errors currently stored in the field. If a new error occurs, it is added to sub-index 1 and all existing errors are moved down one sub-index. A maximum of 20 error entries is supported. If more than 20 errors occur, the error at sub-index 20 is overwritten.

Standard error field structure:

Table 91: Object 0x1003 – Structure – Predefined Error Field

Bit31	Bit16	Bit15	Bit0
Additional Information		Error code	

The “Additional Information” corresponds to the first 2 bytes of the “Additional Code” from the Emergency telegram. The Error Code is the same as the Error Code in the Emergency telegram.
By writing a 0 into sub-index 0, the complete error memory is deleted.

10.1.3.4.2.5 Object 0x1005 – COB-ID SYNC Message

Table 92: Object 0x1005 – COB-ID SYNC Message

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1005	0	COB-ID SYNC	Unsigned32	RW	0x00000080

Table 93: Object 0x1005 – Structure, COB-ID SYNC message

Bit31	Bit11	Bit10	Bit0
Reserved (always 0)		COB-ID	

10.1.3.4.2.6 Object 0x1006 – Communication Cycle Period

Table 94: Object 0x1006 – Communication Cycle Period

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1006	0	Communication cycle period	Unsigned32	RW	0

The object defines the maximum time in µs for two consecutive SYNC messages. Internal resolution is 2ms. If this value is 0, SYNC monitoring does not take place.

10.1.3.4.2.7 Object 0x1008 – Manufacturer Device Name

Table 95: Object 0x1008 – Manufacturer Device Name

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1008	0	Manufacturer Device Name	Visible string	RO	Fieldbus coupler/controller item number

The object specifies the device name of the fieldbus coupler/controller.

10.1.3.4.2.8 Object 0x1009 – Manufacturer Hardware Version

Table 96: Object 0x1009 – Manufacturer Hardware Version

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1009	0	Manufacturer hardware version	Visible string	RO	Current hardware version

The object specifies the current hardware version of the fieldbus coupler/controller.

10.1.3.4.2.9 Object 0x100A – Manufacturer Software Version

Table 97: Object 0x100A – Manufacturer Software Version

Idx	S-Idx	Name	Type	Attribute	Default Value
0x100A	0	Manufacturer software version	Visible string	RO	Current software version

The object specifies the current software version of the fieldbus coupler/controller.

10.1.3.4.2.10 Object 0x100C – Guard Time

Table 98: Object 0x100C – Guard Time

Idx	S-Idx	Name	Type	Attribute	Default Value
0x100C	0	Guard time	Unsigned16	RW	0

This object specifies the *Guard Time* in milliseconds. An NMT master requests the state of the NMT slave in a cyclical manner. The time between two requests is the *Guard Time*.

10.1.3.4.2.11 Object 0x100D – Life Time Factor

Table 99: Object 0x100D – Life Time Factor

Idx	S-Idx	Name	Type	Attribute	Default Value
0x100D	0	Life time factor	Unsigned8	RW	0

The *Life Time Factor* is part of the *Node Guarding Protocol*. The NMT slave checks whether it was queried within the *Node Life Time* (guard time multiplied by the *Life Time Factor*). If not, the slave must assume that the NMT master is no longer in normal operation. It then initiates a *Life Guarding Event*.
If the *Node Life Time* is zero, there is no monitoring.

10.1.3.4.2.12 Object 0x1010 – Store Parameters

Table 100: Object 0x1010 – Store Parameters

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1010	0	Max. supported sub-index	Unsigned8	RO	1
	1	Save all parameters	Unsigned32	RW	1

This object can be used to permanently store the user's settings. To do so, the signature “save” (lower case letters ASCII - MSB – 0x65 76 61 73 - LSB) must be written to index 0x1010 sub-index 1. The save operation is carried out in the background. It takes approx. 2 - 3 seconds. When done, the SDO reply telegram is sent. During the save operation, SDO communication can continue. An error message only appears if an attempt to save is made when the previous command is not yet completed. Likewise, it is not possible to save when “Restore” is still active.
Once a setting has been saved, the Emergency “Changed HW configuration” is no longer sent when rebooting the fieldbus coupler/controller with changing the module configuration.

⚠ WARNING

Attention!

If only the module ID is changed via the DIP switch after saving the configuration, the saved configuration is still used. In other words, all module ID-specific entries in the object directory (objects that are module ID dependent and that have the “rw” attribute) report with the old values (e.g., Emergency ID, etc.).

10.1.3.4.2.13 Object 0x1011 – Restore Default Parameters

Table 101: Object 0x1011 – Restore Default Parameters

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1011	0	Max. supported sub-index	Unsigned8	RO	4
	1	Set all parameters to default values	Unsigned32	RW	1
	2	-	Unsigned32	RW	0
	3	-	Unsigned32	RW	0
	4	Set all parameters to default values once	Unsigned32	RW	1

This object can be used to reset the parameters stored by the user to the default settings.
Sub-indices 2 and 3 are not supported.
The Load command is processed in the background. It takes approx. 2 - 3 seconds. When done, the SDO reply telegram is sent. During the Load operation, SDO communication can continue. An error message only appears if an attempt is made to process a load command again when the previous command is not yet completed. Likewise, it is not possible to trigger a load command when "Save" is still active.

10.1.3.4.2.13.1 Sub-Index 1 – Permanent Entry Default Parameters

By writing the "load" signature (lower-case letters ASCII - MSB 0x64 0x61 0x6F 0x6C LSB) to index 0x1011, sub-index 1, the factory default settings are loaded after the next and all subsequent "Power ON" operations (up to the next SAVE command).

10.1.3.4.2.13.2 Sub-Index 4 – One-Time Entry of Default Paramters

By writing the "load" signature (lower-case letters ASCII - MSB 0x64 0x61 0x6F 0x6C LSB) to index 0x1011, sub-index 4, the factory default settings are loaded once after the subsequent "Power ON". The saved configuration is loaded after each subsequent "Power ON". This can be used, for example, in the development phase to compare behaviors between the saved and default configuration without having to reset or save all parameters each time.

Procedure:

Replace the saved configuration by the default configuration once.

-> load (Index 0x1011, Sub-Index 4)
-> Reset
-> Default Values
    (In this state, the Load command (Index 0x1011, Sub-Index 4) cannot be used again!)
-> Reset
-> Saved Configuration

10.1.3.4.2.14 Object 0x1014 – COB-ID Emergency Object

Table 102: Object 0x1014 – COB-ID Emergency Object

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1014	0	COB-ID EMCY	Unsigned32	RW	0x80+ModuleID

The object defines the COB-ID for the EMCY message.

Table 103: Object 0x1014 – Structure, COB-ID Emergency Object

Bit31	Bit 30	Bit11	Bit10	Bit0
0/1 valid/invalid	Reserved (always 0)		COB-ID	

To enter a new COB-ID, bit 31 must first be set to 1 because a valide COB-ID (Bit31=0) cannot be changed according to standard DS301.

10.1.3.4.2.15 Object 0x1015 – Inhibit Time Emergency Object

Table 104: Object 0x1015 – Inhibit Time Emergency Object

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1015	0	Inhibit time EMCY	Unsigned16	RW	0

This object specifies the minimum time that must elapse before another emergency is sent.

An entry equal to zero deactivates delayed sending.

Since with delayed transmission the entries are entered in a queue, the maximum number of consecutive fast emergencies is limited to the queue size (20 entries). If this number is exceeded, an emergency is sent immediately that reports the overflow.

One unit of time is 100 µs.

Example: Minimum time interval between two EMCY 30 ms
Index 0x1015 = 300 = 0x12C

10.1.3.4.2.16 Object 0x1016 – Consumer Heartbeat Time

Table 105: Object 0x1016 – Consumer Heartbeat Time

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1016	0	Max. monitorable modules	Unsigned8	RO	5
	1	1st heartbeat time entry	Unsigned32	RW	0
	2	2nd heartbeat time entry	Unsigned32	RW	0
	3	3rd heartbeat time entry	Unsigned32	RW	0
	4	4th heartbeat time entry	Unsigned32	RW	0
	5	5th heartbeat time entry	Unsigned32	RW	0

This entry enables monitoring of up to 5 modules. A check is made to determine whether each module defined in this object has generated a *heartbeat* within the set time. If the set time has been exceeded, a *heartbeat event* is triggered. The heartbeat time is entered in milliseconds. If the time is 0, monitoring is deactivated.

Table 106: Object 0x1016 – Structure, Consumer Heartbeat Time

	MSB		LSB
Bit	31-24	23-16	15-0
Value	Reserved	Module ID	Heartbeat time
Data Type	-	Unsigned8	Unsigned16

10.1.3.4.2.17 Object 0x1017 – Producer Heartbeat Time

Table 107: Object 0x1017 – Producer Heartbeat Time

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1017	0	Producer heartbeat time	Unsigned16	RW	0

This object defines the time (in milliseconds) between two transmitted heartbeats. No heartbeat is sent if the time is set to 0. As soon as a value other than 0 is entered, the heartbeat transfer begins.

10.1.3.4.2.18 Object 0x1018 – Identity Object

Table 108: Object 0x1018 – Identity Object

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1018	0	Max. supported entries	Unsigned8	RO	4
	1	Manufacturer ID	Unsigned32	RO	33
	2	Device Description	Unsigned32	RO	3xx / 8xx
	3	Revision number	Unsigned32	RO	akt. Rev.-Nr.
	4	Serial number	Unsigned32	RO	Curr. serial No.

This object specifies the device being used.

The manufacturer ID contains a unique number for each manufacturer. WAGO has been assigned an ID of 33.
The device description reflects the family of products concerned.
The revision No. includes a specific CANopen behavior. The *Major Rev. No.* is increased when the functionality changes. You can use the *Minor Rev. No.* to distinguish between different versions with the same CANopen behavior.

Table 109: Structure of the Rev. No.

Bit31	Bit16	Bit15	Bit0
Major Rev. No.		Minor Rev. No.	

The serial number is a number unique to this device family.

10.1.3.4.2.19 Object 0x1200 … 0x1201 – Server SDO

Table 110: Object 0x1200 … 0x1201, Server SDO

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1200 to 0x1201	0	Max. supported entries	Unsigned8	RO	2
	1	COB-ID Client->Server (Rx)	Unsigned32	Idx 0x1200 RO Idx 0x1201 RW	Idx 0x1200 0x600+Module ID Idx 0x1201 0x80000000
	2	COB-ID Server->Client (Tx)	Unsigned32	Idx 0x1200 RO Idx 0x1201 RW	Idx 0x1200 0x580+Module ID Idx 0x1201 0x80000000

This object is used to access the entries in the object directory.
The second SDO is disabled by default. It is not permitted to change the COB-IDs of the second SDO if they are active (bit 31 = 0).

Table 111: Structure of the COB-ID – Server SDO

Bit31	Bit 30	Bit11	Bit10	Bit0
0/1 valid/invalid		Reserved (always 0)		COB-ID

10.1.3.4.2.20 Object 0x1400 … 0x141F – Receive PDO Communication Parameter

Table 112: Object 0x1400 … 0x141F – Receive PDO Communication Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1400 through 0x141F	0	Max. supported entries	Unsigned8	RO	2
	1	COB-ID	Unsigned32	RW	Idx 0x1400 0x200+Module ID Idx 0x1401 0x300+Module ID Idx 0x1402 0x400+Module ID Idx 0x1403 0x500+Module ID Idx 0x1404-141F 0x80000000
	2	Transfer type	Unsigned8	RW	255

This object is used to set the communication parameters of the Rx-PDOs. 32 Rx-PDOs are supported. The default COB-IDs from the first four PDOs are preassigned per DS301. All other PDOs are disabled. If not all default PDOs are used (e.g., fewer I/O modules are connected), the unused default PDOs are also disabled.

Table 113: Structure of the COB-ID – Receive PDO Communication Parameter

Bit31	Bit 30		Bit11	Bit10	Bit0
0/1 valid/invalid		Reserved (always 0)		COB-ID	

If a new COB-ID is entered, bit 31 must first be set to 1 because a valid COB-ID (Bit31=0) cannot be changed according to standard DS301.

One data transfer mode (transmission type in "Index Communication Parameter") can be defined for each PDO. Digital and analog entries are transmitted by default 'Change of Value' (COV).

The following table explains the type of transfer depending on the set "Transmission Type".

Table 114: Mode for Data Transmission

Transfer Type \ PDO Transmission	cyclic	acyclic	Synchronous	Asynchronous	RTR only	Tx-PDO (inputs)	Rx-PDO (outputs)
0		X	X			If COV is transmitted with each SYNC	Set outputs after each SYNC as requested by the last PDO received
1 – 240	X		X			Transmission with each nth SYNC (x = 1 through 240)	Set outputs after each SYNC as required by the last PDO received
241 - 251	- reserved -						
252			X		X	Data is read again with a SYNC, but not sent, Request via RTR	Not supported
253				X	X	Request via RTR	COV
254				X		COV [1)]	COV
255				X		COV [1)]	COV

[1)] The data is transmitted at the interval of the "Inhibit Time" set

10.1.3.4.2.21 Object 0x1600 … 0x161F – Receive PDO Mapping Parameter

Table 115: Object 0x1600 … 0x161F – Receive PDO Mapping Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1600 through 0x161F	0	Number of mapped objects	Unsigned8	RW	-
	1 to 8	1st object through 8th object	Unsigned32	RW	-

This object is used to determine the data that is sent via the PDO.

Sub-index 0 contains the number of objects that are valid for the PDO.

Table 116: Object 0x1600 … 0x161F- Structure, Objects 1 to 8

Bit31	Bit16	Bit 15	Bit8	Bit7	Bit0
Index (Index of the object to be transmitted)		Sub-Index (Sub-index of the object to be transmitted)		Object Length (size of object in bits Because a maximum of 8 bytes can be transmitted in one PDO, the sum of the valid object lengths may not exceed 64 (8 bytes x 8 bits).)	

10.1.3.4.2.22 Object 0x1800 … 0x181F – Transmit PDO Communication Parameter

Table 117: Object 0x1800 … 0x181F – Transmit PDO Communication Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1800 through 0x181F	0	Max. supported entries	Unsigned8	RO	5
	1	COB-ID	Unsigned32	RW	Idx 0x1800 0x180+Module ID Idx 0x1801 0x280+Module ID Idx 0x1802 0x380+Module ID Idx 0x1803 0x480h+Module ID Idx 0x1804-181F 0x80000000
	2	Transfer type	Unsigned8	RW	255
	3	Inhibit Time	Unsigned16	RW	Idx 0x1800 0 Idx 0x1801 – 181F 100
	4	Reserved	Unsigned8	RW	0
	5	Event Timer	Unsigned16	RW	0

This object is used to set the communication parameters of the Tx-PDOs. 32 Tx-PDOs are supported. The default COB-IDs from the first four PDOs are preassigned per DS301. All other PDOs are disabled. If not all default PDOs are used (e.g., fewer I/O modules are connected), the unused default PDOs are also disabled.
"Inhibit Time" specifies the minimum time between two consecuitive PDOs with the same COB-ID. One unit of time is 100°µs. The transmitted value is rounded down internally to the nearest millisecond.
If a new value is entered, the COB-ID must be set invalid (Bit31 = 1) because a new time cannot be entered when the COB-ID is valid (Bit31 = 0) according to standard DS301.

Example: The minimum interval between two PDOs with the same COB-ID should be 30 ms:
Sub-Index 3 = 300 = 0x12C

The "Event Timer" specifies the time after which a PDO is sent, even if no change has been made to the PDO data. The time should be entered in milliseconds, and the timer is started each time an event (change of PDO data) occurs. If the time is shorter than the "Inhibit Time", another event is generated after the "Inhibit Time" elapses!

⚠ CAUTION

Attention

The "Event Timer" can only be used for transfer types 254/255.

⚠ CAUTION

Attention

An object entry can only be mapped in **max. 3 different** PDOs.

10.1.3.4.2.23 Object 0x1A00 … 0x1A1F, Transmit PDO Mapping Parameter

Table 118: Object 0x1A00 … 0x1A1F, Transmit PDO Mapping Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1A00 through 0x1A1F	0	Number of mapped objects	Unsigned8	RW	-
	1 to 8	1st object through 8th object	Unsigned32	RW	-

This object is used to determine the data that is sent via the PDO.
Sub-index 0 contains the number of objects that are valid for the PDO.
Structure of objects 1 to 8:

Table 119: Object 0x1A00 … 0x1A1F-Structure, Transmit PDO Mapping Parameter

Bit31	Bit16	Bit 15	Bit8	Bit7	Bit0
Index (Index of the object to be transmitted)		Sub-Index (Sub-index of the object to be transmitted)		Object Length (Size of object in bits Because a maximum of 8 bytes can be transmitted in one PDO, the sum of the valid object lengths may not exceed 64 (8 bytes x 8 bits).)	

10.1.3.4.3 Manufacturer Specific Profile Area

The non-standard device profile-specific I/O functionality of I/O modules and other modules, as well as other functionalities (e.g., empty module configuration, etc.) are mapped in the 'Master Specific Area' profile. The objects defined therein provide data word widths from 1 to 8 bytes.

Indices 0x2000 (digital inputs), 0x2100 (digital outputs), 0x2400 (2-byte I/O module inputs) and 0x2500 (2-byte I/O module outputs) are relfected by the respective indices of device profile DS 401 (0x6000, 0x6200, 0x6401, 0x6411). This means that, for example, object 0x2000 and object 0x6000 refer to the same location in the process image.

The following table lists all objects of the manufacturer profile supported by the fieldbus coupler/controller.

Idx	Name	Type	Explanation
0x2000	Digital Inputs	Array Unsigned 8	Data of digital input modules
0x2100	Digital Outputs	Array Unsigned 8	Data of digital output modules
0x2200	1-byte module inputs	Array Unsigned 8	Data of 1-byte input modules
0x2300	1-byte module outputs	Array Unsigned 8	Data of 1-byte output modules
0x2400	2-byte module inputs	Array Unsigned 16	Data of 2-byte input modules
0x2500	2-byte module outputs	Array Unsigned 16	Data of 2-byte output modules
0x2600	3-byte module inputs	Record	Data of 3-byte input modules
0x2700	3-byte module outputs	Record	Data of 3-byte output modules
0x2800	4-byte module inputs	Record	Data of 4-byte input modules
0x2900	4-byte module outputs	Record	Data of 4-byte output modules
0x3000	5-byte module inputs	Record	Data of 5-byte input modules
0x3100	5-byte module outputs	Record	Data of 5-byte output modules
0x3200	6-byte module inputs	Record	Data of 6-byte input modules
0x3300	6-byte module outputs	Record	Data of 6-byte output modules
0x3400	7-byte module inputs	Record	Data of 7-byte input modules
0x3500	7-byte module outputs	Record	Data of 7-byte output modules
0x3600	8-byte module inputs	Record	Data of 8-byte input modules
0x3700	8-byte module outputs	Record	Data of 8-byte output modules
0x3800-0x380F	Special 9+ byte input	Array Unsigned 8	Data of I/O modules with more than 8 bytes of input data
0x3900-0x390F	Special 9+ byte output	Array Unsigned 8	Data of I/O modules with more than 8 bytes of output data
0x4200-0x4202	Gateway module input	Record	Input data of gateway modules
0x4300-0x4302	Gateway module output	Record	Output data of gateway modules
0x4500	Empty module configuration	Record	Configuration of virtual I/O modules
0x5000	Input PI	Record	Reading the input process image
0x5001	Output PI	Record	Writing the output process image
0x5200	Coupler configuration object	Record	PDO processing, flashing indicator, disable I/O module diagnostics, selection of content for object 0x1002, data content acc. mode setting of object 0x5200/Sub6, content of SDO 0x1002
0x5201	Configuration object diagnostics	Array of Byte	Diagnostic behavior of connected I/O modules
0x5202	Module configuration object	Array of Byte	Physical configuration of connected I/O modules

10.1.3.4.3.1 Object 0x2000 – Digital Inputs

Table 120: Object 0x2000 – Digital Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2000	0	8-bit digital input block	Unsigned8	RO	-	Number of 8-bit digital input blocks
	1	1st digital input block	Unsigned8	RO	-	1st digital input block
	….	….	….	….	….	….
	255	255th digital input block	Unsigned8	RO	-	255th digital input block

10.1.3.4.3.2 Object 0x2100 – Digital Outputs

Table 121: Object 0x2100 – Digital Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2100	0	8-bit digital output block	Unsigned8	RO	-	Number of 8-bit digital output blocks
	1	1st digital output block	Unsigned8	RW	0	1st digital output block
	….	….	….	….	….	….
	255	255th digital output block	Unsigned8	RW	0	255th digital output block

10.1.3.4.3.3 Object 0x2200 – 1-Byte I/O Modules, Inputs

Table 122: Object 0x2200 – 1-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2200	0	Special 1 byte input	Unsigned8	RO	-	Number of 1-byte special channels
	1	1st special input	Unsigned8	RO	-	1st input channel
	….	….	….	….	….	….
	254	254th special input	Unsigned8	RO	-	254th input channel

10.1.3.4.3.4 Object 0x2300 – 1-Byte I/O Modules, Outputs

Table 123: Object 0x2300 – 1-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2300	0	special 1 byte output	Unsigned8	RO	-	Number of 1-byte special channels
	1	1st special output	Unsigned8	RW	0	1st output channel
	….	….	….	….	….	….
	254	254th special output	Unsigned8	RW	0	254th output channel

10.1.3.4.3.5 Object 0x2400 – 2-Byte I/O Modules, Inputs

Table 124: Object 0x2400 – 2-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2400	0	special 2 byte output	Unsigned8	RO	-	Number of 2-byte special channels
	1	1st special input	Unsigned16	RO	-	1st input channel
	….	….	….	….	….	….
	254	254th special input	Unsigned16	RO	-	254th input channel

10.1.3.4.3.6 Object 0x2500 – 2-Byte I/O Modules, Outputs

Table 125: Object 0x2500 – 2-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2500	0	special 2 byte output	Unsigned8	RO	-	Number of 2-byte special channels
	1	1st special output	Unsigned16	RW	0	1st output channel
	….	….	….	….	….	….
		254th special output	Unsigned16	RW	0	254th output channel

10.1.3.4.3.7 Object 0x2600 – 3-Byte I/O Modules, Inputs

Table 126: Object 0x2600 – 3-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2600	0	special 3 byte input	Unsigned8	RO	-	Number of 3-byte special channels
	1	1st special input	Unsigned24			1st input channel
	….	….	….	….	….	….
	170	170th special input	Unsigned24	RO	-	170th input channel

10.1.3.4.3.8 Object 0x2700 – 3-Byte I/O Modules, Outputs

Table 127: Object 0x2700 – 3-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2700	0	special 3 byte output	Unsigned8	RO	-	Number of 3-byte special channels
	1	1st special output	Unsigned24	RW	0	1st output channel
	….	….	….	….	….	….
	170	170th special output	Unsigned24	RW	0	170. Output channel

10.1.3.4.3.9 Object 0x2800 – 4-Byte I/O Modules, Inputs

Table 128: Object 0x2800 – 4-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2800	0	special 4 byte input	Unsigned8	RO	-	Number of 4-byte special channels
	1	1st special input	Unsigned32	RO	-	1st input channel
	....	....	....	....	....	....
	128	128th special input	Unsigned32	RO	-	128th input channel

10.1.3.4.3.10 Object 0x2900 – 4-Byte I/O Modules, Inputs

Table 129: Object 0x2900 – 4-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x2900	0	special 4 byte output	Unsigned8	RO	-	Number of 4-byte special channels
	1	1st special input	Unsigned32	RO	0	1st output channel
	….	....	....	....	....	....
	128	128th special output	Unsigned32	RW	0	128th output channel

10.1.3.4.3.11 Object 0x3000 – 5-Byte I/O Modules, Inputs

Table 130: Object 0x3000 – 5-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3000	0	special 5 byte input	Unsigned8	RO	-	Number of 5-byte special channels
	1	1st special input	Unsigned40	RO	-	1st input channel
	….	….	….	….	….	….
	102	102th special input	Unsigned40	RO	-	102nd input channel

10.1.3.4.3.12 Object 0x3100 – 5-Byte I/O Modules, Outputse

Table 131: Object 0x3100 – 5-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3100	0	special 5 byte output	Unsigned8	RO	-	Number of 5-byte special channels
	1	1st special output	Unsigned40	RW	0	1st output channel
	....	....	....	....	....	....
	102	102th special output	Unsigned40	RW	0	102nd output channel

10.1.3.4.3.13 Object 0x3200 – 6-Byte I/O Modules, Inputs

Table 132: Object 0x3200 – 6-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3200	0	special 6 byte input	Unsigned8	RO	-	Number of 6-byte special channels
	1	1st special input	Unsigned48	RO	-	1st input channel
	....	....	....	....	....	....
	85	85th special input	Unsigned48	RO	-	85th input channel

10.1.3.4.3.14 Object 0x3300 – 6-Byte I/O Modules, Outputs

Table 133: Object 0x3300 – 6-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3300	0	Special 6 byte output	Unsigned8	RO	-	Number of 6-byte special channels
	1	1st special output	Unsigned48	RW	0	1st output channel
	....	....	....	....	....	....
	85	85th special output	Unsigned48	RW	0	85th input channel

10.1.3.4.3.15 Object 0x3400 – 7-Byte I/O Modules, Inputs

Table 134: Object 0x3400 – 7-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3400	0	Special 7 byte input	Unsigned8	RO	-	Number of 7-byte special channels
	1	1st special input	Unsigned56	RO	-	1st input channel
	....	....	....	....	....	....
	73	73rd special input	Unsigned56	RO	-	73rd input channel

10.1.3.4.3.16 Object 0x3500 – 7-Byte I/O Modules, Outputs

Table 135: Object 0x3500 – 7-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3500	0	Special 7 byte output	Unsigned8	RO	-	Number of 7-byte special channels
	1	1st special output	Unsigned56	RW	0	1st output channel
	....	....	....	....	....	....
	73	73th special output	Unsigned56	RW	0	73rd output channel

10.1.3.4.3.17 Object 0x3600 – 8-Byte I/O Modules, Inputs

Table 136: Object 0x3600 – 8-Byte I/O Modules, Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3600	0	special 8 byte input	Unsigned8	RO	-	Number of 8-byte special channels
	1	1st special input	Unsigned64	RO	-	1st input channel
	….	….	….	….	….	….
	64	64th special input	Unsigned64	RO	-	64th input channel

10.1.3.4.3.18 Object 0x3700 – 8-Byte I/O Modules, Outputs

Table 137: Object 0x3700 – 8-Byte I/O Modules, Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3700	0	Special 8 byte output	Unsigned8	RO	-	Number of 8-byte special channels
	1	1st special output	Unsigned64	RW	0	1st output channel
	….	….	….	….	….	….
	64	64th special output	Unsigned64	RW	0	64th output channel

10.1.3.4.3.19 Object 0x3800 … 0x380F – 9+ Byte I/O Modules – Inputs

Table 138: Object 0x3800 … 0x380F – 9+ Byte I/O Modules – Inputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3800-0x380F	0	Special 9+ byte input	Unsigned8	RO	-	Number of data bytes of the I/O module
	1	1st data byte	Unsigned8	RO	0	1st data byte
	….	….	….	….	….	….
	48	48th data byte	Unsigned8	RO	0	48th data byte

These object contain I/O modules with a data width of min. 9 bytes. One I/O module is mapped per object. Each data byte is assigned to a sub-index. Each sub-index can be mapped in PDOs.
Attention: The respective application is responsible for data consistency.

10.1.3.4.3.20 Object 0x3900 … 0x390F – 9+ Byte I/O Modules – Outputs

Table 139: Object 0x3900 … 0x390F – 9+ Byte I/O Modules – Outputs

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x3900-0x390F	0	Special 9+ byte input	Unsigned8	RO	-	Number of data bytes of the I/O module
	1	1st data byte	Unsigned8	RW	0	1st data byte
	….	….	….	….	….	….
	48	48th data byte	Unsigned8	RW	0	48th data byte

These object contain I/O modules with a data width of min. 9 bytes. One I/O module is mapped per object. Each data byte is assigned to a sub-index. Each sub-index can be mapped in PDOs.
Attention: The respective application is responsible for data consistency.

10.1.3.4.3.21 Object 0x4200 … 0x4202 – Gateway Module Input

Table 140: Object 0x4200 … 0x4202 – Gateway Module Input

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x4200-0x4202	0	Largest sub-index supported	Unsigned8	RO	-	Max. supported sub-index
	1	Mailbox length	Unsigned8	RO	-	Size of the mailbox
	2	Mailbox	Octet String	RO	-	Mailbox
	3-30	Gateway process data	Unsigned8	RW	-	Process data of the gateway module

The status byte is shown per module. It takes the first position in the mailbox followed by an empty byte and then the actual mailbox data.

Example:
Status, 6-byte mailbox, 4 bytes of data

Table 141: Data in the Input Process Image

Data in the input process image											
S	-	MB1	MB2	MB3	MB4	MB5	MB6	D1	D2	D3	D4

Table 142: Entries in the Object Directory

Entries in the object directory								
Sub0	6							
Sub1	8							
Sub2	S	-	MB1	MB2	MB3	MB4	MB5	MB6
Sub3	D1							
Sub4	D2							
Sub5	D3							
Sub6	D4							

10.1.3.4.3.22 Object 0x4300 … 0x4302 – Gateway Module Output

Table 143: Object 0x4300 … 0x4302 – Gateway Module Output

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x4300-0x4302	0	Largest sub-index supported	Unsigned8	RO	-	Max. supported sub-index
	1	Mailbox length	Unsigned8	RO	-	Size of the mailbox
	2	Mailbox	Octet String	RW	-	Mailbox
	3-30	Gateway process data	Unsigned8	RW	-	Process data of the gateway module

The control byte is shown per module. It takes the first position in the mailbox followed by an empty byte and then the actual mailbox data.

Example:
Control, 6-byte mailbox, 4 bytes of data

Table 144: Control – 6-Byte Mailbox, 4 Bytes of Data

Data in the input process image											
C	-	MB1	MB2	MB3	MB4	MB5	MB6	D1	D2	D3	D4

Table 145: Entries in the Object Directory

Entries in the object directory								
Sub0	6							
Sub1	8							
Sub2	C	-	MB1	MB2	MB3	MB4	MB5	MB6
Sub3	D1							
Sub4	D2							
Sub5	D3							
Sub6	D4							

10.1.3.4.3.23 Object 0x4500 – Empty Module Configuration

Table 146: Object 0x4500 – Empty Module Configuration

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x4500	0	Number of virtual I/O modules connected	Unsigned8	RW	0	0 not active 1 .. 64 number of max. I/O modules connected (physical + virtual)
	1	1st I/O module description	Unsigned16	RW	0	1st I/O module
	....	....	....	....	....	....
	64	64th I/O module description	Unsigned16	RW	0	64th I/O module

This object can be used to add virtual I/O modules to the node. You can, for example, design a maximum node configuration and then create a new node that represents a subset of the maximum configuration. From it's object entries, the new node behaves as the maximum configuration with respect to connected I/O

modules. Other applications (CANopen master, etc.), designed for the maximum configuration, access any subset without changing its settings.
Likewise, you can add blanks spaces from the start for possible future extensions, making subsequent mapping adjustments unnecessary.

- Sub-index 0
  Sub-index 0 = 0: Show virtual I/O modules not activ
  Sub-index 0 ≠ 0 Show virtual I/O modules active

The entry specifies the number of connected I/O modules in the maximum configuration.

When setting the value of 0 to > 0, the configruation as described from sub-index 1 is generated. When generating the new configuration, all previous settings that are not stored permanently are overwritten and the process image reset. For this reason, index 0x4500 should always be configured first and then all other settings (mapping, sync time, etc.).
Sub-index 0 can only be set when in the PRE-OPERATIONAL state.

When the new configuration is generated with no errors, an emergency is transmitted with the parameters PP=LL=SS=0.
If an error occurs when generating the new configuration (e.g., more I/O modules are connected than configured), a corresponding Emergency is transmitted. The fieldbus coupler/controller starts with the default configuration based on the connected I/O modules and goes into the STOP state.

- Emergency Message
  Error Code 0x5000
  Error Register 0x81

- Additional Code 00 03 **PP LL SS**
  PP: Specifies the physical slot of the I/O module where the error has occurred
  LL: Specifies the logical slot (slot in the maximum configuration) of the I/O module where the error has occurred
  SS: Cause of the error

Emergency setup in the event of a faulty configuration

Table 147: Parameter: SS (Cause of the Error)

Bit 4..7	Bit 0..3	Description
0	1	Analog input/output module acc. configuration expected
0	2	Digital input/output module acc. configuration expected
0	3	Output module acc. configuration expected
0	4	Input module acc. configuration expected
0 *N*	5	Digital input/output module: - Incorrect number of bits of the entire module (bits pro channel * No. of channels). Analog input/output module: - Module with n channels acc. configuration expected
0	6	More I/O modules connected than configured
0	7	Digital input module / output modules: - invalid Analog input module / output modules: - Incorrect number of bytes per channel Gateway I/O modules: - Incorrect total size in the process image
0	8	Gateway I/O module acc. configuration expected
0	9	Incorrect specification of the mailbox size
0	10	Fewer I/O modules connected than configured

Note

Important note!

If dignostic messages from I/O modules arise from the Emergence message, display of the I/O module position always refers to the logical position of the I/O module in the node. Consequently, these messages are always the same, regardless of the node configuration.

Sub-Index 1..64

Sub-index 1..64 contains the configuration of the node in the maximum configuration. Each index stands for a connected I/O module (sub-index.1 1st I/O module, sub-index.2 2nd I/O module, etc.). In these indices, the respective I/O module is described in detail.

Structure of the sub-index:

Table 148: Structure of the Sub-Index

MSB															LSB
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Kl	Reserved			Bits/Bytes						Channels			Output	Input	A/D

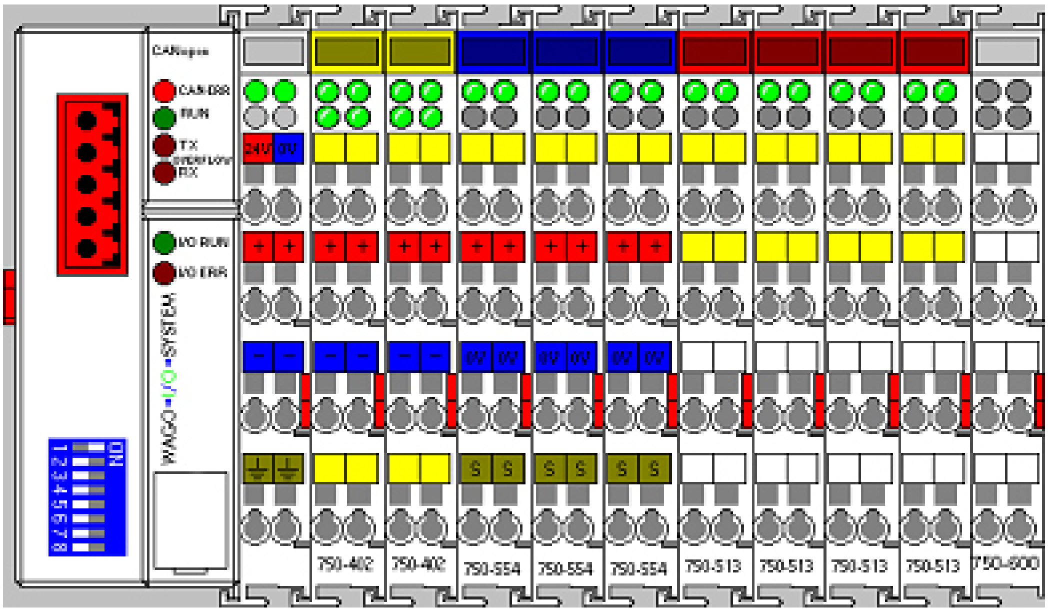
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Kl	G	Total size						MB size					Outp ut	Inpu t	A/D

A/D:	Indicates whether an analog or digital input/output module is at issue 0 = analog, 1 = digital
Input:	Indicates whether an input module is at issue 0 = No input module, 1 = Input module [*1)]
Output:	Indicates whether an output module is at issue 0 = No output module, 1 = Output module [*1)]
Channels:	Indicates the number of channels of the I/O module
Bits/Bytes:	Specifies the number of bytes (analog input/output module) or bits (digital input/output module) per channel shown in the process image
Reserved:	Reserved
Kl:	Indicates if the I/O module is plugged in 0 = I/O module not plugged in, 1 = I/O module plugged in
MB size:	Indicates the size of the mailbox
Total size:	Indicates the size in bytes of the entire gateway module in the PI (mailbox size + process data)
G	Indicates whether a gateway module is at issue 0 = No gateway module, 1 = Gateway module

[*1)] The input and output bit can be set at the same time (e.g., digital output module with diagnostics, this I/O module has input and output bits)

Example:

A node with the following configuration is designed:



Module 1 Module 2 Module 3 Module 4 Module 5 Module 6 Module 7 Module 8 Module 9

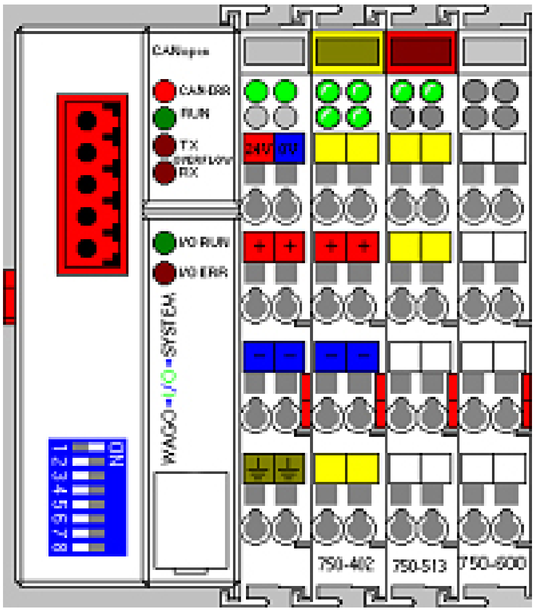
Figure 69: Example Configuration of a Node, Maximum Configuration

Configuration for the maximum configuration:

Table 149: Configuration for the Maximum Configuration

Sub-Index	Value	Explanation
0	0x09	Total number of connected(9) and virtual(0) I/O modules
1	0x8063	Digital, input, 4 channels, 1 bit per channel, connected (Bl. 1)
2	0x8063	Digital, input, 4 channels, 1 bit per channel, connected (Bl. 2)
3	0x 8094	Analog, input, 2 channels, 2 bytes per channel, connected (Bl. 3)
4	0x 8094	Analog, input, 2 channels, 2 bytes per channel, connected (Bl. 4)
5	0x 8094	Analog, input, 2 channels, 2 bytes per channel, connected (Bl. 5)
6	0x 8055	Digital, output, 2 channels, 1 bit per channel, connected (Bl. 6)
7	0x 8055	Digital, output, 2 channels, 1 bit per channel, connected (Bl. 7)
8	0x 8055	Digital, output, 2 channels, 1 bit per channel, connected (Bl. 8)
9	0x 8055	Digital, output, 2 channels, 1 bit per channel, connected (Bl. 9)

Secondly, a node is configured that is a subset of the first. Only I/O modules 2 and 8 are used.



Module 2 Module 8

Figure 70: Example Configuration of a Node, Subset

Configuration for extension Bl. 2 and Bl. 3, connected:

Table 150: Configuration for the Extension Bl. 2 and Bl. 8, Connected:

S-Idx.	Value	Explanation
0	0x09	Total number of connected(2) and virtual(7) I/O modules
1	0x0063	Digital, input, 4 channels, 1 bit per channel, not connected (Bl. 1)
2	0x8063	Digital, input, 4 channels, 1 bit per channel, connected (Bl. 2)
3	0x 0094	Analog, input, 2 channels, 2 bytes per channel, not connected (Bl. 3)
4	0x 0094	Analog, input, 2 channels, 2 bytes per channel, not connected (Bl. 4)
5	0x 0055	Analog, input, 2 channels, 2 bytes per channel, not connected (Bl. 5)
6	0x 0055	Digital, output, 2 channels, 1 bit per channel, not connected (Bl. 6)
7	0x 8055	Digital, output, 2 channels, 1 bit per channel, not connected (Bl. 7)
8	0x 0055	Digital, output, 2 channels, 1 bit per channel, connected (Bl. 8)
9	0x 0055	Digital, output, 2 channels, 1 bit per channel, not connected (Bl. 9)

With respect to their entries in the object directory (connected I/O modules), the behavior of both nodes is identical. Thus, the behavior is identical for PDO mapping.
For example, bit 4 must be set in index 0x6200, sub-index 1 to set the 1st channel of output module 8 (750-513).

This procedure is the same for both configurations.

Without an empty module configuration, in contrast, bit 0 must be set in the 2nd configuration, index 0x6200, sub-index 1 to set the same output channel.

10.1.3.4.3.24 Object 0x5000 – Read Input Process Image

Table 151: Object 0x5000 – Read Input Process Image

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x5000	0	Number of input bytes	Unsigned16	RO	-	Number of relevent bytes in the input PI
	1	Input segment 1	Octet_String	RO	-	1st input PI segment (the lowest 255 byte from the 512-byte PI)
	2	Input segment 2	Octet_String	RO	-	2nd input PI segment (the highest 255 byte from the 512-byte PI. Only available if > 255 bytes of input data)

Permits reading the entire input process image as a domain via SDO with access to all input data “at one time”.

Note

Important note!

Because access via the SDO is slow, time-critical data should only be transferred via the PDO.

10.1.3.4.3.25 Object 0x5001 – Write Output Process Image

Table 152: Object 0x5001 – Write Output Process Image

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x5001	0	Number of output bytes	Unsigned16	RO	none	Number of relevant bytes in the output PI
	1	Output segment 1	Octet_String	RW	none	1st output PI segment (the lowest 255 byte from the 512-byte PI)
	2	Output segment 2	Octet_String	RW	none	2nd output PI segment (the highest 255 byte from the 512-byte PI. Only available if > 255 bytes of output data, corresponding I/O module configuration possible)

Permits writing the entire output process image as a domain via SDO “at one time”.

Note

Important note!

Because access via the SDO is slow, time-critical data should only be transferred via the PDO.

10.1.3.4.3.26 Object 0x5200 – Fieldbus Coupler/Controller Configuration

Table 153: Object 0x5200 – Fieldbus Coupler/Controller Configuration

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x5200	0	Max. supported sub-index	Unsigned8	RO	-	Maximum supported sub-index
	1	PDO processing	Unsigned8	RW	0	Defines how received PDOs are processed 0: Consequent output of data, i.e., when a PDO has arrived twice, an internal data bus cycle is carried out 1: Obtain data from the last PDO
	2	Disable flashing indicator “Warning Level”	Unsigned8	RW	0	Enable/disable transmission of I/O module diagnostic messages via Emergency 0: Send diagnostic messages 1: Do not send diagnostic messages
	3	Disable I/O module diagnostics globally via “Emergency Messages”	Unsigned8	RW	0	Enable/disable transmission of I/O module diagnostic messages via Emergency 0: Send diagnostic messages 1: Do not send diagnostic messages
	4	-	-	-	-	Reserved, no access possible
	5	-	-	-	-	Reserved, no access possible
	6	Selection of content for object 0x1002	Unsigned 32	RW	0	Sub-index 5 defines how the fieldbus coupler/controller should respond to a read request of object 0x1002.

Table 153: Object 0x5200 – Fieldbus Coupler/Controller Configuration

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation

Table 154: Object 0x5200/Sub-Index 6 – Structure

31	30	8	7	0
Copy	Reserved		Mode	

Mode: The mode defines the status of object 0x1002 and the data content for object 0x5200/Sub-Index 7.

- 0: Object 0x1002 disabled.
  Read and write access to object 0x5200/sub-index 7/sub-index 8 are not possible
- 1: Object 0x1002 enabled.
  Object 0x5200/sub-index 7 data content according to table "Object 0x5200/Sub-Index 7 – Structure for Mode = 1" (see Sub-Index 7).
- 2 … 255: reserved

Copy: Defines content and access rights of object 0x5200/sub-index 8

- 0: Object 0x5200/sub-index 8 can be written by the user (read/write)
- 1: Object 0x5200/sub-index 8 contains a copy of object 0x5200/sub-index 7 (read only)

Note

Attention!

The data for object 0x1002 is always a copy of object 0x5200/sub-index 8!

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
	7	Data content acc. mode selection of object 0x5200/Sub6	Unsigned 32	RO	0	Sub-index 7 contains the data acc. mode setting of object 0x5200/sub-index 6.

Table 155: Object 0x5200/Sub-Index 7 – Structure for Mode = 1

	LSB						MSB	
UINT32	Status Byte 1		Status Byte 1		Status Byte 1		Status Byte 1	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	TxOver	RxOver	SyncE	MailbW	GatewE	EmergE	-	HwE
Byte 2	-	9PlusE	SubDis	PDO>	PDO<	HeartbE	LifeE	PassiveE
Byte 3	Current CANopen bus state							
Byte 4	NN							

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
	8	Content of SDO 0x1002	Unsigned 32	RW/RO	0	This value is used as the source for object 0x1002. The value depends on the "Copy" bit of object 0x5200/sub-index 6.

Table 153: Object 0x5200 – Fieldbus Coupler/Controller Configuration

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
	9	Conversion diagnostic message Emergency	Unsigned8	RW	0	0: Diagnostic emergency messages are sent by the device in the previous format (no digital 8-channel diagnostics possible; backwards compatible). 1: Diagnostic emergency messages for digital input/output I/O modules are sent channel-selected with the diagnostic value of the channel. In this format, the diagnostic messages from digital 8-channel modules are transmitted correctly (not backwards compatible).

10.1.3.4.3.27 Object 0x5201 – Diagnostic Configuration

Table 156: Object 0x5201 – Diagnostic Configuration

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x5201	0	Max. supported sub-index = Number of I/O modules	Unsigned8	RO	-	Maximum supported sub-index = Number of I/O modules
	1	Diagnostic behavior of the 1st connected I/O module	Unsigned8	RW	0	Each bit within the byte represents a channel of the I/O module (bit 0 channel 0, bit 1 channel 1 ... bit 7 channel 7) Bit n = 1: No diagnostic messages are sent via Emergency (n=0...7) If all bits are set to 1 (255), transmission of diagnostic messages via Emergency for this I/O module is disabled, regardless of the number of channels of the I/O module.
	…					…
	64	Diagnostic behavior of the 64th connected I/O module	Unsigned8	RW	0	Each bit within the byte represents a channel of the I/O module (bit 0 channel 0, bit 1 channel 1 ... bit 7 channel 7) Bit n = 1: No diagnostic messages are sent via Emergency (n=0...7) If all bits are set to 1 (255), transmission of diagnostic messages via Emergency for this I/O module is disabled, regardless of the number of channels of the I/O module.

10.1.3.4.3.28 Object 0x5202 – Module Configuration

Note

Important note!

The object always refers to the physical configuration, i.e., only the physical modules and no virtual modules are considered even when using the empty module configuration.

Table 157: Object 0x5202 – Module Configuration

Idx	S-Idx	Name	Type	Attributes	Default Value	Explanation
0x5202	0	Max. supported sub-index	Unsigned8	RO	-	Maximum supported sub-index = Number of physical I/O modules including fieldbus couplers/controllers
	1	Module description	Unsigned64	RO	-	Sub-index 1 always refers to the description of the fieldbus coupler/controller
	2	Module description	Unsigned64	RO	-	1st I/O module
	3	Module description	Unsigned64	RO	-	2nd I/O module
	...	...	...	...	...	...
	65	Module description	Unsigned64	RO	-	64th I/O module

Structure of the module description:

Fieldbus couplers/controllers / analog input/output modules

Table 158: Fieldbus Couplers/Controllers / Analog Input/Output Modules

Bit 63 Bit 48	Bit 47 Bit 32	Bit 31 Bit 16	Bit 15 Bit 0
Series	Part number	Series extension	Part number extension

Example of fieldbus coupler/controller 750-3xx / 750-8xx I/O module 750-404/000-001:			
750	3xx / 8xx	0	0
750	404	0	1

Digital input/output modules

Table 159: Digital Input/Output Modules

Bit 63 Bit 48	Bit 47 Bit 32	Bit 31 Bit 16	Bit 15 Bit 0
0	0	0	Digital module description: Bit 0…3: 0: Diagnostics only 1: Inputs only 2: Outputs only 3: Outputs + diagnostics 4: Not used 5: Inputs + diagnostics 6: Not used 7: Inputs + outputs + diagnostics 8: Not used 9: Not used A: Not used B: Outputs + diagnostics C: Not used D: Not used E: Not used F: Inputs + outputs Bit 4…7: Not used Bit 8…14: Data width internal in bit Bit 15: Always 1

Example 750-501:			
0	0	0	0x8202

10.1.3.4.4 Standard Device Profile Area – DS 401

The fieldbus coupler supports the standard device profile *Device Profile for Generic I/O Modules*.

The following table lists all objects of the standard profile DS 401 that are supported by the fieldbus coupler/controller.

Idx	Name	Type	Explanation
0x6000	Read digital input 8-bit	Array Unsigned 8	Data of digital input modules
0x6005	Global interrupt enable digital 8-Bit	Boolean	Global release of transmission of 8-bit digital input data
0x6006	Digital interrupt mask any change 8-Bit	Array Unsigned 8	Release of transmission for each change of 8-bit digital input data
0x6007	Digital interrupt mask low-to-high 8-Bit	Array Unsigned 8	Release of transmission in occurrence of a positive edge of 8-bit digital input data
0x6008	Digital interrupt Mask High-to-Low 8-Bit	Array Unsigned 8	Release of transmission in occurrence of a negative edge of 8-bit digital input data
0x6100	Read digital input 16-bit	Array Unsigned 16	Data of digital input modules
0x6200	Write digital output 8-bit	Array Unsigned 8	Data of digital output modules
0x6206	Error mode digital output 8-bit	Array Unsigned 8	Release of predefined error values of 8-bit digital output data
0x6207	Error value digital output 8-bit	Array Unsigned 8	Predefined error values of the 8-bit digital output data
0x6300	Write digital output 16-bit	Array Unsigned 16	Data of digital output modules
0x6401	Read analog input 16-bit	Array Unsigned 16	Data of the 16-bit analog input modules
0x6411	Write analog output 16-bit	Array Unsigned 16	Data of the 16-bit analog output modules
0x6421	Analog input trigger selection	Array Unsigned 16	Specify trigger frequency for 16-bit analog input data
0x6423	Analog input global interrupt enable	Boolean	Global release of transmission of 16-bit analog input data
0x6424	Analog input interrupt upper limit integer	Array Unsigned 16	Transmission of 32-bit input data when threshold value has been exceeded
0x6425	Analog input interrupt lower limit integer	Array Unsigned 16	Transmission of 32-bit input data when threshold value has been undershot
0x6426	Analog input interrupt delta unsigned	Array Unsigned 16	Transmission when the 16-bit input data has changed by at least the delta value
0x6427	Analog input interrupt negative delta unsigned	Array Unsigned 16	Transmission when the 16-bit input data has decreased by at least the delta value
0x6428	Analog input interrupt positive delta unsigned	Array Unsigned 16	Transmission when the 16-bit input data has increased by at least the delta value
0x6443	Analog output error mode	Array Unsigned 8	Enabling for predefined error values of the 16-bit output data
0x6444	Analog output error value integer	Array Unsigned 16	Value in the case of error in the 16-bit output data
0x67FE	Error behavior	Array Unsigned 8	State change in the case of error

10.1.3.4.4.1 Object 0x6000 – Read Digital Input 8-Bit

Table 160: Object 0x6000 – Read Digital Input 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6000	0	Number of digital input blocks	Unsigned8	RO	-
	1	1st digital input block	Unsigned8	RO	-
	2	2nd digital input block	Unsigned8	RO	-
	…	…	…	…	…
	255	255th digital input block	Unsigned8	RO	-

This object contains the process data of the digital input modules.
Sub-index 1 contains the first 8 digital input channels counted from left to right – beginning from the fieldbus coupler/controller. Sub-index 2 contains the next, etc.

10.1.3.4.4.2 Object 0x6005 – Global Interrupt Enable Digital 8-Bit

Table 161: Object 0x6005 – Global Interrupt Enable Digital 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6005	0	Global interrupt enable digital 8-bit	Boolean	RW	1

This object is used to transmit digital input data via PDO. If the value = 1, the transmission is generally enabled; it depends only on objects 0x6006 … 0x6008 and the PDO's transmission type. If the value = 0, no digital input data is transmitted, independent of objects 0x6006...0x6008.

10.1.3.4.4.3 Object 0x6006 – Digital Interrupt Mask Any Change 8-Bit

Table 162: Object 0x6006 – Digital Interrupt Mask Any Change 8-bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6006	0	Number of digital input blocks	Unsigned8	RO	-
	1	Mask 1st input block	Unsigned8	RW	255
	2	Mask 2nd input block	Unsigned8	RW	255
	....	....	....	....	....
	64	Mask 64th input block	Unsigned8	RW	255

This object is used to determine which digital input channel transmits its data with a change. Condition for this is that the transmission is generally enabled (object 0x6005 = 1).
0 = Transmission with change disabled (per channel)
1 = Transmission with change enabled (per channel)

Example: Sub-index 0 = 1, sub-index 1 = 65 = 0x41
only channels 1 and 7 transmit data with change

10.1.3.4.4.4 Object 0x6007 – Digital Interrupt Mask Low-to-High 8-Bit

Table 163: Object 0x6007 – Digital Interrupt Mask Low-to-High 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6007	0	Number of digital input blocks Unsigned8	Unsigned8	RO	-
	1	Mask 1st input block	Unsigned8	RW	0
	2	Mask 2nd input block	Unsigned8	RW	0
	....	....	....	....	....
	64	Mask 64th input block	Unsigned8	RW	0

This object is used to determine which digital input channel transmits its data with a positive edge (0 to 1 change). Condition for this is that the transmission is generally enabled (object 0x6005 = 1). This object has the OR link for object 0x6006.

0 = Transmission with positive edge disabled (per channel)
1 = Transmission with positive edge enabled (per channel)

Example: Index 0x6006 sub-index 0 = 1, sub-index 1 = 65 = 0x41
Index 0x6007 sub-index 0 = 1 sub-index 1 = 33 =0x21
Channels 1 and 7 always transmit their data with change
Channel 6 is only transmitted with a positive edge

10.1.3.4.4.5 Object 0x6008 – Digital Interrupt Mask High-to-Low 8-Bit

Table 164: Object 0x6008 – Digital Interrupt Mask High-to-Low 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6008	0	Number of digital input blocks	Unsigned8	RO	-
	1	Mask 1st input block	Unsigned8	RW	0
	2	Mask 2nd input block	Unsigned8	RW	0
	....	....	....	....	....
	64	Mask 64th input block	Unsigned8	RW	0

This object is used to determine which digital input channel transmits its data with a negative edge (1 to 0 change). Condition for this is that the transmission is generally enabled (object 0x6005 = 1). This object has the OR link for object 0x6006.

0 = Transmission with negative edge disabled (per channel)
1 = Transmission with negative edge enabled (per channel)

Example: Index 0x6006 sub-index 0 = 1, sub-index 1 = 65 = 0x41
Index 0x6008 sub-index 0 = 1 sub-index 1 = 33 =0x21
Channels 1 and 7 always transmit their data with change
Channel 6 is only transmitted with a negative edge

10.1.3.4.4.6 Object 0x6100 – Read Digital Input 16-Bit

Table 165: Object 0x6100 – Read Digital Input 16-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6100	0	Number of digital input blocks	Unsigned8	RO	-
	1	1st digital input block	Unsigned16	RO	-
	2	2nd digital input block	Unsigned16	RO	-
	…	…	…	…	…
	128	128th digital input block	Unsigned16	RO	-

This object contains the process data of the digital input modules.
Sub-index 1 contains the first 16 digital input channels counted from left to right – beginning from the fieldbus coupler/controller. Sub-index 2 contains the next, etc.

10.1.3.4.4.7 Object 0x6200 – Write Digital Output 8-Bit

Table 166: Object 0x6200 – Write Digital Output 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6200	0	Number of digital output blocks	Unsigned8	RO	-
	1	1st digital output block	Unsigned8	RW	0
	2	2nd digital output block	Unsigned8	RW	0
	....	....	....	....	....
	255	255th digital output block	Unsigned8	RW	0

This object contains the process data of the digital output modules.
Sub-index 1 contains the first 8 digital output channels counted from left to right – beginning from the fieldbus coupler/controller. Sub-index 2 contains the next, etc.

10.1.3.4.4.8 Object 0x6206 – Error Mode Digital Output 8-Bit

Table 167: Object 0x6206 – Error Mode Digital Output 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6206	0	Number of digital output blocks	Unsigned8	RO	-
	1	Mask 1st digital output block	Unsigned8	RW	255
	2	Mask 2nd digital output block	Unsigned8	RW	255
	....	....	....	....	....
	64	Mask 64th digital output block	Unsigned8	RW	255

This object is used to define if the outputs switch to a predefined fault condition (see object 0x6207) in case of error (e.g., fieldbus coupler/controller switches to the *Stopped* state, Node-Guarding has failed, etc.). When the error is corrected, the outputs remain in their current state, i.e., the set fault condition of the output channels remains unchanged.

0 = Outputs remain unchanged (per channel)
1 = Outputs switch to a predefined fault condition (per channel)

10.1.3.4.4.9 Object 0x6207 – Error Value Digital Output 8-Bit

Table 168: Object 0x6207 – Error Value Digital Output 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6207	0	Number of digital output blocks	Unsigned8	RO	-
	1	Mask 1st digital output block	Unsigned8	RW	0
	2	Mask 2nd digital output block	Unsigned8	RW	0
	....	....	....	....	....
	64	Mask 64th digital output block	Unsigned8	RW	0

This object is used to define the values that the outputs should adopt in the error occurs. This requires that the corresponding bit is set in object 0x6206.

0 = Output to 0 (per channel)

1 = Output to 1 (per channel)

Example: Index 0x6206 sub-index 0 = 1, sub-index 1 = 65 = 0x41
Index 0x6207 sub-index 0 = 1 sub-index 1 = 33 = 0x21
Channel 1 is set to 1, channel 7 is set to 0,
all other output channels remain unchanged if an error occurs

10.1.3.4.4.10 Object 0x6300 – Write Digital Output 16-Bit

Table 169: Object 0x6300 – Write Digital Output 16-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6300	0	Number of digital output blocks	Unsigned8	RO	-
	1	1st digital output block	Unsigned16	RW	0
	2	2nd digital output block	Unsigned16	RW	0
	....	....	....	....	....
	128	128th digital output block	Unsigned16	RW	0

This object contains the process data of the digital output modules.
Sub-index 1 contains the first 16 digital output channels counted from left to right – beginning from the fieldbus coupler/controller. Sub-index 2 contains the next, etc.

10.1.3.4.4.11 Object 0x6401 – Read Analog Input 16-Bit

Table 170: Object 0x6401 – Read Analog Input 16-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6401	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	1th channel	Unsigned16	RO	-
	....	....	....	....	....
	254	254th channel	Unsigned16	RO	-

This object contains the process data of the analog input modules.
Sub-index 1 contains the first analog input channel counted from left to right – beginning from the fieldbus coupler/controller. Sub-index 2 the second, etc.

10.1.3.4.4.12 Object 0x6411 – Write Analog Output 16-Bit

Table 171: Object 0x6411 – Write Analog Output 16-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6411	0	Number of analog output channels (16-bit)	Unsigned8	RO	-
	1	1th channel	Unsigned16	RW	0
	....	....	....	....	....
	254	254th channel	Unsigned16	RW	0

This object contains the process data of the analog output modules.
Sub-index 1 contains the first analog input channel counted from left to right – beginning from the fieldbus coupler/controller. Sub-index 2 the second, etc.

10.1.3.4.4.13 Object 0x6421 – Analog Input Interrupt Trigger Selection

Table 172: Object 0x6421 – Analog Input Interrupt Trigger Selection

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6421	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	Trigger 1th channel	Unsigned8	RW	7
	....	....	....	....	....
	128	Trigger 128th channel	Unsigned8	RW	7

This object is used to determine the condition of the transmission. The transmission requires a 1 to be entered in object 0x6423, thus releasing general transmission.

Table 173: Object 0x6421 – Structure of Sub-Index 1..128

Bit	Transmission Requirement	Configuration Sub-Index
0	Threshold value exceeded	0x6424
1	Threshold value undershot	0x6425
2	Change of the input value is greater than the delta value to the last transmission	0x6426
3	Reduction in input value by more than delta value from last transmission	0x6427
4	Increase in input value by more than delta value from last transmission	0x6428
5 to 7	Reserved	-

10.1.3.4.4.14 Object 0x6423 – Analog Input Global Interrupt Enable

Table 174: Object 0x6423 – Analog Input Global Interrupt Enable

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6423	0	Analog input global interrupt enable	Unsigned8	RW	0

This object is used to transmit analog input data via PDO. If the value = 1, the transmission is generally enabled; it depends only on object 0x6421 and the

PDO's transmission type. If the value = 0, no analog input data is transmitted, independent of object 0x6421.

⚠ WARNING

Attention!

Transmission of analog input data is disabled by default.

10.1.3.4.4.15 Object 0x6424 – Analog Input Interrupt Upper Limit Integer

Table 175: Object 0x6424 – Analog Input Interrupt Upper Limit Integer

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6424	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	Threshold value of 1th channel	Unsigned16	RW	0
	....	....	....	....	....
	128	Threshold 128th channel	Unsigned16	RW	0

This object makes threshold monitoring possible when configured in object 0x6423. If an input value is greater than or equal to the specified threshold value, transmission occurs as long as another trigger condition (e.g., object 0x6426) is not set.

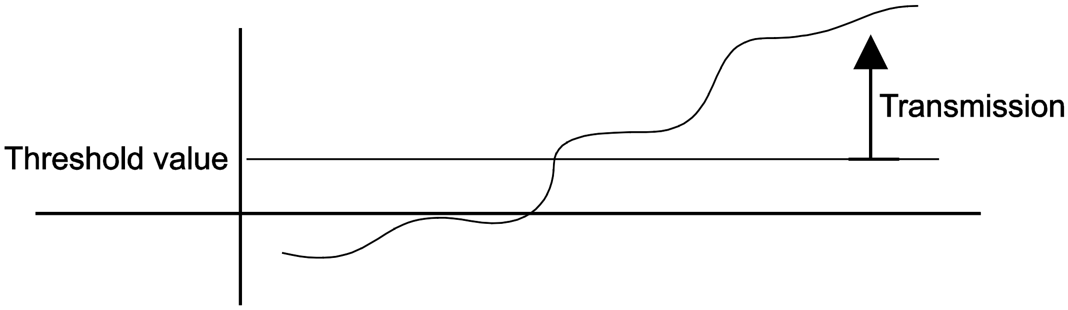


Figure 71: Threshold Value Monitoring

10.1.3.4.4.16 Object 0x6425 – Analog Input Interrupt Lower Limit Integer

Table 176: Object 0x6425 – Analog Input Interrupt Lower Limit Integer

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6425	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	Threshold value of 1th channel	Unsigned16	RW	0
	....	....	....	....	....
	128	Threshold 128th channel	Unsigned16	RW	0

This object makes threshold monitoring possible when configured in object 0x6423. If an input value is less than or equal to the specified threshold value, transmission occurs as long as another trigger condition (e.g., object 0x6426) is not set.

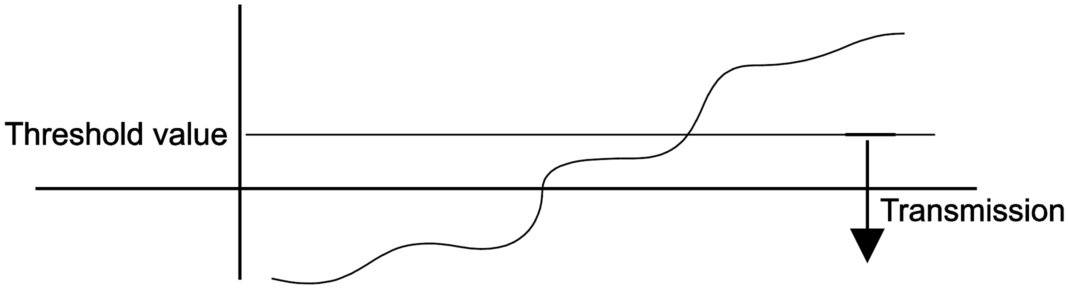


Figure 72: Threshold Value Monitoring

10.1.3.4.4.17 Object 0x6426 – Analog Input Interrupt Delta Unsigned

Table 177: Object 0x6426 – Analog Input Interrupt Delta Unsigned

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6426	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	Delta value 1st channel	Unsigned16	RW	0
	....	....	....	....	....
	128	Delta value 128th channel	Unsigned16	RW	0

By defining this object, the value that is to be retransmitted must be larger or smaller than the previously sent value by at least the delta value.
This object can, for example, be linked with object 0x6424 so that transmission only takes place after both the set threshold value and the delta function are achieved.

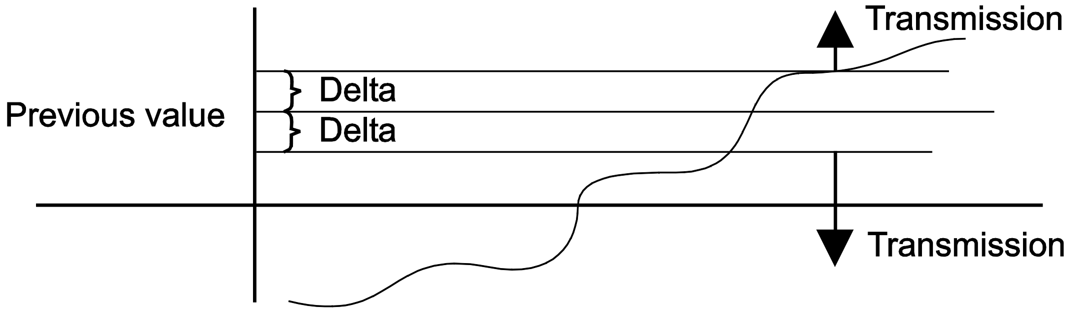


Figure 73: Threshold Value Monitoring

10.1.3.4.4.18 Object 0x6427 – Analog Input Interrupt Negative Delta Unsigned

Table 178: Object 0x6427 – Analog Input Interrupt Negative Delta Unsigned

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6427	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	Delta value 1st channel	Unsigned16	RW	
	....	....	....	....	....
	128	Delta value 128th channel	Unsigned16	RW	0

By defining this object, the value that is to be retransmitted must be smaller than the previously sent value by at least the delta value.

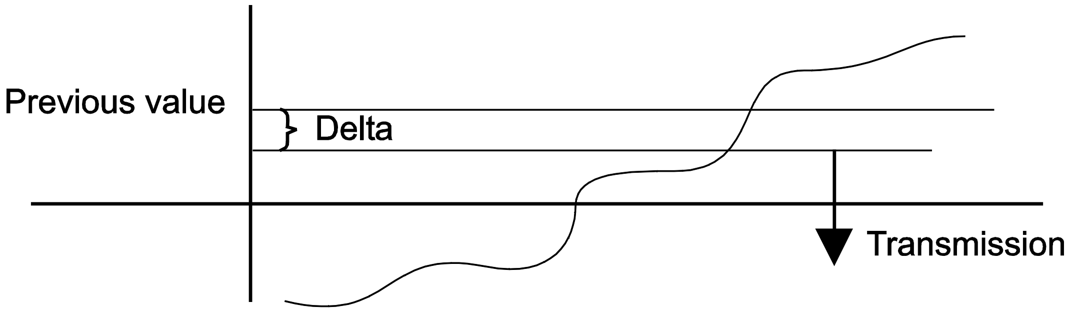


Figure 74: Threshold Value Monitoring

10.1.3.4.4.19 Object 0x6428 – Analog Input Interrupt Positive Delta Unsigned

Table 179: Object 0x6428 – Analog input interrupt positive delta unsigned

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6428	0	Number of analog input channels (16-bit)	Unsigned8	RO	-
	1	Delta value 1st channel	Unsigned16	RW	0
	....	....	....	....	....
	128	Delta value 128th channel	Unsigned16	RW	0

By defining this object, the value that is to be retransmitted must be larger than the previously sent value by at least the delta value.

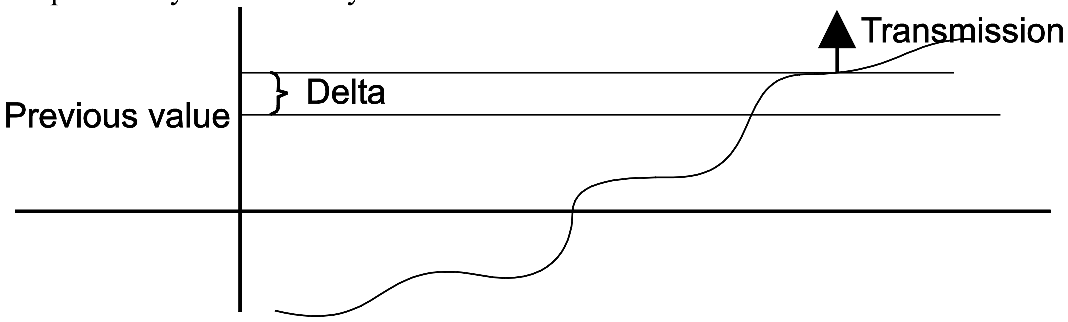


Figure 75: Threshold Value Monitoring

10.1.3.4.4.20 Object 0x6443 – Analog Output Error Mode

Table 180: Object 0x6443 – Analog Output Error Mode

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6443	0	Number of analog output channels (16-bit)	Unsigned8	RO	-
	1	Error mode 1st channel	Unsigned8	RW	1
	....	....	....	....	....
	128	Error mode 128 channel	Unsigned8	RW	1

This object is used to define if the outputs switch to a predefined fault condition (see object 0x6444) in case of error (e.g., fieldbus coupler/controller switches to the *Stopped* state, Node-Guarding has failed, etc.). When the error is corrected, the outputs remain in their current state, i.e., the set fault condition of the output channels remains unchanged.

All analog outputs not covered with object 0x6444 (e.g., 6-byte analog input/output modules) are always set to 0 in case of error.
0 = Output remains unchanged
1 = Output switches to a predefined fault condition

10.1.3.4.4.21 Object 0x6444 – Analog Output Error Value Integer

Table 181: Object 0x6444 – Analog Output Error Value Integer

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6444	0	Number of analog output channels (16-bit)	Unsigned8	RO	-
	1	Error value of 1st channel	Unsigned16	RW	0
	....	....	....	....	....
	128	Error value 128th channel	Unsigned16	RW	0

This object is used to define the values that the outputs should adopt in the error occurs. This requires that the corresponding bit is set in object 0x6443.

10.1.3.4.4.22 Object 0x67FE – Error Behavior

Table 182: Object 0x67FE – Error Behavior

Idx	S-Idx	Name	Type	Attribute	Default Value
0x67FE	0	Max. supported sub-index	Unsigned8	RO	1
	1	General communication error	Unsigned8	RW	0

This object is used to determine in which state the module switches in the event of a communication error (e.g., Node Guarding failure).

Table 183: Object 0x67FE – Structure of a Communication Error Entry:

General communication error	Action
0	Switches to the PRE-OPERATIONAL state (only if the current state was OPERATIONAL)
1	No state change
2	Switch to STOPPED state

10.1.3.4.4.23 Object 0xA000-0xFFFF, Reserved Area

This object directory area remains unassigned for the fieldbus coupler/controller.

10.1.3.5 PDO Transmission

Data transmission via PDO is only possible in the OPERATIONAL state.

When switching to the OPERATIONAL state, all Tx-PDOs are transmitted once with the transmission type 254 and 255.

Note

Special transmission type 254 and 255 (index 0x1800 ... 0x181F, sub-index 2)
Please note that due to the default value (=FALSE) that is specified by the device profile DS401 object 0x6423 (Analogue Input Global Interrupt Enable), analog changes are not transmitted. This prevents the CAN bus from being overloaded with CAN messages. To prevent an overflow when setting object 0x6423 = TRUE, a correspondingly long Inhibit Time can be selected. In addition, there is the option to reduce the amount of messages by configuring the objects for the threshold value monitoring (objects 0x6421, 0x6424, 0x6425) and for the delta functions (objects 0x6426, 0x6427, 0x6428).

10.1.3.5.1 Mapping

This object is used to determine which data is transmitted via PDOs.

If no stored customer-specific configuration is used and no other settings are made, the objec directory is assigned with a default configuration according to device profile DS 401 (see section “Fieldbus Communication” > … > “Initialization”).

If the fieldbus coupler/controller is in the PRE-OPERATIONAL state, it's mapping can be modified via SDOs instead based on the application.

Information

Additional information
An example for creating a customized mapping configuration is explained in the section “Commissioning”.

10.1.3.5.2 Transmit PDO1

The following diagram shows an overview of the relevant objects and their connection for PDO transmission of digital inputs.

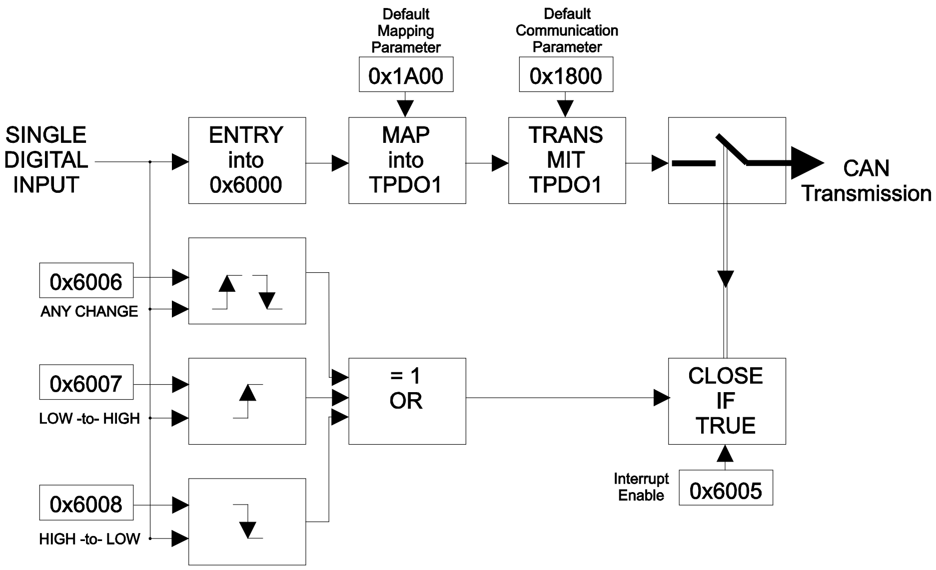


Figure 76: PDO Transmission of Digital Input Module Data

Index	Object name	Description
0x1800	Transmit PDO communication parameter	Communication parameters for the transmit PDOs
0x1A00	Transmit PDO mapping parameters	Mapping parameters for the transmit PDOs
0x6000	Read digital input 8-bit	Data of digital input modules
0x6005	Global interrupt enable digital 8-bit	Global release of the transmission of 8-bit digital input data
0x6006	Digital interrupt mask any change 8-Bit	Release of transmission for each change of 8-bit digital input modules
0x6007	Digital interrupt mask low-to-high 8-bit	Enabling transmission in occurerence of a positive edge of 8-bit digital input data
0x6008	Digital interrupt mask high-to-low 8-bit	Release of transmission in occurrence of a negative edge of 8-bit digital input data

10.1.3.5.3 Receive PDO1

The following diagram shows an overview of the relevant objects and their connection for PDO transmission of digital outputs.

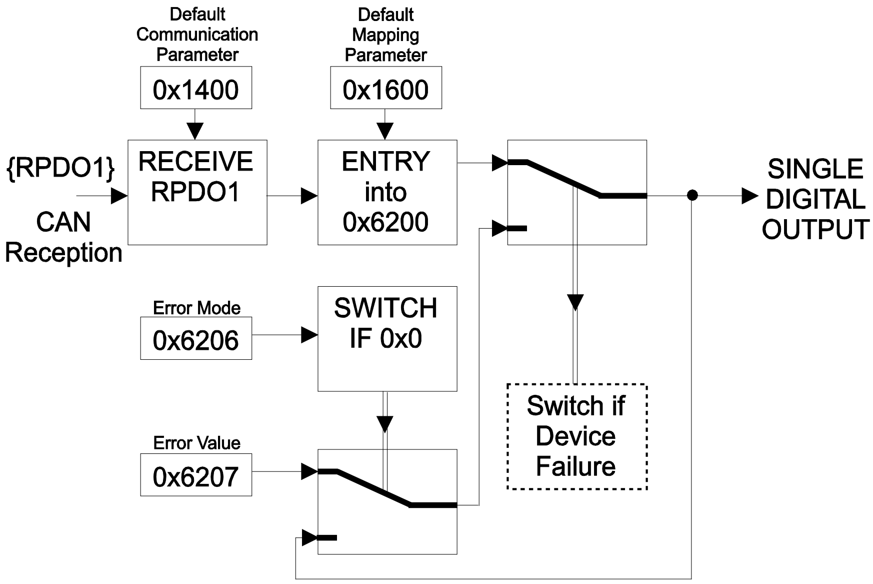


Figure 77: PDO Transmission of Digital Output Module Data

Index	Object name	Description
0x1400	Receive PDO communication parameter	Communication parameter for receive PDO
0x1600	Transmit PDO mapping parameters	Mapping parameters for the transmit PDOs
0x6200	Write digital output 8-bit	Data of digital output modules
0x6206	Error mode output 8-bit	Release of predefined error values of 8-bit digital output data
0x6207	Error value digital output 8-bit	Predefined error values of the 8-bit digital output data

10.1.3.5.4 Transmit PDO2

The following diagram shows an overview of the relevant objects and their connection for PDO transmission of analog inputs.

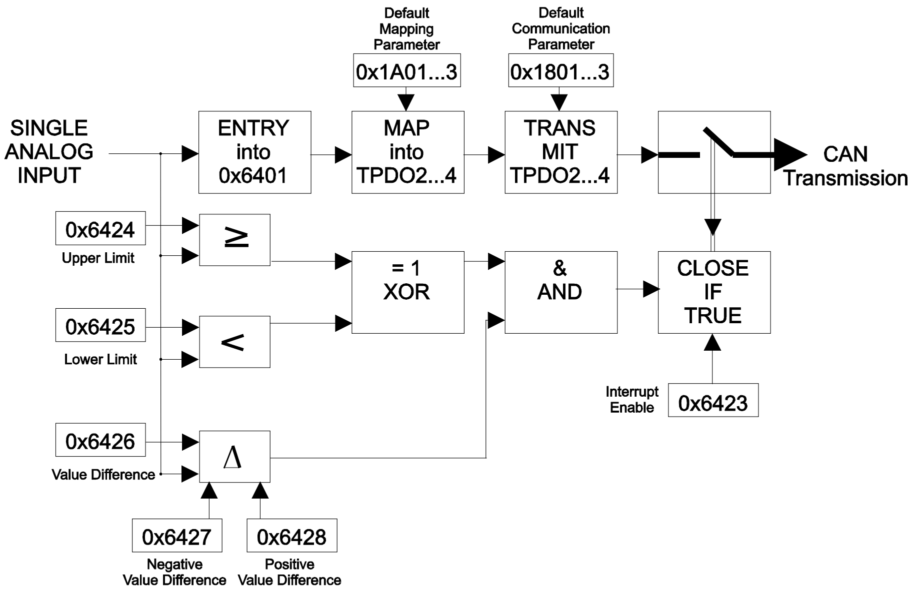


Figure 78: PDO Transmission of Analog Input Module Data

Index	Object name	Description
0x1801...3	Transmit PDO communication parameter	Communication parameters for the transmit PDO
0x1A01…3	Transmit PDO mapping parameters	Mapping parameters for the transmit PDOs
0x6401	Read analog input 16-bit	Specify trigger frequency for 16-bit analog input data
0x6423	Analog input global interrupt enable	Global release of transmission of 16-bit analog input data
0x6424	Analog input interrupt upper limit integer	Transmission of 16-bit input data when threshold value has been exceeded
0x6425	Analog input interrupt lower limit integer	Transmission of 16-bit input data when threshold value has been undershot
0x6426	Analog input interrupt delta unsigned	Transmission when the 16-bit input data has changed by at least the delta value
0x6427	Analog input interrupt negative delta unsigned	Transmission when the 16-bit input data has decreased by at least the delta value
0x6428	Analog input interrupt positive delta unsigned	Transmission when the 16-bit input data has increased by at least the delta value

10.1.3.5.5 Receive PDO2

The following diagram shows an overview of the relevant objects and their connection for PDO transmission of analog output.

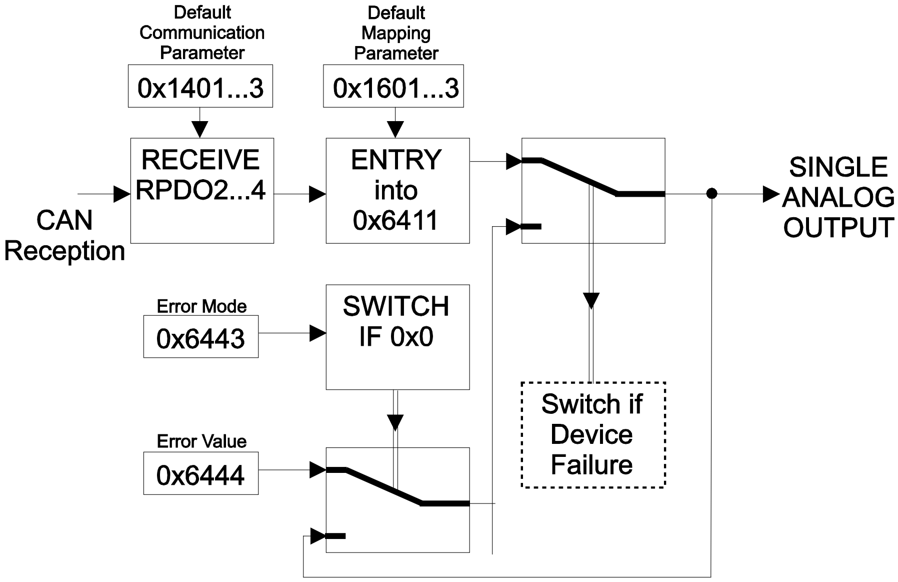


Figure 79: PDO Transmission of Analog Output Module Data

Index	Object name	Description
0x1401…3	Receive PDO communication parameter	Communication parameter for receive PDO
0x1601…3	Transmit PDO mapping parameters	Mapping parameter for the transmit PDO
0x6411	Write analog output 16-bit	Data of the 16-bit analog output modules
0x6443	Analog output error mode	Enabling for predefined error values of the 16-bit output data
0x6444	Analog output error value integer	Value in the case of error in the 16-bit output data

10.1.3.6 SYNC Monitoring

If the “Communication Cycle Period” value does not equal 0, and if the fieldbus coupler/controller is in the OPERATIONAL state, monitoring takes place at the first arrival of a SYNC message.

Failure of the SYNC telegram:

The RUN LED flashes rapidly to indicate that no SYNC telegram is received within the monitoring time (Communication Cycle Period). At the same time, an Emergency telegram (Error Code: 0x8100, Error Register: 0x81, Additional Code: 00 04 00 00 00) is transmitted. If the MASTER causes a state change to come about, the failure of the SYNC telegram is indicated nonetheless.

The LEDs indicate normal operating status only after repeated receipt of the SYNC message in the OPERATIONAL state. An Emergency telegram (Error Code: 0x0000, Error Register: 0x81, Additional Code: 00 04 00 000) is transmitted again to indicate that SYNC monitoring is working again.

10.1.3.7 Node Guarding

Node Guarding starts for the fieldbus coupler/controller when the first “Remote Transmit Request” telegram (RTR) is received with the COB ID for “Node Guarding” (0x700+moduleID). If the fieldbus coupler/controller receives no corresponding telegram, it does not monitor the “Node Guarding”.

Node Guarding is switched off by default because 0 is entered in the respective indices (0x100C = Guard-Time, 0x100D = Life Time Factor).

The NMT master prompts the fieldbus coupler/controller at regular time intervals. This period is called the Guard Time (index 0x100C). The internal state of the fieldbus coupler/controller can be found in the reply telegram.

If an RTR request arrives and the Guard Time has not been set, Node Guarding is not monitored, but the fieldbus coupler/controller responds with its internal state.

The states are coded as follows:

Table 184: States

State:	Value:
PRE-OPERATIONAL	127
OPERATIONAL	5
STOP	4

The Life Time is the product of the Guard Time (index 0x100C) and the Life Time Factor (index 0x100D).

Failure of Node Guarding:

The RUN LED flashes rapidly to indicate that no Node Guarding telegram is received within the Life Time. At the same time, an Emergency telegram (Error Code: 0x8130, Error Register: 0x11, Additional Code: 0x00 04 00 000 0) is transmitted, the outputs are set according to objects 0x6206, 0x6207, 0x6443 and 0x6444 and the fieldbus coupler/controller switches to the predefined state according to object 0x67FE.

Once the Node Guarding protocol is recorded again, an Emergency telegram (Error Code: 0x0000, Error Register: 0x11, Additional Code: 00 04 00 000 0) is transmitted again to indicate that Node Guarding is enabled again. The outputs and fieldbus coupler/controller state remain unchanged.

Only the node guarding protocol or the heartbeat protocol may be used. The Heartbeat protocol is always used if the Heartbeat Producer Time is configured.

10.1.3.8 Heartbeat Monitoring

This protocol makes it possible to carry out module monitoring without using RTR frames.

The heartbeat generator generates a message in a cyclical manner (time interval defined in object 0x1017), in which it transmits the state of the module. Transmission begins immediately after configuration of object 0x1017. The message can be evaluated by one or more heartbeat consumers (object 0x1016). Up to 5 modules can be monitored simultaneously. Monitoring begins with the first arrival of a heartbeat telegram (separately for each module to be monitored).

Failure of the heartbeat:

The RUN LED flashes rapidly to indicate that no Heartbeat telegram is received within the time configured (object 0x1016). At the same time, an Emergency telegram (Error Code: 0x8130, Error Register: 0x11, Additional Code: 0x00 05 KK 00 00, KK node number that triggered the EMCY) is transmitted. The outputs are set according to objects 0x6206, 0x6207, 0x6443 and 0x6444 and the fieldbus coupler/controller switches to the predefined state according to object 0x67FE.

Once the Heartbeat protocol is recorded again, an Emergency telegram (Error Code: 0x0000, Error Register: 0x11, Additional Code: 0x00 05 KK 00 00) is transmitted again to indicate that the Heartbeat is enabled again. The outputs and fieldbus coupler/controller state remain unchanged. If several modules are monitored, the blink code for the heartbeat failure only stops when the last heartbeat resumes.

Only the node guarding protocol or the heartbeat protocol may be used. The Heartbeat protocol is always used if the Heartbeat Producer Time is configured.

10.1.3.9 Error Messages (Emergency)

Emergency messages are always transmitted when a critical error situation in the device has occured / has been resolved or important information must be communicated to other devices.

The structure and meaning of the entries in the Emergency object are explained in the following table. They are encoded in the bus telegram in the order low byte / high byte.

After the error has been corrected, an Emergency object is also transmitted (Error Code = 0x0000, the Error Register and Additional Code function as described in the “EMCY-CODE” table below).

After power-on, an Emergency object is transmitted when the loaded settings are the default settings. There can be two reasons for this:

- No settings have been saved (index 0x1010).
- The fieldbus coupler/controller has discarded the saved settings because I/O modules were plugged in or pulled.

Table 185: EMCY-CODE

Byte:	**0 1**	**2**	**3 7**	
Name	**Error code**	**Error register**	**Additional code**	**Explanation**
	0x0000*	0x00	00 00 00 00 00	The “predefined error field” index 0x1003, sub-index 0 has been set to zero or all errors have been corrected.
	0x5000*	0x81	00 01 00 00 00	Modified hardware configuration after power-own or reset node/communication of the fieldbus coupler/controller has reinitialized because there is no saved configuration or the saved configuration does not match the current configuration.
	0x5000*	0x81	00 02 00 00 00	Flash error An error occurred when saving the configuration to the flash.
	0x5000*	0x81	00 03 PP LL SS	Programmed configuration does not match the actual configuration. PP: Physical slot of the I/O module where the error occurred LL: Logical slot of the I/O module where the error occurred SS: Error cause
	0x5000*	0x81	00 09 00 00 00	Queue overflow for Emergency messages (only occurs when the inhibit time is used for emergency)
	0x5000*	0x81	00 0A 01 00 00	Max. number of gateway I/O modules exceeded or max. PI size exceeded by gateway I/O modules
	0x5000*	0x81	00 0A 02 00 00	Max. mailbox size exceeded

Table 185: EMCY-CODE

Byte:	**0 1**	**2**	**3 7**	
Name	**Error code**	**Error register**	**Additional code**	**Explanation**
	0x8100*	0x81	00 04 00 00 00	The time between two synch objects is longer than the Communication_Cycle_Period
	0x8110*	0x11	00 01 00 00 00	Internal receive buffer overflow, state change as defined in object 0x67FE. The outputs are switched as defined in Error Mode/Value Objects.
	0x8110*	0x11	00 02 00 00 00	Internal transmit buffer overflow, state change as defined in object 0x67FE. The outputs are switched as defined in Error Mode/Value Objects.
	0x8120*	0x11	00 03 00 00 00	CAN fieldbus coupler/controller in “Error Passive Mode”
	0x8130*	0x11	00 04 00 00 00	The duration between two Node-Guarding Telegrams is longer than the Guard_Time X Life_Time_Factor.
	0x8130*	0x11	00 05 KK 00 00	The time between two heartbeat telegrams is longer than configured. KK: Node that triggered the time-out
	0x8210*	0x81	00 05 SS II NN	PDO was transmitted with a smaller number of bytes than configured in the Communication Profile. The PDO data was discarded; i.e., the outputs remain unchanged. SS: Target value, configured value (e.g., in index 0x1600 sub-index 0) II: Actual value, number of bytes sent NN: Number of PDOs (1..32)
	0x8220*	0x81	00 08 SS II NN	PDO was transmitted with a larger number of bytes than configured in the Communication Profile. Only the first n-data is used (n = total length configured in object directory). SS: Target value, configured value (total length of all valid configured objects in bytes) II: Actual value, number of bytes sent NN: Number of PDOs (1..32)
	0xFF00*	0x81	00 06 PP 00 00	Internal data buse error, switch to STOP state - PP: Position of the I/O module

Table 185: EMCY-CODE

Byte:	0 1	2	3 7	
Name	**Error code**	**Error register**	**Additional code**	**Explanation**
	0xFF00*	0x81	DD 07 PP SK NN	Diagnostic message - DD: Diagnostic byte - PP: Position of the I/O module - SK: Error status and channel number - NN: Number of current I/O module errors

* Byte 0 = Low Byte and Byte 1 = High Byte

Example: Error Code 0x8220: Byte 0 = 0x20, Byte 1= 0x82

10.1.3.9.1 Diagnostic Messages of the I/O Modules

If an error occurs in an I/O module that supports diagnostics, the diagnostic status of the respective channel is transmitted via the Emergency message.

Design of the Additional Code of the Diagnosis message:

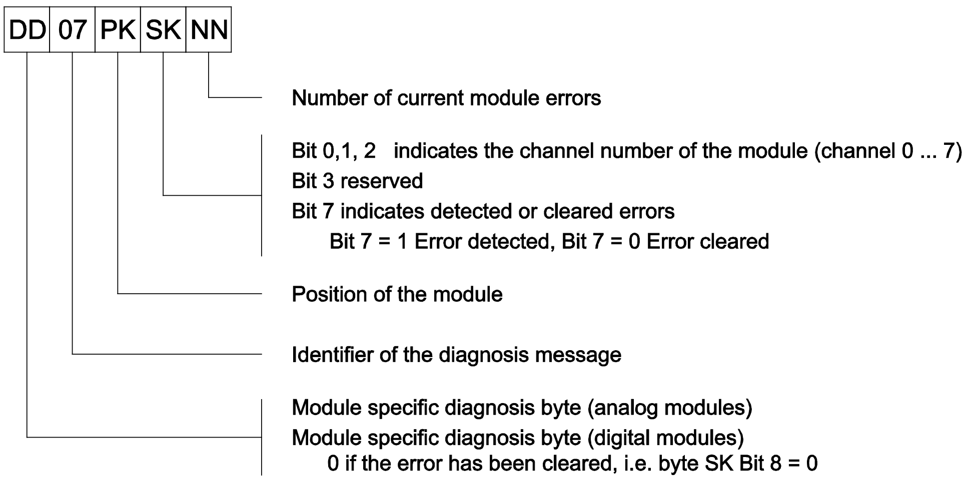


Figure 80: Structure of the "Additional Code"

Note

Display of max. 255 errors!
If 255 or more errors occur, the number of I/O module errors displayed remains at 255! The counter value is represented as 0xFF even with an increasing number if I/O module errors.

All I/O module errors beyond 255 are counted internally. The counter value remains unchanged at 0xFF.

Note

Updating the EMCY message with parallel diagnostics!

If a diagnostic is resolved on one channel at the same time a new diagnostic arises, an EMCY message is transmitted with updated DD and SK bit 7 = 1.

Example 1:

- 2-channel analog input module 750-465, plugged into position 14, the current on channel 0 is greater than 20 mA

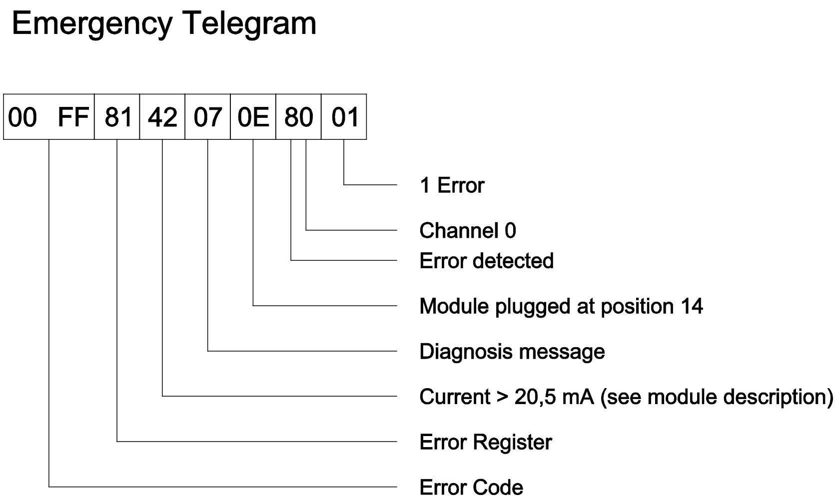


Figure 81: Structure of Emergency Telegram 1

- If an error in addition to the first error occurs on a 2-channel digital output module 750-506, there is a wire break on channel 1, the I/O module is at position 17.

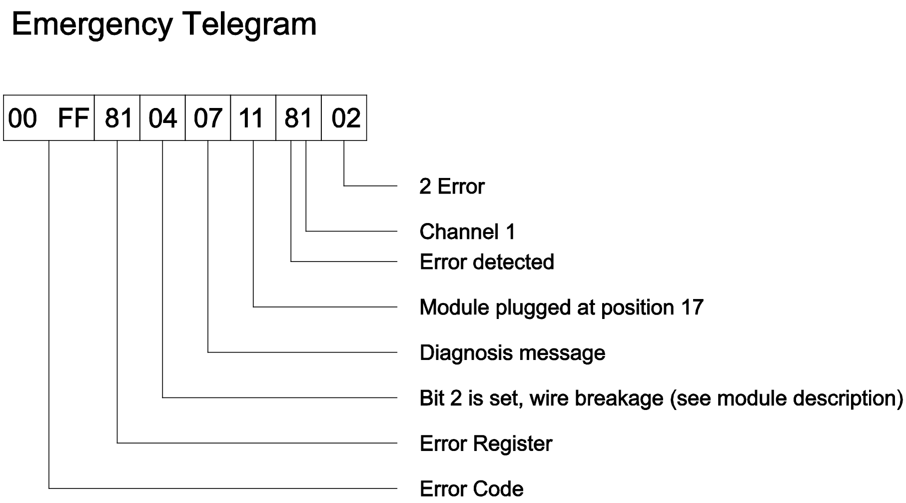


Figure 82: Structure of Emergency Telegram 2

- The error (wire break on the 2-channel digital output module 750-506) is corrected.

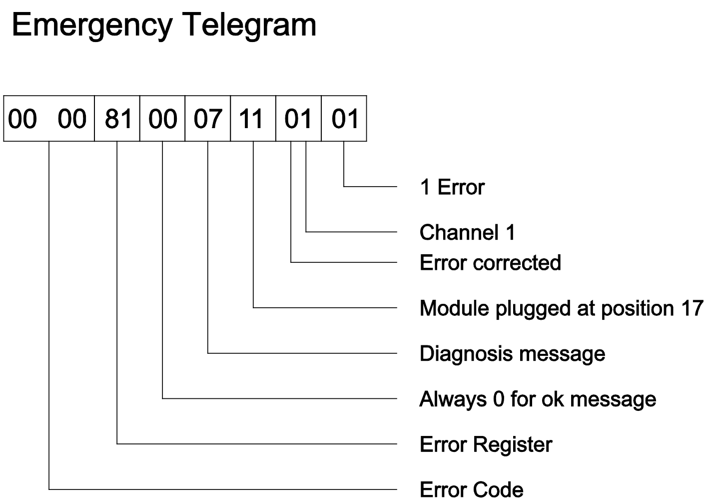


Figure 83: Structure of Emergency Telegram 3

Note

Diagnostic data is shown in the process image!

Please note that the diagnostic data is shown in the process image for digital input/output modules with diagnostics.
A 2-channel digital output module 750-506 sends 4 bits total for the error case described above to the output process image (1 bit per channel and 1 bit unused) and 49 bits total to the input process image (2 bit diagnostics per channel - wire break, short circuit).

Example 2:

- Proportional valve module 750-632 (1 channel, 2 outputs), plugged in at position 4, error at both outputs.

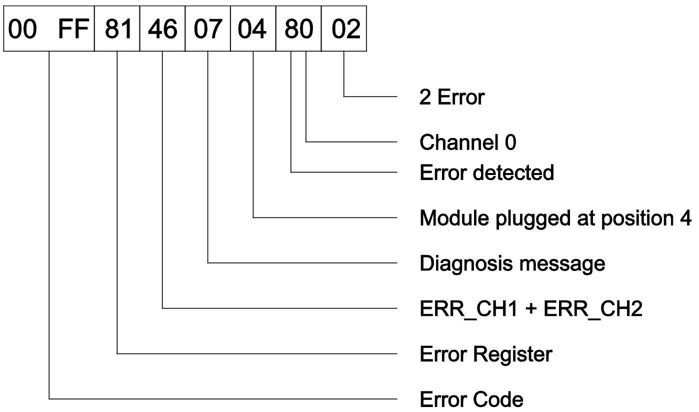


Figure 84: Example – Error at Proportional Valve Module

- Error at output 2 has been corrected, error at output 1 remains (750-632)

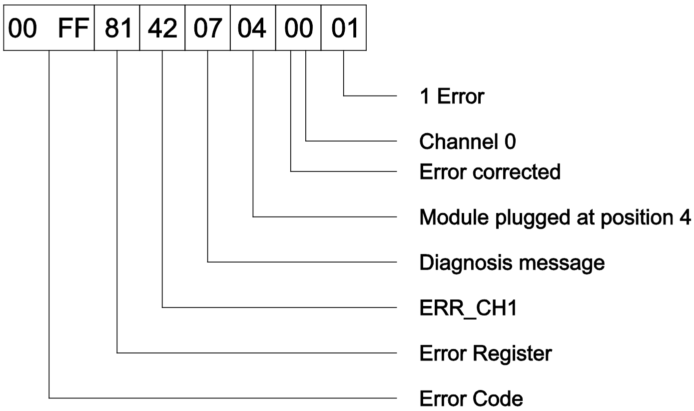


Figure 85: Example – Error at Proportional Valve Module

- All errors at 750-632 (Proportional Valve Module) corrected

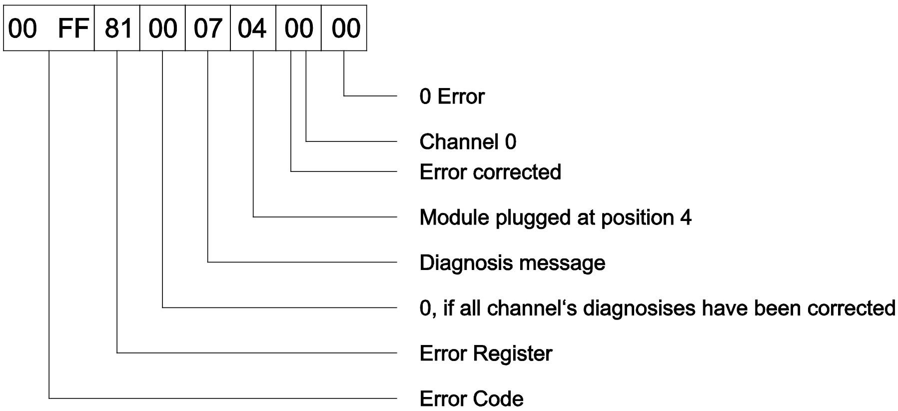


Figure 86: Example – Error at Proportional Valve Module

10.1.3.9.2 Layout of the Module-Specific Diagnostic Bits of the Digital Input/Output Modules

The layout of module-specific diagnostic bits for digital input/output modules can be modified via object 0x5200, sub-index 9.

In the default layout, the diagnostic bits are written to byte DD according to their significance in the entire input process image of the I/O module. This representation is not suitable for digital input/output modules with an input process image > 8 bits, e.g., 750-439 (8-channel DI NAMUR).

Table 186: DD in the Default Layout (750-506)

Input process image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnostic bit S3 channel 2	Diagnostic bit S2 channel 2	Diagnostic bit S1 channel 1	Diagnostic bit S0 channel 1

DD=0x01
DD=0x02
DD=0x04
DD=0x08

For I/O modules whose diagnostics cannot be displayed in the default layout, DD contains a 0xFF.

To display the diagnostics for I/O modules not suitable for the default layout, the EMCY layout can be switched to an alternative representation via object 0x5200 sub-index 9. In the alternative layout, the module-specific diagnostic bits are interpreted channel-by-channel. For I/O modules with one bit diagnostic per channel, DD can only take the values 0x01 and 0x00.

Table 187: DD in the Alternative Layout (750-506)

Input process image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnostic bit S3 channel 2	Diagnostic bit S2 channel 2	Diagnostic bit S1 channel 1	Diagnostic bit S0 channel 1

```
|         |         |         |
|         |         |          DD=0x01
|         |          DD=0x02
|          DD=0x01
 DD=0x02
```

11 I/O Modules

11.1 Overview

For modular applications with the WAGO-I/O-SYSTEM 750/753, different types of I/O modules are available

- Digital Input Modules
- Digital Output Modules
- Analog Input Modules
- Analog Output Modules
- Specialty Modules
- System 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 Structure of Process Data for CANopen

The process image uses a byte structure (without word alignment) for the CANopen fieldbus coupler/controller. The internal mapping method for data greater than one byte conforms to Intel formats.

The following section describes the representation for WAGO-I/O SYSTEM 750 and 753 Series I/O modules in the process image of the fieldbus coupler/controller, as well as the configuration of the process values.

In the CANopen object directory, the entire input process image can be read via index 0x5000. The entire output process image can be written via index 0x5001.

NOTICE

Equipment damage due to incorrect address!

To prevent any damage to the device in the field, you must always take the process data for all previous byte or bit-oriented I/O modules into account when addressing an I/O module at any position in the fieldbus node.

11.2.1 Digital Input Modules

Digital input modules output one bit as the process value per signal channel that indicates the status of the respective channel. Bits that represent input process values are entered in 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/controller. In the event of a diagnostic message, the fieldbus coupler/controller enters the state of the diagnostic bit in the diagnostic status word. The entries in the diagnostic status word are made channel-specific.

If analog input modules are in the node, the digital data is grouped in bytes and added to the analog input data in the input process image.

1-Channel Digital Input Modules with Diagnostics

750-435

Table 188: 1-Channel Digital Input Modules with Status

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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

2-Channel Digital Input Modules

750-400, -401, -405, -406, -410, -411, -412, -425, -427, -438, (and all variants),
753-400, -401, -405, -406, -410, -411, -412, -425, -427

Table 189: 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

2-Channel Digital Input Modules with Diagnostics

750-400, -401, -410, -411, -419, -421, -424, -425
753-400, -401, -410, -411, -421, -424, -425

Table 190: Process Data of 2-Channel Digital Input Modules 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

2-Channel Digital Input Modules with Diagnostics and Output Data

750-418, -419, -421
753-418, -421

In addition to process values for input process image, the digital input module provide 4 bits of data in the output process image.

Table 191: 2-Channel Digital Input Modules with Diagnostics and Output 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

Output process image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Acknowledgement bit Q 2 Channel 2	Acknowledgement bit Q 1 Channel 1	0	0

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

4-Channel Digital Input Modules

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

Table 192: 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

8-Channel Digital Input Modules

750-430, -431, -436, -437
753-430, -431, -434

Table 193: 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

16-Channel Digital Input Modules

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

Table 194: 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

11.2.2 Digital Output Modules

The digital output modules contain one bit as the process value per channel that indicates the status of the respective 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/controller. 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.

If analog output modules are in the node, the digital data is grouped in bytes and added after the analog output data in the output process image.

1-Channel Digital Output Modules with Input Data

750-523

In addition to the process value bit in the output process image, the digital output modules also provides 1 bit that is represented in the input process image. This status image shows “Manual operation”.

Table 195: 1-Channel Digital Output Modules with Input 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 operation"

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

2-Channel Digital Output Modules

750-501, -502, -509, -512, -513, -514, -517, -535, (and all variants),
753-501, -502, -509, -512, -513, -514, -517

Table 196: 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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

2-Channel Digital Output Modules with Input Data

750-507 (-508), -522,
753-507

Table 197: 2-Channel Digital Output Modules with Input Data

Input process image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diag. bit S2 Channel 2	Diag. bit S1 Channel 1

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

750-506,
753-506

In addition to the 4-bit process values in the output process image, the digital output module 750-506, 753-506 provides 4 bits of data in the input process image. A diagnostic bit for each output channel indicates an overload, a short circuit or a wire breakage via 2-bit error code.

Table 198: 4-Channel Digital Output Modules 75x-506 with Input Data

Input process image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diag. bit S3 Channel 2	Diag. bit S2 Channel 2	Diag. bit S1 Channel 1	Diag. bit S0 Channel 1

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

Output process image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Not used	Not used	Controls DO 2 Channel 2	Controls DO 1 Channel 1

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

4-Channel Digital Output Modules

750-504, -516, -519, -531
753-504, -516, -531, -540

Table 199: 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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

4-Channel Digital Output Modules with Input Data

750-532

In addition to 4-bit process values in the output process image, the digital output modules 750-532 provide 4 bits of data in the input process image. A diagnostic bit for each output channel indicates an overload, short circuit or wire breakage.

Table 200: 4-channel digital output modules 750-532 with Input Data

Input process image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diag. bit S3 Channel 4	Diag. bit S2 Channel 3	Diag. bit S1 Channel 2	Diag. bit S0 Channel 1

Diag. bit S = '0' no error
Diag. bit S = '1' wire break, short circuit or overload

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

8-Channel Digital Output Modules

750-530, -536
753-530, -534

Table 201: 8-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 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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

8-Channel Digital Output Modules with Input Data

750-537

In addition to the 8-bit process values in the output process image, the digital output modules provide 8 bits of data in the input process image. A diagnostic bit for each output channel indicates an overload, short circuit or wire breakage.

Table 202: 4-Channel Digital Output Modules 750-537 with Input Data

Input process image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Diag. bit S7 Channel 8	Diag. bit S6 Channel 7	Diag. bit S5 Channel 6	Diag. bit S4 Channel 5	Diag. bit S3 Channel 4	Diag. bit S2 Channel 3	Diag. bit S1 Channel 2	Diag. bit S0 Channel 1

Diag. bit S = '0' no error
Diag. bit S = '1' wire break, short circuit or overload

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

16-Channel Digital Output Modules

750-1500, -1501, -1504, -1505

Table 203: 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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

8-Channel Digital Input/Output Modules

750-1502, -1506

The digital input/output modules provide 8-bit process values in the input and output process image.

Table 204: 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

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

Object 0x6200 (0x2100 also possible) is used for the digital outputs.

11.2.3 Analog Input Modules

The analog input modules provide 16-bit measured values. In the input process image, 16-bit measured values for each channel are mapped in Intel format byte by byte for the CANopen fieldbus coupler/controller.

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.

When digital input modules are also present in the node, the analog input data is always mapped into the Input Process Image in front of the digital data.

1-Channel Analog Input Modules

750-491 (and all variants)

Table 205: 1-Channel Analog Input Modules

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Measured value U_D
	1	D1	
n+1	2	D2	Measured value U_{ref}
	3	D3	

These I/O modules occupy 2x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per measured value.

2-Channel Analog Input Modules

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -492, (and all variants),

753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, 476, -477, 478, -479, -483, -492, (and all variants)

Table 206: 2-Channel Analog Input Modules

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Measured value channel 1
	1	D1	
n+1	2	D2	Measured value channel 2
	3	D3	

These I/O modules occupy 2x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

4-Channel Analog Input Modules

750-453, -455, -457, -459, -460, -468, (and all variants),
753-453, -455, -457, -459

Table 207: 4-Channel Analog Input Modules

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Measured value channel 1
	1	D1	
n+1	2	D2	Measured value channel 2
	3	D3	
n+2	4	D4	Measured value channel 3
	5	D5	
n+3	6	D6	Measured value channel 4
	7	D7	

These I/O modules occupy 4x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

8-Channel Analog Input Modules

Table 208: 8-Channel Analog Input Modules

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Measured value channel 1
	1	D1	
n+1	2	D2	Measured value channel 2
	3	D3	
n+2	4	D4	Measured value channel 3
	5	D5	
n+3	6	D6	Measured value channel 4
	7	D7	
n+4	8	D8	Measured value channel 5
	9	D9	
n+5	10	D10	Measured value channel 6
	11	D11	
n+6	12	D12	Measured value channel 7
	13	D13	
n+7	14	D14	Measured value channel 8
	15	D15	

These I/O modules occupy 8x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

11.2.4 Analog Output Modules

The analog output modules contain 16-bit output values per channel.

In the output process image, 16-bit output values for each channel are mapped in Intel format byte by byte for the CANopen fieldbus coupler/controller.

When digital output modules are also present in the node, the analog output data is always mapped into the Output Process Image in front of the digital data.

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.

2-Channel Analog Output Modules

750-550, -552, -554, -556, -560, -562, 563, -585, (and all variants),
753-550, -552, -554, -556

Table 209: 2-Channel Analog Output Modules

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Output value channel 1
	1	D1	
n+1	2	D2	Output value channel 2
	3	D3	

These I/O modules occupy 2x2 bytes. Object 0x6411 (0x2500 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

4-Channel Analog Output Modules

750-553, -555, -557, -559,
753-553, -555, -557, -559

Table 210: 4-Channel Analog Output Modules

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Output value channel 1
	1	D1	
n+1	2	D2	Output value channel 2
	3	D3	
n+2	4	D4	Output value channel 3
	5	D5	
n+3	6	D6	Output value channel 4
	7	D7	

These I/O modules occupy 4x2 bytes. Object 0x6411 (0x2500 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

8-Channel Analog Output Modules

Table 211: 8-Channel Analog Output Modules

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Output value channel 1
	1	D1	
n+1	2	D2	Output value channel 2
	3	D3	
n+2	4	D4	Output value channel 3
	5	D5	
n+3	6	D6	Output value channel 4
	7	D7	
n+4	8	D8	Output value channel 5
	9	D9	
n+5	10	D10	Output value channel 6
	11	D11	
n+6	12	D12	Output value channel 7
	13	D13	
n+7	14	D14	Output value channel 8
	15	D15	

These I/O modules occupy 8x2 bytes. Object 0x6411 (0x2500 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

11.2.5 Specialty Modules

In addition to the data bytes, the control/status byte is also displayed for select I/O modules. This byte is used for the bi-directional data exchange of the I/O module with the higher-level control system.

The control byte controls the I/O module's behavior and operation. The status byte reports the I/O module's status. It is possible to set the counter with the control byte or indicate a range overflow/underflow with the status byte.

The control/status byte is always in the low byte in the process image.

Information

Information about the control/status byte structure

Please refer to the corresponding description of the I/O modules for the structure of the control/status bytes. You can find a manual with the relevant I/O module description at: http://www.wago.com.

Counter Modules

750-: 404 (and all variants except /000-005)

753-: 404 (and version /000-003)

These counter modules occupy 5 bytes of user data in the input and output area of the process image, 4 data bytes and 1 additional control/status byte. The I/O modules then provide 32-bit counter values. Three words are assigned in the process image via word alignment.

Table 212: Counter Modules 750-404, 753-404

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	-	Not used
	2	D0	Counter value
	3	D1	
	4	D2	
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. One sub-index is occupied per channel.

Table 213: Counter Modules 750-404, 753-404

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C	Status byte
	1	-	Not used
	2	D0	Counter value
	3	D1	
	4	D2	
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per channel.

750-404/000-005

In the input and output process image, the counter modules occupy 5 bytes of user data, 4 data bytes and 1 additional control/status byte. The I/O modules then provide 16-bit counter values per counter. Three words are assigned in the process image via word alignment.

Table 214: Counter Modules 750-404/000-005

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	-	Not used
	2	D0	Counter value of counter 1
	3	D1	
	4	D2	Counter value of counter 2
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. One sub-index is occupied per channel.

Table 215: Counter Modules 750-404/000-005

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C	control byte
	1	-	Not used
	2	D0	Counter setting value counter 1
	3	D1	
	4	D2	Counter setting value counter 2
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per channel.

750-638,
753-638

In the input and output process image, counter modules occupy 6 bytes of user data, 4 data bytes and two additional control/status bytes. The I/O modules then provide 16-bit counter values. 6 bytes are occupied in the process image.

Table 216: Counter Modules 750-638, 753-638

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S0	Status byte of counter 1
	1	D0	Counter value of counter 1
	2	D1	
n+1	3	S1	Status byte of counter 2
	4	D2	Counter value of counter 2
	5	D3	

These I/O modules occupy 2x3 bytes. Object 0x2600 is used for 3-byte specialty module inputs. One sub-index is occupied per channel.

Table 217: Counter Modules 750-638, 753-638

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0	Control byte of counter 1
	1	D0	Counter value of counter 1
	2	D1	
n+1	3	C1	Status byte of counter 2
	4	D2	Counter value of counter 2
	5	D3	

These I/O modules occupy 2x3 bytes. Object 0x2700 is used for 3-byte specialty module outputs. One sub-index is occupied per channel.

3-Phase Power Measurement Modules

750-493

In the input and output process image, these 3-phase power measurement modules occupy 9 bytes of user data, 6 data bytes and two additional control/status bytes. 12 bytes are occupied in the process image.

Table 218: 3-Phase Power Measurement Modules 750-493

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0/S0	Control/status byte of channel 1
	1	-	Empty byte
	2	D0	Counter value of channel 1
	3	D1	Counter value of channel 1
n+1	4	C1/S1	Control/status byte of channel 2
	5	-	Empty byte
	6	D2	Counter value of channel 2
	7	D3	Counter value of channel 2
n+2	8	C2/S2	Control/status byte of channel 3
	9	-	Empty byte
	10	D4	Counter value of channel 3
	11	D5	Counter value of channel 3

These I/O modules occupy 4x4 bytes. Object 0x2800 is used for 4-byte specialty module inputs. Object 0x2900 is used for 4-byte specialty module outputs. One sub-index is occupied per channel.

750-494, -495

In the input and output process image, the 3-phase power measurement modules 750-494 and 495 occupy 24 bytes of user data, 16 data bytes and 8 additional control/status bytes. 24 bytes are occupied in the process image.

Table 219: 3-Phase Power Measurement Modules 750-494, -495

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S0	Status word
n+1	1	S1	
n+2	2	S2	Expanded status word 1
n+3	3	S3	
n+4	4	S4	Expanded status word 2
n+5	5	S5	
n+6	6	S6	Expanded status word 3
n+7	7	S7	
n+8	8	D0	Process value 1
n+9	9	D1	
n+10	10	D2	
n+11	11	D3	
n+12	12	D4	Process value 2
n+13	13	D5	
n+14	14	D6	
n+15	15	D7	
n+16	16	D8	Process value 3
n+17	17	D9	
n+18	18	D10	
n+19	19	D11	
n+20	20	D12	Process value 4
n+21	21	D13	
n+22	22	D14	
n+23	23	D15	

Table 220: 3-Phase Power Measurement Modules 750-494, -495

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0	Control word
n+1	1	C1	
n+2	2	C2	Expanded control word 1
n+3	3	C3	
n+4	4	C4	Expanded control word 2
n+5	5	C5	
n+6	6	C6	Expanded control word 3
n+7	7	C7	
n+8	8	D0	Not used
n+9	9	D1	
n+10	10	D2	
n+11	11	D3	
n+12	12	D4	
n+13	13	D5	
n+14	14	D6	
n+15	15	D7	
n+16	16	D8	
n+17	17	D9	
n+18	18	D10	
n+19	19	D11	
n+20	20	D12	
n+21	21	D13	
n+22	22	D14	
n+23	23	D15	

These I/O modules occupy 24 bytes. Object 0x380n is used for 9+ byte specialty module inputs. Object 0x390n is used for 9+ byte specialty module outputs. One I/O module is mapped per object. Each data byte is assigned to a sub-index.

Pulse Width Modules

750-511, (and all variants / xxx-xxx)

In the input and output process image, pulse width modules occupy 6 bytes of user data, 4 data bytes and two additional control/status bytes. 6 bytes are occupied in the process image.

Table 221: Pulse width Modules 750-511 / xxx-xxx

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0/S0	Control/status byte of channel 1
	1	D0	Data value of channel 1
	2	D1	
n+1	3	C1/S1	Control/status byte of channel 2
	4	D2	Data value of channel 2
	5	D3	

These I/O modules occupy 2x3 bytes. Object 0x2600 is used for 3-byte specialty module inputs. Object 0x2700 is used for 3-byte specialty module outputs. One sub-index is occupied per channel.

Serial Interfaces with Alternative Data Format

750-650, (and the variants /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (and the variants /000-001, -002, -003),
750-653, (and the variants /000-002, -007)

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. The structure of the process image of this I/O module then depends on which operating mode is set.

In the input and output process image, the I/O modules with serial interface that are set to the alternative data format occupy 4 bytes of user data, 3 data bytes and 1 additional control/status byte. 4 bytes are occupied in the process image.

Table 222: Serial Interfaces with Alternative Data Format

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C/S	Control/status byte
	1	D0	Data bytes
n+1	2	D1	
	3	D2	

These I/O modules occupy 2x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. Object 0x6411 (0x2500 also possible) is used for 2-byte specialty module outputs. One sub-index is occupied per channel.

Serial Interfaces with Standard Data Format

750-650/000-001, -014, -015, -016
750-653/000-001, -006

In the input and output process image, the I/O modules with serial interface set to the standard data format occupy 6 bytes of user data, 5 data bytes and 1 additional control/status byte. 6 bytes are occupied in the process image.

Table 223: Serial Interface with Standard Data Format

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C/S	Control/status byte
	1	D0	Data bytes
	2	D1	
	3	D2	
	4	D3	
	5	D4	

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per I/O module.

KNX/EIB/TP1 Module

753-646

In the input and output process image, the KNX/TP1 module occupies 24 bytes of user data in router and device mode, 20 data bytes and 1 control/status byte. Even though the additional bytes S1 or C1 are transferred as data bytes, they are used as extended status and control bytes. The opcode is used for the data read/write command and for triggering specific functions of the KNX/EIB/TP1 module.

Access to the process image is not possible in router mode. Telegrams can only be tunneled. In device mode, access to the KNX data can only be performed via special function blocks of the IEC application. Configuration using the ETS engineering tool software is not required for KNX.

Table 224: Input/Output Process Image of the KNX/EIB/TP1-Module

Input/Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	-	Not used
n+1	1	C0/S0	Control/status byte
n+2	2	C1/S1	Additional control/status byte
n+3	3	OP	Opcode
n+4	4	D0	Data byte 0
…	…	…	…
n+23	23	D19	Data byte 19

These I/O modules occupy 24 bytes. Object 0x380n is used for 9+ byte specialty module inputs. Object 0x390n is used for 9+ byte specialty module outputs.
One I/O module is mapped per object. Each data byte is assigned to a sub-index.

Serial Interface RS-232 / RS-485

750-652

Serial Transmission Mode

The data to be sent and received is stored in up to 46 input and output bytes. The data flow is controlled with the control/status byte. The input bytes form the memory area for up to 46 characters, which were received by the interface. The characters to be sent are passed in the output bytes.

Table 225: Input/Output Process Image, Serial Interface, Serial Transmission Mode

Input/Output Process Image				
Sub-Index			**Byte designation**	**Remark**
n	0	8 bytes	S0/C0	Control/status byte S0
	1		S1/C1	Control/status byte S1
	2		D0	Data byte 0
	3		D1	Data byte 1
	4		D2	Data byte 2
	…		…	…
	7		D5	Data byte 5
n+8	8	24 bytes	D6	Data byte 6
…	…		…	…
n+23	23		D21	Data byte 21
n+24	24	48 bytes	D22	Data byte 22
…	…		…	…
n+47	47		D45	Data byte 45

Data Exchange Mode

The data to be sent and received is stored in up to 47 input and output bytes. The data flow is controlled with the control/status byte.

Table 226: Input/Output Process Image, Serial Interface, Data Exchange Mode

Input/Output Process Image				
Sub-Index	**Offset**		**Byte designation**	**Remark**
n	0		S0/C0	Control/status byte S0
	1		D0	Data byte 0
	2		D1	Data byte 1
	3		D2	Data byte 2
	…		…	…
	7	8 bytes	D6	Data byte 6
n+8	8		D7	Data byte 7
…	…		…	…
n+23	23	24 bytes	D22	Data byte 22
n+24	24		D23	Data byte 23
…	…		…	…
n+47	47	48 bytes	D46	Data byte 46

These I/O modules occupy 1x8, 1x24 or 1x48 bytes. Object 0x3600 is used for 8-byte specialty module inputs. Object 0x3700 is used for specialty module outputs. One sub-index is occupied per I/O module. For 24 bytes or 48 bytes, object 0x380n is used for 9+ byte specialty module inputs and object 0x390n for 9+ byte specialty module outputs. One I/O module is mapped per object. Each data byte is assigned to one sub-index.

Data Exchange Module

750-654 (and variant /000-001)

In the input and output process image, data exchange modules occupy 4 data bytes. 4 bytes are occupied in the process image.

Table 227: Data Exchange Modules

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Data bytes
	1	D1	
n+1	2	D2	
	3	D3	

These I/O modules occupy 2x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. Object 0x6411 (0x2500 also possible) is used for 2-byte specialty module outputs. One sub-index is occupied per channel.

SSI Transmitter Interface Modules with Alternative Data Format

750-630 (and all variants)

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. The structure of the process image of this I/O module then depends on which operating mode is set.

In the input process image, SSI transmitter interface modules with status occupy 4 data bytes. Two words are assigned in the process image via word alignment.

Table 228: SSI Transmitter Interface Modules with Alternative Data Format

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	D0	Data bytes
	1	D1	
n+1	2	D2	
	3	D3	

These I/O modules occupy 2x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. One sub-index is occupied per channel.

SSI Transmitter Interface Modules with Standard Data Format

750-630/000-004, -005, -007

In the input process image, SSI transmitter interface modules with status occupy 4 data bytes, 4 data bytes and 1 additional status byte. 6 bytes are occupied in the process image.

Table 229: SSI Transmitter Interface Modules with Standard Data Format

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	-	Not used
	2	D0	Data bytes
	3	D1	
	4	D2	
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. One sub-index is occupied per channel.

Distance and Angle Measurement Modules

750-631

The I/O module occupies 5 bytes in the input process image and 3 bytes in the output process image. 6 bytes are occupied in the process image.

Table 230: Distance and Angle Measurement Modules

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	D0	Counter word
	2	D1	
	3	-	Not used
	4	D2	Latch word
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. One sub-index is occupied per channel.

Table 231: Distance and Angle Measurement Modules

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C	control byte
	1	D0	Counter word
	2	D1	
	3	-	Not used
	4	-	
	5	-	

These I/O modules occupy 1x6 bytes. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per channel.

750-634

The I/O module 750-634 occupies 5 bytes in the input process image, or 6 bytes in cycle duration measurement operating mode, and 3 bytes in the output process image. 6 bytes are occupied in the process image.

Table 232: Incremental Encoder Interface 750-634

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	D0	Counter word
	2	D1	
	3	D2*)	(Cycle duration)
	4	D3	Latch word
	5	D4	

*) If the control byte sets the operating mode to cycle duration measurement, D2 together with D3/D4 provides a 24-bit value for the cycle duration.

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. One sub-index is occupied per channel.

Table 233: Incremental Encoder Interface 750-634

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C	Status byte
	1	D0	Counter word
	2	D1	
	3	-	Not used
	4	-	
	5	-	

These I/O modules occupy 1x6 bytes. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per channel.

750-637

The incremental encoder interface module occupies 6 bytes of user data in the input and output area of the process image, 4 data bytes and two additional control/status bytes. 6 bytes are occupied in the process image.

Table 234:Incremental Encoder Interface 750-637

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0/S0	Control/status byte 1
	1	D0	Data values
	2	D1	
n+1	3	C1/S1	Control/status byte 2
	4	D2	Data values
	5	D3	

These I/O modules occupy 2x3 bytes. Object 0x2600 is used for 3-byte specialty module inputs. Object 0x2700 is used for 3-byte specialty module outputs. Two sub-indices are occupied per I/O module.

750-635,
753-635

In the input and output process image, the digital pulse interface module occupies 4 bytes, 3 data bytes and 1 additional control/status byte. 4 bytes are occupied in the process image.

Table 235: Digital Impulse Interface 750-635

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0/S0	Control/status byte
	1	D0	Data values
	2	D1	
	3	D2	

These I/O modules occupy 1x4 bytes. Object 0x2800 is used for 4-byte specialty module inputs. Object 0x2900 is used for 4-byte specialty module outputs. One sub-index is occupied per I/O module.

RTC Module

750-640

In the input and output process image, the RTC module occupies 6 bytes of user data, 4 data bytes, 1 additional control/status byte, as well as 1 command byte (ID) each. 6 bytes are occupied in the process image.

Table 236: RTC Module 750-640

Input and Output Process Image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C/S	Control/status byte
	1	ID	Command byte
	2	D0	Data bytes
	3	D1	
	4	D2	
	5	D3	

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per I/O module.

Stepper Modules

750-670, -671, -672, -673

The stepper module makes a 12-byte input/output process image available.

The data to be sent and received is stored in up to 7 input/output bytes depending on the operating mode. If the mailbox is activated, the first 6 data bytes are overlaid with mailbox data.

Table 237: Input Process Image, Stepper Module with Mailbox Deactivated

Input/Output Process Image			
Sub-Index	Offset	Byte designation	Remark
n	0	C0/S0	Control/status byte
	1	-	Reserved
	2	D0	Data bytes
	3	D1	
	4	D2	
	5	D3	
	6	D4	
	7	D5	
	8	D6	
	9	C3/S3	Control/status byte
	10	C2/S2	Control/status byte
	11	C1/S1	Control/status byte

Table 238: Output Process Image, Stepper Module with Mailbox Activated

Input/Output Process Image			
Sub-Index	Offset	Byte designation	Remark
n	0	C0/S0	Control/status byte
	1	-	Reserved
	2	MBX0	Mailbox bytes (mailbox activated)
	3	MBX1	
	4	MBX2	
	5	MBX3	
	6	MBX4	
	7	MBX5	
	8	-	Reserved
	9	C3/S3	Control/status byte
	10	C2/S2	Control/status byte
	11	C1/S1	Control/status byte

These I/O modules occupy 1x12 bytes. Object 0x4200 is used for gateway module inputs. Object 0x4300 is used for gateway module outputs. One sub-index is occupied per I/O module.

DALI/DSI Master Module

750-641

In the input and output process image, the DALI/DSI master module occupies 6 data bytes, 5 data bytes and 1 additional control/status byte. 6 bytes are occupied in the process image.

Table 239: DALI/DSI Master Module 750-641

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	D0	DALI response
	2	D1	DALI address
	3	D2	Message 3
	4	D3	Message 2
	5	D4	Message 1

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. One sub-index is occupied per channel.

Table 240: DALI/DSI Master Module 750-641

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C	control byte
	1	D0	DALI command, DSI dimming value
	2	D1	DALI address
	3	D2	Parameter 2
	4	D3	Parameter 1
	5	D4	Command extension

These I/O modules occupy 1x6 bytes. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per channel.

DALI-Multi-Master Module

753-647

The DALI Multi-Master module occupies a total of 24 bytes in the input and output 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 ECGs, groups or scenes to the 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.

The structure of the process data is described in detail in the following tables.

Table 241: Overview of Input Process Image in the “Easy” Mode

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status, activate broadcast Bit 0: 1-/2-button mode Bit 2: Broadcasts status ON/OFF Bit 1,3-7: -
n+1	1	-	res.
n+2	2	DA0…DA3	
n+3	3	DA4…DA7	
n+4	4	DA8…DA11	
n+5	5	DA12…DA15	
n+6	6	DA16…DA19	
n+7	7	DA20…DA23	
n+8	8	DA24…DA27	Bit pair for DALI address DA0:
n+9	9	DA28…DA31	Bit 1: Bit set = ON
n+10	10	DA32…DA35	Bit not set = OFF
n+11	11	DA36…DA39	Bit 2: Bit set = Error
n+12	12	DA40…DA43	Bit not set = No error
n+13	13	DA44…DA47	Bit pairs DA1 to DA63 similar to DA0.
n+14	14	DA48…DA51	
n+15	15	DA52…DA55	
n+16	16	DA56…DA59	
n+17	17	DA60…DA63	
n+18	18	GA0…GA3	Bit pair for DALI group address GA0:
n+19	19	GA4…GA7	Bit 1: Bit set = ON
n+20	20	GA8…GA11	Bit not set = OFF
n+21	21	GA12…GA15	Bit 2: Bit set = Error
n+22	22		Bit not set = No error
n+23	23		Bit pairs GA1 to GA15 similar to GA0.
n+24	24	-	Not used
n+25	25	-	

DA = DALI address
GA = Group address

Table 242: Overview of the Output Process Image in the “Easy” mode

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	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 Bit 4…7: reserved
n+1	1	-	res.
n+2	2	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 to DA63 similar to DA0.
n+3	3	DA4…DA7	
n+4	4	DA8…DA11	
n+5	5	DA12…DA15	
n+6	6	DA16…DA19	
n+7	7	DA20…DA23	
n+8	8	DA24…DA27	
n+9	9	DA28…DA31	
n+10	10	DA32…DA35	
n+11	11	DA36…DA39	
n+12	12	DA40…DA43	
n+13	13	DA44…DA47	
n+14	14	DA48…DA51	
n+15	15	DA52…DA55	
n+16	16	DA56…DA59	
n+17	17	DA60…DA63	
n+18	18	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 to GA15 similar to GA0.
n+19	19	GA4…GA7	
n+20	20	GA8…GA11	
n+21	21	GA12…GA15	
n+22	22		
n+23	23		
n+24	24	Bit 0…7	Switch to scene 0…15
n+25	25	Bit 8…15	

DA = DALI address
GA = Group address

These I/O modules occupy 25 bytes. Object 0x380n is used for 9+ byte specialty module inputs. Object 0x390n is used for 9+ byte specialty module outputs. One I/O module is mapped per object. Each data byte is assigned to a sub-index.

LON® FTT Module

753-648

The process image of the LON® FTT module consists of a control/status byte and 23 bytes of bidirectional communication data that is processed by the WAGO-I/O-*PRO* function block "LON_01.lib". This block is required for the function of the LON® FTT module and makes a user interface available on the control side.

EnOcean Radio Receiver I/O Module

750-642

In the input and output process image, the EnOcean radio receiver module occupies 4 bytes of user data, 3 data bytes and 1 additional control/status byte. However, the 3 bytes of output data are not used. 4 bytes are occupied in the process image.

Table 243: EnOcean 750-642 Radio Receiver I/O Module

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S	Status byte
	1	D0	Data bytes
n+1	2	D1	
	3	D2	

Table 244: EnOcean 750-642 Radio Receiver I/O Module

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C	control byte
	1	-	Not used
n+1	2	-	
	3	-	

These I/O modules occupy 2x2 bytes. Object 0x6401 (0x2400 also possible) is used for 2-byte specialty module inputs. Object 0x6411 (0x2500 also possible) is used for 2-byte specialty module outputs. One sub-index is occupied per channel.

Bluetooth® RF Transceiver

750-644

The size of the process image for the *Bluetooth*® I/O module can be set at a fixed size of 12, 24 or 48 bytes. It consists of one control byte (input) or one status byte (output), one empty byte, one 6-, 12- or 18-byte overlayable mailbox (mode 2) and the *Bluetooth*® process data with a size of 4 to 46 bytes.

The *Bluetooth*® I/O module uses between 12 to 48 bytes in the process image. The size 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*® RF Transceivers 750-644.

Table 245: *Bluetooth*® RF Transceiver 750-644

Input and Output Process Image			
Process image size	**12 bytes**	**24 bytes**	**48 bytes**
n PDO	1 status/ control byte 1 empty byte 6 bytes mailbox or 6 bytes process data	1 status/ control byte 1 empty byte 6 bytes mailbox or 6 bytes process data	1 status/ control byte 1 empty byte 6 bytes mailbox or 6 bytes process data
n+1 PDO	4 bytes process data 4 bytes empty (reserved)	8 bytes process data	8 bytes process data
n+2 PDO	free for next I/O module	8 bytes process data	8 bytes process data
n+3 PDO	-	free for next I/O module	8 bytes process data
n+4 PDO	-	-	2 bytes process data
n+5 PDO	-	-	8 bytes process data
n+6 PDO	-	-	free for next I/O module

These I/O modules appear as follows depending on the data width set:

Data width	**Object**
1x12 Byte Gateway 1 Input	0x4200
1x12 Byte Gateway 1 Output	0x4300
1x24 Byte Gateway 1 Input	0x4200
1x24 Byte Gateway 1 Output	0x4300
1x48 Byte Gateway 1 Input	0x4200
1x48 Byte Gateway 1 Output	0x4300

One sub-index is occupied per channel.

MP Bus Master Module

750-643

In the input and process image, the MP Bus Master module occupies 8 bytes of user data, 6 data bytes and two additional control/status bytes. 8 bytes are occupied in the process image.

Table 246: MP Bus Master Module 750-643

Input and Output Process Image			
Sub-Index	Offset	Byte designation	Remark
n	0	C0/S0	Control/status byte
	1	C1/S1	Additional control/status byte
	2	D0	Data bytes
	3	D1	
	4	D2	
	5	D3	
	6	D4	
	7	D5	

These I/O modules occupy 1x8 bytes. Object 0x3600 is used for 8-byte specialty module inputs. Object 0x3700 is used for 8-byte specialty module outputs. One sub-index is occupied per I/O module.

Vibration Velocity/Bearing Condition Monitoring VIB I/O Module

750-645

In the input and the output process image, the vibration velocity/bearing condition monitoring VIB I/O module occupies 12 bytes of user data, 8 data bytes and 4 additional control/status bytes. 12 bytes are occupied in the process image.

Table 247: Vibration Velocity/Bearing Condition Monitoring VIB I/O 750-645

Input and Output Process Image			
Sub-Index	Offset	Byte designation	Remark
n	0	C0/S0	Control/status byte (log. channel 1, sensor input 1)
	1	D0	Data bytes (log. channel 1, sensor input 1)
	2	D1	
n+1	3	C1/S1	Control/status byte (log. channel 2, sensor input 2)
	4	D2	Data bytes (log. channel 2, sensor input 2)
	5	D3	
n+2	6	C2/S2	Control/status byte (log. channel 3, sensor input 1)
	7	D4	Data bytes (log. channel 3, sensor input 1)
	8	D5	
n+3	9	C3/S3	Control/status byte (log. channel 4, sensor input 2)
	10	D6	Data bytes (log. channel 4, sensor input 2)
	11	D7	

These I/O modules occupy 4x3 bytes. Object 0x2600 is used for 3-byte specialty module inputs. Object 0x2700 is used for 3-byte specialty module outputs. One sub-index is occupied per logical channel.

DC Drive Controller

750-636

The I/O module occupies 6 bytes of input and output data in the process image. The position data to be sent and received is stored in 4 output bytes and 4 input bytes. 2 control/status bytes are used to control the I/O module and drive. In addition to the position data in the input process image, extended status information can be shown.

Table 248: DC Drive Controller Input Process Image

<table>
<tr><th colspan="6">Input process image</th></tr>
<tr><th>Sub-Index</th><th>Offset</th><th colspan="2">Byte designation</th><th colspan="2">Remark</th></tr>
<tr><td rowspan="6">n</td><td>0</td><td colspan="2">S0</td><td colspan="2">Status byte S0</td></tr>
<tr><td>1</td><td colspan="2">S1</td><td colspan="2">Status byte S1</td></tr>
<tr><td>2</td><td>D0</td><td>S2</td><td>Actual position (LSB)</td><td>Ext. status byte S2</td></tr>
<tr><td>3</td><td>D1</td><td>S3</td><td>Actual position</td><td>Ext. status byte S3</td></tr>
<tr><td>4</td><td>D2</td><td>S4</td><td>Actual position</td><td>Ext. status byte S4</td></tr>
<tr><td>5</td><td>D3</td><td>S5</td><td>Actual position (MSB)</td><td>Ext. status byte S5</td></tr>
</table>

Table 249: DC Drive Controller Output Process Image

<table>
<tr><th colspan="4">Output process image</th></tr>
<tr><th>Sub-Index</th><th>Offset</th><th>Byte designation</th><th>Remark</th></tr>
<tr><td rowspan="6">n</td><td>0</td><td>C0</td><td>Control byte C0</td></tr>
<tr><td>1</td><td>C1</td><td>Control byte C1</td></tr>
<tr><td>2</td><td>D0</td><td>Setpoint position (LSB)</td></tr>
<tr><td>3</td><td>D1</td><td>Setpoint position</td></tr>
<tr><td>4</td><td>D2</td><td>Setpoint position</td></tr>
<tr><td>5</td><td>D3</td><td>Setpoint position (MSB)</td></tr>
</table>

These I/O modules occupy 1x6 bytes. Object 0x3200 is used for 6-byte specialty module inputs. Object 0x3300 is used for 6-byte specialty module outputs. One sub-index is occupied per I/O module.

4-Channel IO-Link Master

750-657

In the input and output process image, the I/O module 750-657 occupies 24 bytes of user data, 20 data bytes and 4 additional control/status bytes, mailbox bytes and SIO bytes.

Table 250: 4-Channel IO-Link-Master Input/Output Process Image

Input/Output Process Image					
Sub-Index	**Offset**		**Byte designation**	**Remark**	
n	0		S0/C0	Control/status byte	
	1		FC0	Acyclic channel	Register byte 0
	2		MB0	Mailbox byte	Register byte 1
	3	4 bytes	SIO	SIO Byte	
	4		D0	Data byte 0	
	5	6 bytes	D1	Data byte 1	
	6		D2	Data byte 2	
	7	8 bytes	D3	Data byte 3	
n+8	8		D4	Data byte 4	
n+9	9	10 bytes	D5	Data byte 5	
n+10	10		D6	Data byte 6	
n+11	11	12 bytes	D7	Data byte 7	
n+12	12		D8	Data byte 8	
n+13	13		D9	Data byte 9	
n+14	14		D10	Data byte 10	
n+15	15	16 bytes	D11	Data byte 11	
n+16	16		D12	Data byte 12	
n+17	17		D13	Data byte 13	
n+18	18		D14	Data byte 14	
n+19	19	20 bytes	D15	Data byte 15	
n+20	20		D16	Data byte 16	
n+21	21		D17	Data byte 17	
n+22	22		D18	Data byte 18	
n+23	23	24 bytes	D19	Data byte 19	

These I/O modules appear as follows depending on the data width set:

Data width	Object	Sub-Index
1x4 bytes of input data	0x2800	One sub-index is occupied per I/O module.
1x4 bytes of output data	0x2900	
1x6 bytes of input data	0x3200	
1x6 bytes of output data	0x3300	
1x10/12/16/20/24 bytes of input data	0x380n	One I/O module is mapped per object. Each data byte is assigned to a sub-index.
1x10/12/16/20/24 bytes of output data	0x390n	

CAN Gateway

750-658

The length of the process image of the CAN Gateway module can adjusted to a fixed size of 8, 12, 16, 20, 24, 32, 40 or 48 bytes.

“Sniffer” and “Transparent” Operating Modes

Table 251: CAN Gateway Input/Output Process Image

Input/Output Process Image				
Sub-Index	**Offset**		**Byte designation**	**Remark**
n	0		S0/C0	Control/status byte
	1		MBX0	Mailbox byte 0
	2		MBX1	Mailbox byte 1
	3		MBX2	Mailbox byte 2
	4		MBX3	Mailbox byte 3
	5		MBX4	Mailbox byte 4
	6		MBX5	Mailbox byte 5
	7	8 bytes	MBX6	Mailbox byte 6
n+8	8		D0	Data byte 0
n+9	9		D1	Data byte 1
n+10	10		D2	Data byte 2
n+11	11	12 bytes	D3	Data byte 3
n+12	12		D4	Data byte 4
n+13	13		D5	Data byte 5
n+14	14		D6	Data byte 6
n+15	15	16 bytes	D7	Data byte 7
n+16	16		D8	Data byte 8
n+17	17		D9	Data byte 9
n+18	18		D10	Data byte 10
n+19	19	20 bytes	D11	Data byte 11
n+20	20		D12	Data byte 12
n+21	21		D13	Data byte 13
n+22	22		D14	Data byte 14
n+23	23	24 bytes	D15	Data byte 15
n+24	24		D16	Data byte 16
…	…		…	…
n+31	31	32 bytes	D23	Data byte 23
n+32	32		D24	Data byte 24
…	…		…	…
n+47	47	48 bytes	D39	Data byte 39

These I/O modules appear as follows depending on the data width set:

Data width	**Object**	**Sub-Index**
1x8 bytes of input data	0x3600	One sub-index is occupied per I/O module.
1x8 bytes of output data	0x3700	
1x12/16/20/24/32/40/48 bytes of input data	0x380n	One I/O module is mapped per object. Each data byte is assigned to a sub-index.
1x12/16/20/24/32/40/48 bytes of output data	0x390n	

"Mapped" Operating Mode

Table 252: CAN Gateway Input/Output Process Image

Input/Output Process Image				
Sub-Index	Offset		Byte designation	Remark
n	0		S0/C0	Control/status byte
	1		MBX0	Mailbox byte 0
	2		MBX1	Mailbox byte 1
	3		MBX2	Mailbox byte 2
	4		MBX3	Mailbox byte 3
	5		MBX4	Mailbox byte 4
	6		MBX5	Mailbox byte 5
	7	8 bytes	MBX6	Mailbox byte 6
n+8	8		T	Toggle bit
n+9	9		D0	Data byte 0
n+10	10		D1	Data byte 1
n+11	11	12 bytes	D2	Data byte 2
n+12	12		D3	Data byte 3
n+13	13		D4	Data byte 4
n+14	14		D5	Data byte 5
n+15	15	16 bytes	D6	Data byte 6
n+16	16		D7	Data byte 7
n+17	17		D8	Data byte 8
n+18	18		D9	Data byte 9
n+19	19	20 bytes	D10	Data byte 10
n+20	20		D11	Data byte 11
n+21	21		D12	Data byte 12
n+22	22		D13	Data byte 13
n+23	23	24 bytes	D14	Data byte 14
n+24	24		D15	Data byte 15
…	…		…	…
n+31	31	32 bytes	D22	Data byte 22
n+32	32		D23	Data byte 23
…	…		…	…
n+47	47	48 bytes	D38	Data byte 38

These I/O modules appear as follows depending on the data width set:

Data width	Object	Sub-Index
1x8 bytes of input data	0x3600	One sub-index is occupied per I/O module.
1x8 bytes of output data	0x3700	
1x12/16/20/24/32/40/48 bytes of input data	0x380n	One I/O module is mapped per object. Each data byte is assigned to a sub-index.
1x12/16/20/24/32/40/48 bytes of output data	0x390n	

Proportional Valve Module

750-632

The Proportional Valve Module occupies 6 bytes in 1-channel mode (1 valve), 12 bytes in 2-channel mode (2 valves).

1-channel mode

Table 253: Proportional Valve Module Input Process Image

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S0	Status byte
	1	MBX_ST	Mailbox status byte
	2	MBX_DATA	Mailbox data
	3	V1_STATUS	Valve 1 control
	4	V1_ACTUAL_L	Valve 1, actual value, low byte
	5	V1_ACTUAL_H	Valve 1, actual value, high byte

Table 254: Proportional Valve Module Output Process Image

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0	control byte
	1	MBX_CTRL	Mailbox control byte
	2	MBX_DATA	Mailbox data
	3	V1_CONTROL	Valve 1 control
	4	V1_SETPOINTVALUE_L	Valve 1, setpoint, low byte
	5	V1_SETPOINTVALUE_H	Valve 1, setpoint, high byte

These I/O modules occupy 1x6 bytes. Object 0x3800 is used for 6-byte specialty module inputs. Object 0x3900 is used for 6-byte specialty module outputs. One sub-index is occupied per I/O module.

2-channel mode

Table 255: Proportional Valve Module Input Process Image

Input process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	S0	Status byte
n+1	1	MBX_ST	Mailbox status byte
n+2	2	MBX_DATA1	Mailbox data
n+3	3	MBX_DATA2	
n+4	4	MBX_DATA3	
n+5	5	MBX_DATA4	
n+6	6	V1_STATUS	Valve 1 control
n+7	7	V2_STATUS	Valve 2 control
n+8	8	V1_ACTUAL_L	Valve 1, actual value, low byte
n+9	9	V1_ACTUAL_H	Valve 1, actual value, low byte
n+10	10	V2_ACTUAL_L	Valve 2, actual value, low byte
n+11	11	V2_ACTUAL_H	Valve 2, actual value, low byte

Table 256: Proportional Valve Module Output Process Image

Output process image			
Sub-Index	**Offset**	**Byte designation**	**Remark**
n	0	C0	control byte
n+1	1	MBX_CTRL	Mailbox control byte
n+2	2	MBX_DATA1	Mailbox data
n+3	3	MBX_DATA2	
n+4	4	MBX_DATA3	
n+5	5	MBX_DATA4	
n+6	6	V1_CONTROL	Valve 1 control
n+7	7	V2_CONTROL	Valve 2 control
n+8	8	V1_SETPOINTVALUE_L	Valve 1, setpoint, low byte
n+9	9	V1_SETPOINTVALUE_H	Valve 1, setpoint, high byte
n+10	10	V2_SETPOINTVALUE_L	Valve 2, setpoint, low byte
n+11	11	V2_SETPOINTVALUE_H	Valve 2, setpoint, high byte

These I/O modules occupy 12 bytes. Object 0x380n is used for 9+ byte specialty module inputs. Object 0x390n is used for 9+ byte specialty module outputs. One I/O module is mapped per object. Each data byte is assigned to a sub-index.

AS-Interface Master

750-655

The maximum process image of the AS-Interface Master Module is 48 bytes. The mailbox size is limited to 6 bytes when using a CANopen fieldbus coupler/controller.

Each PDO can take up to 8 bytes of data.

The first PDO assigned to an AS-Interface Master Module contains the status/control byte, an empty byte and up to 6 bytes for mailbox or process data.

The following PDOs contain AS-Interface process data. The subsequent table shows assignment of the process image size to the number of occupied PDOs when the mailbox is shown permanently (mode 1).

Table 257: AS-Interface Master, Number of Occupied PDOs when the Mailbox Is Visible Permanently

	Input and Output Process Image					
Process image size	**12 bytes**	**20 bytes**	**24 bytes**	**32 bytes**	**40 bytes**	**48 bytes**
PDO #n	1 status/ Control byte 1 empty byte 6 bytes Mailbox	1 status/ Control byte 1 empty byte 6 bytes Mailbox	1 status/ Control byte 1 empty byte 6 bytes Mailbox	1 status/ Control byte 1 empty byte 6 bytes Mailbox	1 status/ Control byte 1 empty byte 6 bytes Mailbox	1 status/ Control byte 1 empty byte 6 bytes Mailbox
PDO n+1	Process data (flags and Slave 1/1A - Slave 7/7A) 4 bytes empty (reserved)	8 bytes Process data (flags and Slave 1/1A – Sl. 15/15A)	8 bytes Process data (flags and Slave 1/1A – Sl. 15/15A)	8 bytes Process data (flags and Slave 1/1A – Sl. 15/15A)	8 bytes Process data (flags and Slave 1/1A – Sl. 15/15A)	8 bytes Process data (flags and Slave 1/1A – Sl. 15/15A)
PDO n+2	free for next module	4 bytes Process data (Sl. 16/16A – Sl. 23/23A) 4 bytes empty (reserved)	8 bytes Process data (Sl. 16/16A – Sl. 31/31A)	8 bytes Process data (Sl. 16/16A – Sl. 31/31A)	8 bytes Process data (Sl. 16/16A – Sl. 31/31A)	8 bytes Process data (Sl. 16/16A – Sl. 31/31A)
PDO n+3		free for next module	free for next module	8 bytes Process data (Slave 1B – Slave 15B)	8 bytes Process data (Slave 1B – Slave 15B)	8 bytes Process data (Slave 1B – Slave 15B)
PDO n+4				free for next module	8 bytes Process data (Slave 16B – Slave 31B)	8 bytes Process data (Slave 16B – Slave 31B)
PDO n+5					free for next module	8 bytes (reserved) *)
PDO n+6						free for next module

*) Free for data from slaves with analog process data with "Auto-populate ON".

The nth PDO corresponds to the first PDO occupied by the AS-Interface Master Module. It contains the status/control byte, an empty byte and up to 6 bytes of mailbox data.

If the length of the permanently visible mailbox is 0 bytes, the nth PDO contains only the status/control byte and an empty byte.
If the process image of the AS-Interface Master Module is 12 or 20 bytes, the last PDO is not fully occupied. Another I/O module then begins with the next PDO.

Note

Process Image Size When Transmitting Process Data

If the mailbox size is 6 bytes and 62 AS-Interface slaves are connected, a process image size of 40 bytes is sufficient to transmit all process data. The I/O module then occupies five PDOs.

In the operating mode where the mailbox can be hidden (mode 2), the following assignment of the process image size to the number of occupied PDOs applies.

Table 258: AS-Interface Master, Number of Occupied PDOs when the Mailbox Is Hidden

Process image size	12 bytes	20 bytes	24 bytes	32 bytes	40 bytes	48 bytes
PDO #n	1 status/control byte 1 empty byte 6 bytes of mailbox or 6 bytes of process data (flags and Slave 1/1A – Sl. 11/11A)	1 status/control byte 1 empty byte 6 bytes of mailbox or 6 bytes of process data (flags and Slave 1/1A – Sl. 11/11A)	1 status/control byte 1 empty byte 6 bytes of mailbox or 6 bytes of process data (flags and Slave 1/1A – Sl. 11/11A)	1 status/control byte 1 empty byte 6 bytes of mailbox or 6 bytes of process data (flags and Slave 1/1A – Sl. 11/11A)	1 status/control byte 1 empty byte 6 bytes of mailbox or 6 bytes of process data (flags and Slave 1/1A – Sl. 11/11A)	1 status/control byte 1 empty byte 6 bytes of mailbox or 6 bytes of process data (flags and Slave 1/1A – Sl. 11/11A)
PDO n+1	4 bytes process data (Sl. 12/12A – Sl. 19/19A) 4 bytes empty (reserved)	8 bytes process data (Sl. – Sl. 27/27A)	8 bytes process data (Sl. – Sl. 27/27A)	8 bytes process data (Sl. – Sl. 27/27A)	8 bytes process data (Sl. – Sl. 27/27A)	8 bytes process data (Sl. – Sl. 27/27A)
PDO n+2	free for next module	4 bytes process data (Sl. 28/28A – Slave 3B) 4 bytes empty (reserved)	8 bytes process data (Sl. – Slave 11B)	8 bytes process data (Sl. – Slave 11B)	8 bytes process data (Sl. – Slave 11B)	8 bytes process data (Sl. – Slave 11B)
PDO n+3		free for next module	free for next module	8 bytes process data (Slave 12B – Slave 27B)	8 bytes process data (Slave 12B – Slave 27B)	8 bytes process data (Slave 12B – Slave 27B)
PDO n+4				free for next module	2 bytes process data (Slave 28B – Slave 31B) 6 bytes empty (reserved)	2 bytes process data (Slave 28B -Slave 31B) 6 bytes empty (reserved)
PDO n+5					free for next module	8 bytes (reserved) *)(re
PDO n+6						free for next module

*) Free for data from slaves with analog process data with "Auto-populate ON".

The nth PDO contains the status/control byte, 1 empty byte and 6 bytes of mailbox data when the mails box is shown or the first 6 bytes of process data. The following PDOs contain the remaining process data.

Note

No access to the first 6 bytes of process data!

When the mailbox is visible, the first 6 bytes of process data (flags and Slave 1/1A to Slave 11/11A) cannot be accessed.

If the process image of the AS-Interface Master Module is 12, 20, 40 or 48 bytes, the last PDO is not fully occupied. Another I/O module then begins with the next PDO.

These I/O modules occupy 1x 12...48 bytes. Object 0x4200-0x4202 is used for gateway module inputs. Object 0x4300-0x4302 is used for gateway module outputs.

One index is occupied per I/O module, where sub-index 1 contains the mailbox size, sub-index 2 the mailbox and sub-index 3 up to 48 the process data.

11.2.6 System Modules

System Modules with Diagnostics

750-610, -611

Power supply modules 750-610 and -611 with diagnostics provide 2 bits to monitor the power supply.

Table 259: 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**
						Diag. bit S 2 Fuse	Diag. bit S 1 Voltage

Object 0x6000 (0x2000 also possible) is used for the digital inputs.

11.2.6.1 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 260: 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

Object 0x6000 (0x2000 also possible) is used for the digital inputs. Object 0x6200 (0x2100 also possible) is used for the digital outputs.

12 Use in Hazardous Environments

The **WAGO-I/O-SYSTEM 750** (electrical equipment) is designed for use in Zone 2 hazardous areas.

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 IEC-Ex

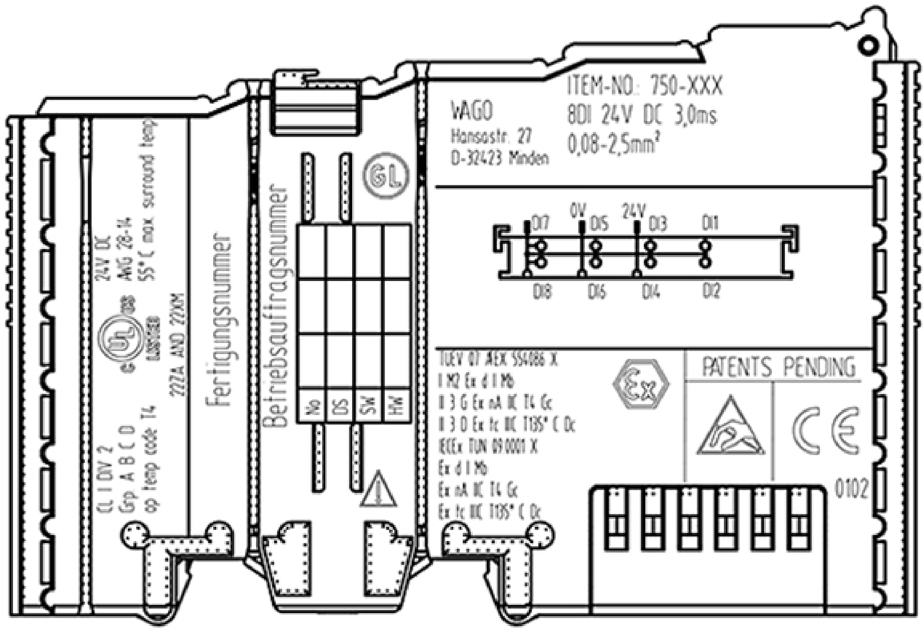


Figure 87: Side Marking Example for Approved I/O Modules According to ATEX and IECEx

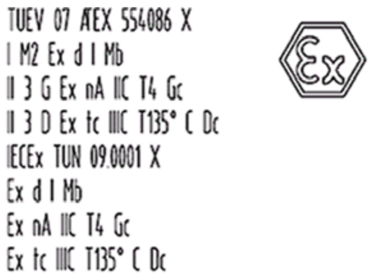


Figure 88: Text Detail – Marking Example for Approved I/O Modules According to ATEX and IECEx.

Table 261: Description of Marking Example for Approved I/O Modules According to ATEX and IECEx

Printing on Text	Description
TÜV 07 ATEX 554086 X IECEx TUN 09.0001 X	Approving authority and certificate numbers
Dust	
II	Equipment group: All except mining
3D	Category 3 (Zone 22)
Ex	Explosion protection mark
tc Dc	Type of protection and equipment protection level (EPL):protection by enclosure
IIIC	Explosion group of dust
T 135°C	Max. surface temperature of the enclosure (without a dust layer)
Mining	
I	Equipment group: Mining
M2	Category: High level of protection
Ex	Explosion protection mark
d Mb	Type of protection and equipment protection level (EPL): Flameproof enclosure
I	Explosion group for electrical equipment for mines susceptible to firedamp
Gases	
II	Equipment group: All except mining
3G	Category 3 (Zone 2)
Ex	Explosion protection mark
nA Gc	Type of protection and equipment protection level (EPL): Non-sparking equipment
nC Gc	Type of protection and equipment protection level (EPL): Sparking apparatus with protected contacts. A device which is so constructed that the external atmosphere cannot gain access to the interior
IIC	Explosion group of gas and vapours
T4	Temperature class: Max. surface temperature 135°C

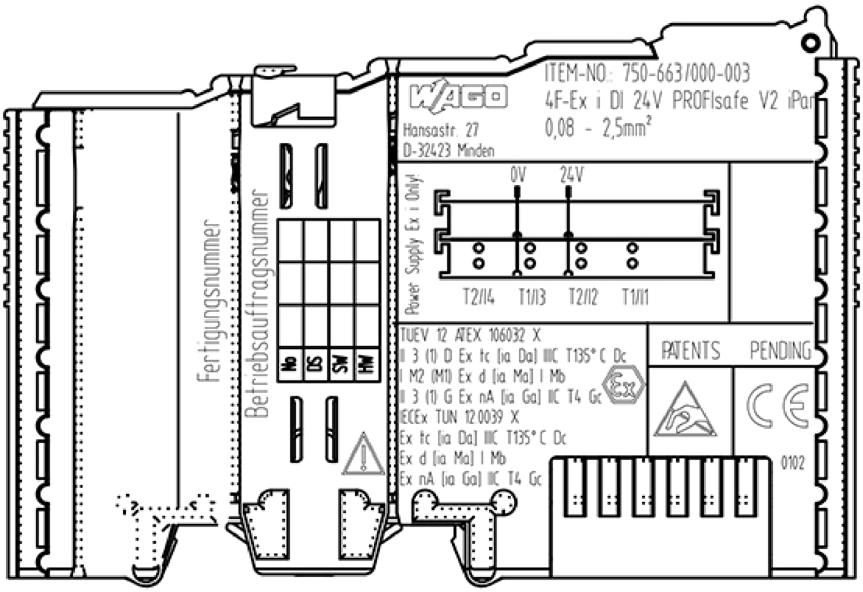


Figure 89: Side Marking Example for Approved Ex i I/O Modules According to ATEX and IECEx.

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 nA [ia Ga] IIC T4 Gc
IECEx TUN 12.0039 X
Ex tc [ia Da] IIIC T135°C Dc
Ex d [ia Ma] I Mb
Ex nA [ia Ga] IIC T4 Gc

Figure 90: Text Detail – Marking Example for Approved Ex i I/O Modules According to ATEX and IECEx.

Table 262: Description of Marking Example for Approved Ex i I/O Modules According to ATEX and IECEx

Inscription Text	Description
TÜV 07 ATEX 554086 X IECEx TUN 09.0001X TÜV 12 ATEX 106032 X IECEx TUN 12.0039 X	Approving authority and 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
3(2)D	Category 3 (Zone 22) equipment containing a safety device for a category 2 (Zone 21) equipment
Ex	Explosion protection mark
tc Dc	Type of protection and equipment protection level (EPL): protection by enclosure
[ia Da]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 20
[ib Db]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 21
IIIC	Explosion group of dust
T 135°C	Max. surface temperature of the enclosure (without a dust layer)
Mining	
I	Equipment Group: Mining
M2 (M1)	Category: High level of protection with electrical circuits which present a very high level of protection
Ex d Mb	Explosion protection mark with Type of protection and equipment protection level (EPL): 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

Table 262: Description of Marking Example for Approved Ex i I/O Modules According to ATEX and IECEx

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
3(2)G	Category 3 (Zone 2) equipment containing a safety device for a category 2 (Zone 1) equipment
Ex	Explosion protection mark
nA Gc	Type of protection and equipment protection level (EPL): Non-sparking equipment
[ia Ga]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 0
[ia Gb]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 1
IIC	Explosion group of gas and vapours
T4	Temperature class: Max. surface temperature 135°C

12.1.2 Marking for America According to NEC 500

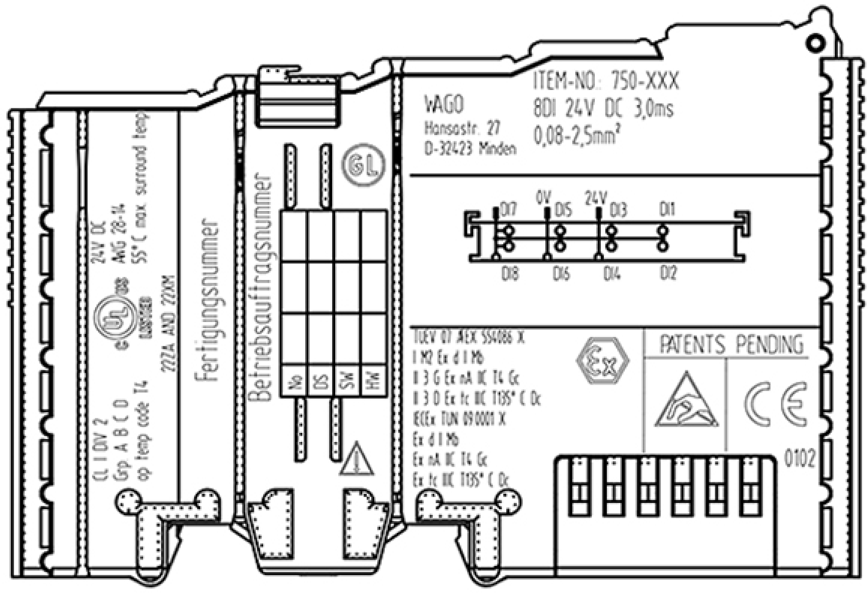


Figure 91: Side Marking Example for I/O Modules According to NEC 500

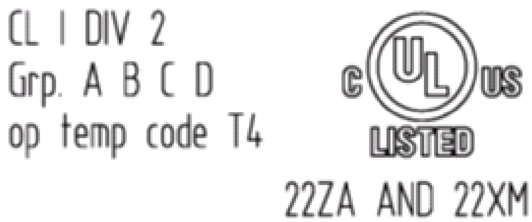


Figure 92: Text Detail – Marking Example for Approved I/O Modules According to NEC 500

Table 263: Description of Marking Example for Approved I/O Modules According to NEC 500

Printing on Text	Description
CL I	Explosion protection group (condition of use category)
DIV 2	Area of application
Grp. ABCD	Explosion group (gas group)
Op temp code T4	Temperature class

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 Conditions for Safe Use (ATEX Certificate TÜV 07 ATEX 554086 X)

1. For use as Gc- or Dc-apparatus (in zone 2 or 22) the Field bus Independent I/O Modules WAGO-I/O-SYSTEM 750-*** shall be erected in an enclosure that fulfils the requirements of the applicable standards (see the marking) EN 60079-0, EN 60079-11, EN 60079-15 and EN 60079-31.
   For use as group I electrical apparatus M2 the apparatus shall be erected in an enclosure that ensures a sufficient protection according to EN 60079-0 and EN 60079-1 and the degree of protection IP64.
   The compliance of these requirements and the correct installation into an enclosure or a control cabinet of the devices shall be certified by an ExNB.
2. Measures have to be taken outside of the device that the rating voltage is not being exceeded of more than 40 % because of transient disturbances.
3. Dip-switches, binary-switches and potentiometers, connected to the module may only be actuated when explosive atmosphere can be excluded.
4. The connecting and disconnecting of the non-intrinsically safe circuits is only permitted during installation, for maintenance or for repair purposes. The temporal coincidence of explosion hazardous atmosphere and installation, maintenance resp. repair purposes shall be excluded.
   This is although and in particular valid for the interfaces “Memory-Card”, “USB”, “Fieldbus connection”, “Configuration and programming interface”, “antenna socket”, “D-Sub”, “DVI-port” and the “Ethernet interface”. These interfaces are not energy limited or intrinsically safe circuits. An operating of those circuits is in the behalf of the operator.
5. For the types 750-606, 750-625/000-001, 750-487/003-000, 750-484 and 750-633 the following shall be considered: The Interface circuits shall be limited to overvoltage category I/II/III (non mains/mains circuits) as defined in EN 60664-1.
6. For replaceable fuses the following shall be considered: Do not remove or replace the fuse when the apparatus is energized.
7. The following warnings shall be placed nearby the unit:
   WARNING – DO NOT REMOVE OR REPLACE FUSE WHEN ENERGIZED
   WARNING – DO NOT SEPARATE WHEN ENERGIZED
   WARNING – SEPARATE ONLY IN A NON-HAZARDOUS AREA

12.2.2 Special Conditions for Safe Use (ATEX Certificate TÜV 12 ATEX 106032 X)

1. For use as Gc- or Dc-apparatus (in zone 2 or 22) the Field bus Independent I/O Modules WAGO-I/O-SYSTEM 750-*** Ex i shall be erected in an enclosure that fulfils the requirements of the applicable standards (see the marking) EN 60079-0, EN 60079-11, EN 60079-15 and EN 60079-31.
   For use as group I electrical apparatus M2 the apparatus shall be erected in an enclosure that ensures a sufficient protection according to EN 60079-0 and EN 60079-1 and the degree of protection IP64.
   The compliance of these requirements and the correct installation into an enclosure or a control cabinet of the devices shall be certified by an ExNB.
2. Measures have to be taken outside of the device that the rating voltage is not being exceeded of more than 40 % because of transient disturbances.
3. The connecting and disconnecting of the non-intrinsically safe circuits is only permitted during installation, for maintenance or for repair purposes. The temporal coincidence of explosion hazardous atmosphere and installation, maintenance resp. repair purposes shall be excluded.
4. For the type the following shall be considered: The Interface circuits shall be limited to overvoltage category I/II/III (non mains/mains circuits) as defined in EN 60664-1.

12.2.3 Special Conditions for Safe Use (IEC-Ex Certificate TUN 09.0001 X)

1. For use as Gc- or Dc-apparatus (in zone 2 or 22) the Field bus Independent I/O Modules WAGO-I/O-SYSTEM 750-*** shall be erected in an enclosure that fulfils the requirements of the applicable standards (see the marking) IEC 60079-0, IEC 60079-11, IEC 60079-15 and IEC 60079-31. For use as group I electrical apparatus M2 the apparatus shall be erected in an enclosure that ensures a sufficient protection according to IEC 60079-0 and IEC 60079-1 and the degree of protection IP64.
   The compliance of these requirements and the correct installation into an enclosure or a control cabinet of the devices shall be certified by an ExCB.
2. Measures have to be taken outside of the device that the rating voltage is not being exceeded of more than 40 % because of transient disturbances.
3. DIP-switches, binary-switches and potentiometers, connected to the module may only be actuated when explosive atmosphere can be excluded.
4. The connecting and disconnecting of the non-intrinsically safe circuits is only permitted during installation, for maintenance or for repair purposes. The temporal coincidence of explosion hazardous atmosphere and installation, maintenance resp. repair purposes shall be excluded.
   This is although and in particular valid for the interfaces “Memory-Card”, “USB”, “Fieldbus connection”, “Configuration and programming interface”, “antenna socket”, “D-Sub”, “DVI-port” and the “Ethernet interface”. These interfaces are not energy limited or intrinsically safe circuits. An operating of those circuits is in the behalf of the operator.
5. For the types 750-606, 750-625/000-001, 750-487/003-000, 750-484 and 750-633 the following shall be considered: The Interface circuits shall be limited to overvoltage category I/II/III (non mains/mains circuits) as defined in IEC 60664-1.
6. For replaceable fuses the following shall be considered: Do not remove or replace the fuse when the apparatus is energized.
7. The following warnings shall be placed nearby the unit:
   WARNING – DO NOT REMOVE OR REPLACE FUSE WHEN ENERGIZED
   WARNING – DO NOT SEPARATE WHEN ENERGIZED
   WARNING – SEPARATE ONLY IN A NON-HAZARDOUS AREA

12.2.4 Special Conditions for Safe Use (IEC-Ex Certificate IECEx TUN 12.0039 X)

1. For use as Gc- or Dc-apparatus (in zone 2 or 22) the Field bus independent I/O Modules WAGO-I/O-SYSTEM 750-*** Ex i shall be erected in an enclosure that fulfils the requirements of the applicable standards (see the marking) IEC 60079-0, IEC 60079-11, IEC 60079-15, IEC 60079-31.
   For use as group I electrical apparatus M2 the apparatus shall be erected in an enclosure that ensures a sufficient protection according to IEC 60079-0 and IEC 60079-1 and the degree of protection IP64.
   The compliance of these requirements and the correct installation into an enclosure or a control cabinet of the devices shall be certified by an ExCB.
2. Measures have to be taken outside of the device that the rating voltage is not being exceeded of more than 40 % because of transient disturbances.
3. The connecting and disconnecting of the non-intrinsically safe circuits is only permitted during installation, for maintenance or for repair purposes. The temporal coincidence of explosion hazardous atmosphere and installation, maintenance resp. repair purposes shall be excluded.
4. For the type the following shall be considered: The Interface circuits shall be limited to overvoltage category I/II/III (non mains/mains circuits) as defined in IEC 60664-1.

12.2.5 Special Conditions for Safe Use According to ANSI/ISA 12.12.01

A. “This equipment is suitable for use in Class I, Division 2, Groups A, B, C, D or non-hazardous locations only.”
B. “This equipment is to be fitted within tool-secured enclosures only.”
C. “WARNING Explosion hazard - substitution of components may impair suitability for Class I, Div. 2.”
D. “WARNING – Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous” has to be placed near each operator accessible connector and fuse holder.
E. When a fuse is provided, the following information shall be provided: “A switch suitable for the location where the equipment is installed shall be provided to remove the power from the fuse.”
F. For devices with EtherCAT/Ethernet connectors “Only for use in LAN, not for connection to telecommunication circuits.”
G. “WARNING - Use Module 750-642 only with antenna module 758-910.”
H. For Couplers/Controllers and Economy bus modules only: The instructions shall contain the following: “The configuration interface Service connector is for temporary connection only. Do not connect or disconnect unless the area is known to be non-hazardous. Connection or disconnection in an explosive atmosphere could result in an explosion.”
I. Modules containing fuses only: “WARNING - Devices containing fuses must not be fitted into circuits subject to over loads, e.g. motor circuits.”
J. Modules containing SD card reader sockets only: “WARNING - Do not connect or disconnect SD-Card while circuit is live unless the area is known to be free of ignitable concentrations of flammable gases or vapors.”

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.

List of Figures

List of Tables

WE INNOVATE!

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Phone: 05 71/8 87 – 0
Fax: 05 71/8 87 – 1 69
E-Mail: info@wago.com
Internet: http://www.wago.com