WAGO-I/O-SYSTEM 750

Modular I/O System

CANopen
750-347, 750-348

Manual

Technical Description,
Installation and
Configuration

750-124/050-002
Version 2002-08-12

WAGO Kontakttechnik GmbH
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 correctness and completeness of this documentation. However, as errors can never be fully excluded we would appreciate any information or ideas at any time.

E-Mail: documentation@wago.com

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

Table of Contents

1 Important Comments

To ensure fast installation and start-up of the units described in this manual, we strongly recommend that the following information and explanations are carefully read and abided by.

1.1 Legal Principles

1.1.1 Copyright

This manual is copyrighted, together with all figures and illustrations contained therein. Any use of this manual which infringes the copyright provisions stipulated herein, is not permitted. Reproduction, translation and electronic and photo-technical archiving and amendments require the written consent of WAGO Kontakttechnik GmbH. Non-observance will entail the right of claims for damages.

WAGO Kontakttechnik GmbH reserves the right to perform modifications allowed by technical progress. In case of grant of a patent or legal protection of utility patents all rights are reserved by WAGO Kontakttechnik GmbH. Products of other manufacturers are always named without referring to patent rights. The existence of such rights can therefore not be ruled out.

1.1.2 Personnel Qualification

The use of the product detailed in this manual is exclusively geared to specialists having qualifications in PLC programming, electrical specialists or persons instructed by electrical specialists who are also familiar with the valid standards. WAGO Kontakttechnik GmbH declines all liability resulting from improper action and damage to WAGO products and third party products due to non-observance of the information contained in this manual.

1.1.3 Intended Use

For each individual application, the components supplied are to work with a dedicated hardware and software configuration. Modifications are only permitted within the framework of the possibilities documented in the manuals. All other changes to the hardware and/or software and the non-conforming use of the components entail the exclusion of liability on part of WAGO Kontakttechnik GmbH.

Please direct any requirements pertaining to a modified and/or new hardware or software configuration directly to WAGO Kontakttechnik GmbH.

1.2 Symbols

Danger
Always abide by this information to protect persons from injury.

Warning
Always abide by this information to prevent damage to the device.

Attention
Marginal conditions must always be observed to ensure smooth operation.

ESD (Electrostatic Discharge)
Warning of damage to the components by electrostatic discharge. Observe the precautionary measure for handling components at risk.

Note
Routines or advice for efficient use of the device and software optimization.

More information
References on additional literature, manuals, data sheets and INTERNET pages

1.3 Font Conventions

Italic

Names of path and files are marked italic
i.e.: *C:\programs\WAGO-IO-CHECK*

Italic

Menu items are marked as bold italic
i.e.: ***Save***

\

A backslash between two names marks a sequence of menu items
i.e.: ***File\New***

END

Press buttons are marked as bold with small capitals
i.e.: **ENTER**

< >

Keys are marked bold within angle brackets
i.e.: **<F5>**

`Courier`

Program code is printed with the font `Courier`.
i.e.: `END_VAR`

1.4 Number Notation

Number Code	Example	Note
Decimal	100	normal notation
Hexadecimal	0x64	C notation
Binary	'100' '0110.0100'	Within ', Nibble separated with dots

1.5 Safety Notes

Attention
Switch off the system prior to working on bus modules!

In the event of deformed contacts, the module in question is to be replaced, as its functionality can no longer be ensured on a long-term basis.

The components are not resistant against materials having seeping and insulating properties. Belonging to this group of materials is: e.g. aerosols, silicones, triglycerides (found in some hand creams).

If it cannot be ruled out that these materials appear in the component environment, then additional measures are to be taken:
- installation of the components into an appropriate housing
- handling of the components only with clean tools and materials.

Attention
Cleaning of soiled contacts may only be done with ethyl alcohol and leather cloths. Thereby, the ESD information is to be regarded.

Do not use any contact spray, as in a worst-case scenario; the functioning of the contact area can be impaired.

The WAGO-I/O-SYSTEM 750 and its components are an open system. It must only be assembled in housings, cabinets or in electrical operation rooms. Access must only be given via a key or tool to authorized qualified personnel.

The relevant valid and applicable standards and guidelines concerning the installation of switch boxes are to be observed.

ESD (Electrostatic Discharge)
The modules are equipped with electronic components that may be destroyed by electrostatic discharge. When handling the modules, ensure that the environment (persons, workplace and packing) is well grounded. Avoid touching conductive components, e.g. gold contacts.

1.6 Scope

This manual describes the modular WAGO-I/O-SYSTEM 750 with the ECO fieldbus coupler for CANopen.

Item no.	Description
750-347	CANopen ECO Fieldbus coupler, MCS; 10 kBaud ... 1 MBaud
750-348	CANopen ECO Fieldbus coupler, D-Sub; 10 kBaud ... 1 MBaud

1.7 Abbreviation

AI	Analog Input
AO	Analog Output
DI	Digital Input
DO	Digital Output
I/O	Input/Output
ID	Identifier
HB	High Byte
HW	Hardware
LB	Low Byte
SW	Software

2 The WAGO-I/O-SYSTEM 750

2.1 System Description

The WAGO-I/O-SYSTEM 750 is a modular, fieldbus independent I/O system. The structure described here consists of an ECO fieldbus coupler (1) and up to 64 connected fieldbus modules (2) for any kind of signal. Together, these make up the fieldbus node. The end module (3) completes the node.

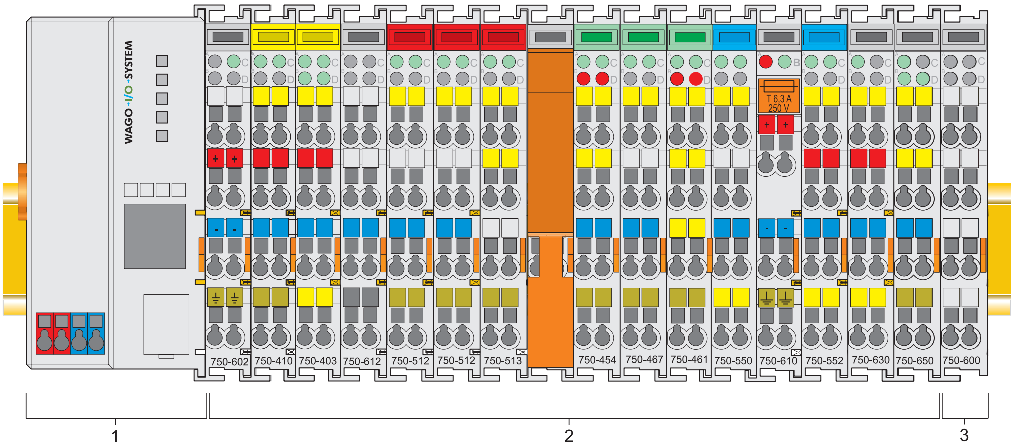


Fig. 2-1: Fieldbus node

g0xxx14x

ECO couplers for fieldbus systems such as PROFIBUS DP, INTERBUS, CANopen and DeviceNet are available.

The ECO coupler contains 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 bus modules and make it available for the fieldbus communication. The 24 V system supply are fed in via the integrated power supply terminal.
The fieldbus coupler communicates via the relevant fieldbus.

At first the ECO coupler is concepted for applications with digital I/O functions. Bus modules for diverse digital and analogue I/O functions and special functions can be connected as well. The communication between the ECO coupler and the bus modules is carried out via an internal bus.

The WAGO-I/O-SYSTEM 750 has a clear port level with LEDs for the status indication, insertable mini WSB markers and pullout group marker carriers. The 3-wire technology supplemented by a ground wire connection allows the direct sensor/actuator wiring.

2.2 Technical Data

Mechanic	
Material	Polycarbonate, Polyamide 6.6
Dimensions Coupler	50 mm x 65* mm x 97 mm
Dimensions I/O module, single	12 mm x 64* mm x 100 mm
Dimensions I/O module, double	24 mm x 64* mm x 100 mm
Installation	on DIN 35 with interlock
modular by	double featherkey-dovetail
Mounting position	any position
Length of entire node	≤ 831 mm
Marking	marking label type 247 and 248 paper marking label 8 x 47 mm
Wire range	
Wire range	CAGE CLAMP® Connection 0,08 mm^2 ... 2,5 mm^2 AWG 28-14 8 – 9 mm Stripped length
Contacts	
Power jumpers contacts	blade/spring contact self-cleaning
Current via power contacts $_{max}$	10 A
Voltage drop at I_{max}	< 1 V/64 modules
Data contacts	slide contact, hard gold plated 1,5µ, self-cleaning
Climatic environmental conditions	
Operating temperature	0 °C ... 55 °C
Storage temperature	-20 °C ... +85 °C
Relative humidity	95 % without condensation
Resistance to harmful substances	acc. to IEC 60068-2-42 and IEC 60068-2-43
Special conditions	Ensure that additional measures for components are taken, which are used in an environment involving: – dust, caustic vapors or gasses – ionization radiation.
Mechanical strength	
Vibration resistance	acc. to IEC 60068-2-6
Shock resistance	acc. to IEC 60068-2-27
Free fall	acc. to IEC 60068-2-32 ≤ 1m (module in original packing)

* from upper edge of DIN 35 rail

Safe electrical isolation			
Air and creepage distance	acc. to IEC 60664-1		
Degree of protection			
Degree of protection	IP 20		
Electromagnetic compatibility*			
Directive	**Test values**	**Strength class**	**Evaluation criteria**
Immunity to interference acc. to EN 50082-2 (96)			
EN 61000-4-2	4kV/8kV	(2/4)	B
EN 61000-4-3	10V/m 80% AM	(3)	A
EN 61000-4-4	2kV	(3/4)	B
EN 61000-4-6	10V/m 80% AM	(3)	A
Emission of interference acc. to EN 50081-2 (94)		**Measuring distance**	**Class**
EN 55011	30 dBµV/m	(30m)	A
	37 dBµV/m		
Emission of interference acc. to EN 50081-1 (93)		**Measuring distance**	**Class**
EN 55022	30 dBµV/m	(10m)	B
	37 dBµV/m		

* Exception: 750-630, 750-631

Range of application	Required specification emission of interference	Required specification immunity to interference
Industrial areas	EN 50081-2 : 1993	EN 50082-2 : 1996
Residential areas	EN 50081-1 : 1993*)	EN 50082-1 : 1992

*) The system meets the requirements on emission of interference in residential areas with the fieldbus coupler/controller for:

ETHERNET	750-342/-842
LONWORKS	750-319/-819
CANopen	750-337/-837
DeviceNet	750-306/-806
MODBUS	750-312/-314/ -315/ -316 750-812/-814/ -815/ -816

With a special permit, the system can also be implemented with other fieldbus couplers/controllers in residential areas (housing, commercial and business areas, small-scale enterprises). The special permit can be obtained from an authority or inspection office. In Germany, the Federal Office for Post and Telecommunications and its branch offices issues the permit.

It is possible to use other couplers/controllers under certain boundary conditions. Please contact WAGO Kontakttechnik GmbH.

Maximum power dissipation of the components	
Bus modules	0,8 W / bus terminal (total power dissipation, system/field)
ECO Fieldbus coupler	2,0 W / coupler

Warning
The power dissipation of all installed components must not exceed the maximal conductible power of the housing (cabinet).

When dimensioning the housing, care is to be taken that even under high external temperatures, the temperature inside the housing does not exceed the permissible ambient temperature of 55 °C.

Dimensions

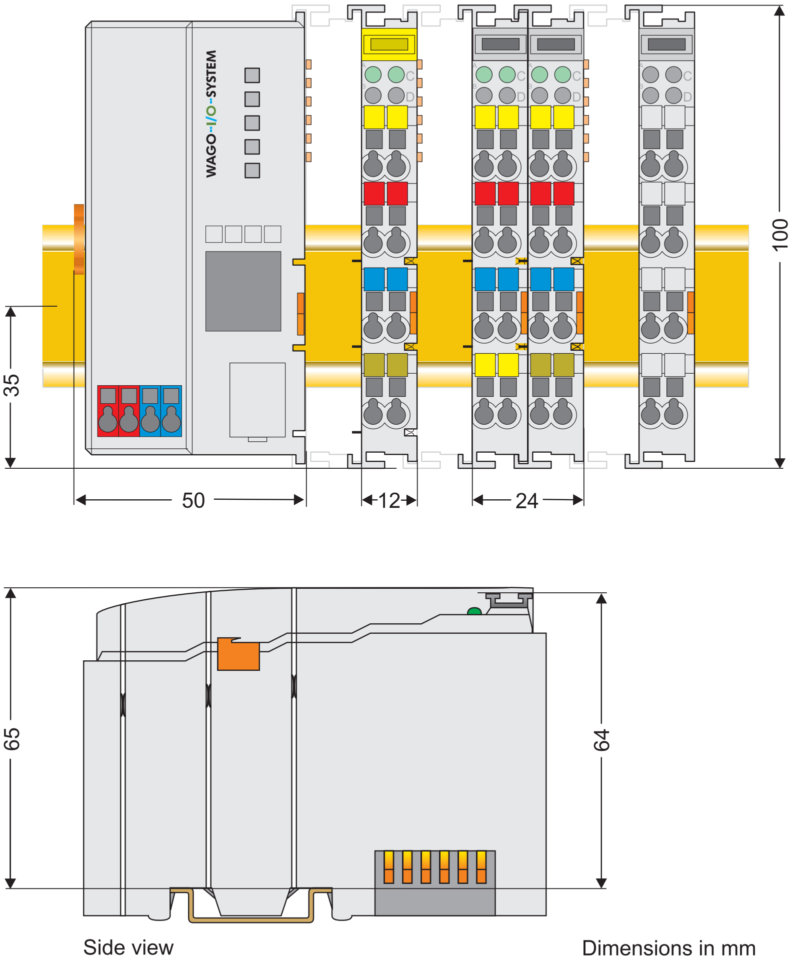


Fig. 2-2: Dimensions

g0xxx15e

2.3 Manufacturing Number

The manufacturing number is part of the lateral marking on the component.

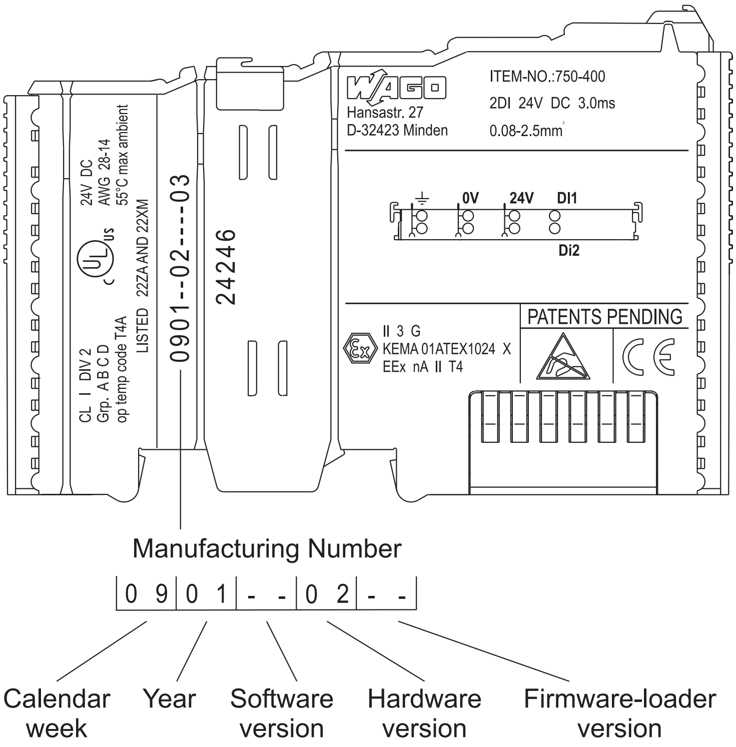


Fig. 2-3: Manufacturing Number

g01xx09e

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.

The number is also printed on the cover of the configuration interface.

2.4 Storage, Consignment and Transport

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

When consigning 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.

Statically shielded transport bags with metal coatings are to be used for the transport of open components for which soiling with amine, amide and silicone has been ruled out, e.g. 3M 1900E.

2.5 Mechanical Setup

2.5.1 Installation Position

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

Attention
In the case of vertical assembly, an end stop has to be mounted as an additional safeguard against slipping.
WAGO item 249-116 End stop for TS 35, 6 mm wide
WAGO item 249-117 End stop for TS 35, 10 mm wide

2.5.2 Total Expansion

The maximum total expansion of a node is calculated as follows:

Quantity	Width	Components
1	50 mm	ECO coupler
64	12 mm	bus modules - inputs / outputs – power supply modules - etc.
1	12 mm	end stop
sum	**830 mm**	

Warning
The maximal total expansion of a node must not exceed 830 mm.

2.5.3 Assembly onto Carrier Rail

2.5.3.1 Carrier Rail Properties

All system components can be snapped directly onto a carrier rail in accordance with the European standard EN 50022 (DIN 35).

Warning
WAGO 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 must take place.

Carrier rails have different mechanical and electrical properties. For the optimal system setup on a carrier rail, certain marginal terms 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 bus module connections.

- A sufficiently stable carrier rail should be selected and, if necessary, several assembly 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 components extends into the profile of the carrier rail. For carrier rails with a height of 7.5 mm, assembly points (screws) are to be riveted under the node in the carrier rail (slotted head captive screws or blind rivets).

2.5.3.2 WAGO DIN Rail

WAGO carrier rails meet the electrical and mechanical requirements.

Item 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

2.5.4 Spacing

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

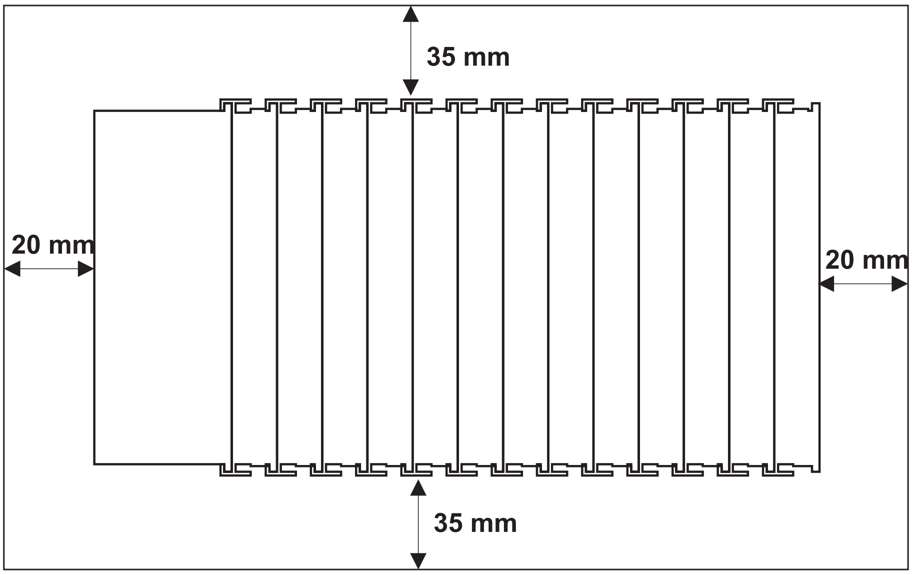


Fig. 2-4: Spacing g01xx13x

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

2.5.5 Plugging and Removal of the Components

Warning
Before work is done on the components, the voltage supply must be turned off.

In order to safeguard the ECO coupler from jamming, it should be fixed onto the carrier rail with the To do so, push on the upper groove of the locking disc using a screwdriver.

To pull out the fieldbus coupler/controller, release the locking disc by pressing on the bottom groove with a screwdriver and then pulling the orange colored unlocking lug.

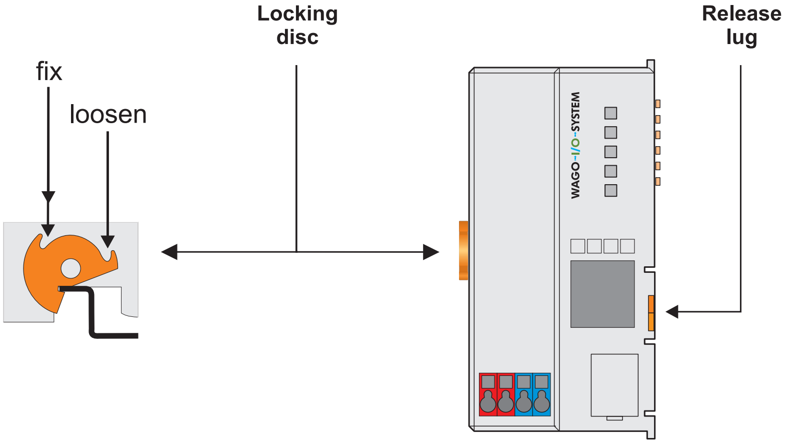


Fig. 2-5: Coupler and unlocking lug g0xxx18e

It is also possible to release an individual I/O module from the unit by pulling an unlocking lug.

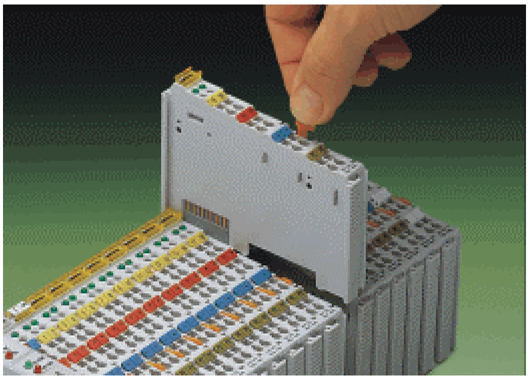


Fig. 2-6: removing bus terminal p0xxx01x

Danger
Ensure that an interruption of the PE will not result in a condition which could endanger a person or equipment!
For planning the ring feeding of the ground wire, please see chapter "Protective Earth".

2.5.6 Assembly Sequence

All system components can be 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 components are securely seated on the rail after installing.

Starting with the ECO coupler, the bus modules are assembled adjacent to each other according to the project planning. Errors in the planning of the node in terms of the potential groups (connection via the power contacts) are recognized, as the bus modules with power contacts (male contacts) cannot be linked to bus modules with fewer power contacts.

⚠ **Attention**
Always link the bus modules with the ECO coupler, always plug from above.

⚠ **Warning**
Never plug bus modules from the direction of the end terminal. A ground wire power contact, which is inserted into a terminal without contacts, e.g. a 4-channel digital input module, has a decreased air and creepage distance to the neighboring contact in the example DI4.

Always terminate the fieldbus node with an end module (750-600).

2.5.7 Internal Bus / Data Contacts

Communication between the ECO coupler and the bus modules as well as the system supply of the bus modules is carried out via the internal bus. It is comprised of 6 data contacts, which are available as self-cleaning gold spring contacts.

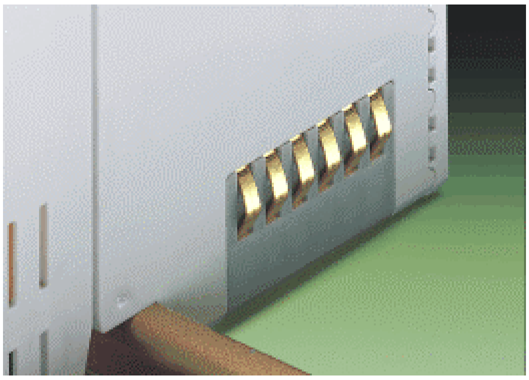


Fig. 2-7: Data contacts p0xxx07x

Warning
Do not touch the gold spring contacts on the I/O Modules in order to avoid soiling or scratches!

ESD (Electrostatic Discharge)
The modules are equipped with electronic components that may be destroyed by electrostatic discharge. When handling the modules, ensure that the environment (persons, workplace and packing) is well grounded. Avoid touching conductive components, e.g. gold contacts.

2.5.8 Power Contacts

Self-cleaning power contacts , are situated on the side of the components which further conduct the supply voltage for the field side. These contacts come as touchproof spring contacts on the right side of the coupler/controller and the bus module. As fitting counterparts the module has male contacts on the left side.

Danger
The power contacts are sharp-edged. Handle the module carefully to prevent injury.

Attention
Please take into consideration that some bus modules have no or only a few power jumper contacts. The design of some modules does not allow them to be physically assembled in rows, as the grooves for the male contacts are closed at the top.

Power jumper contacts

Blade	0	0	3	2
Spring	0	3	3	2

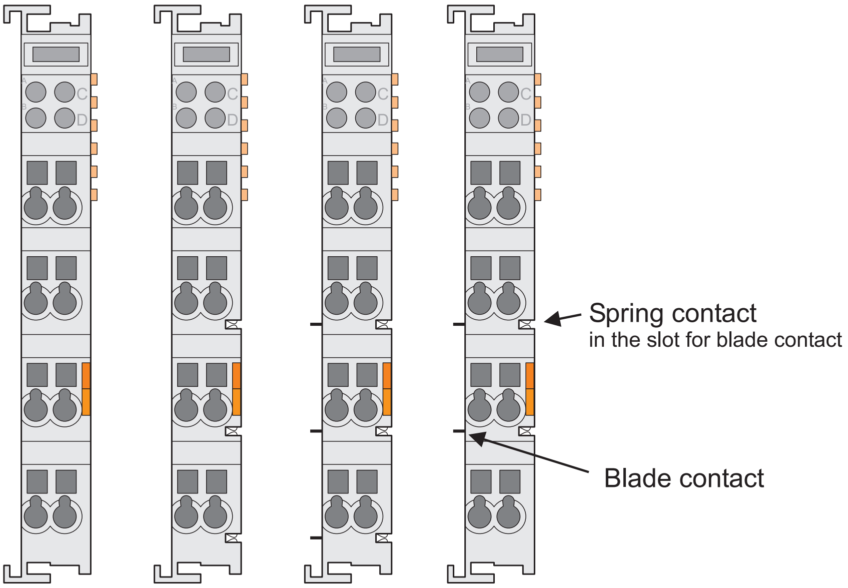


Fig. 2-8: Example for the arrangement of power contacts g0xxx05e

Recommendation
With the WAGO ProServe® Software smartDESIGNER, the assembly of a fieldbus node can be configured. The configuration can be tested via the integrated plausibility check.

2.5.9 Wire Connection

All components have CAGE CLAMP® connections.

The WAGO CAGE CLAMP® connection is appropriate for solid, stranded and fine–stranded conductors. Each clamping unit accommodates one conductor.

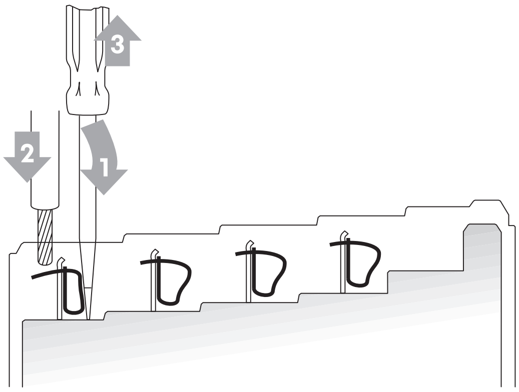


Fig. 2-9: CAGE CLAMP® Connection g0xxx08x

The operating tool is inserted into the opening above the connection. This opens the CAGE CLAMP®. Subsequently the conductor can be inserted into the opening. After removing the operating tool, the conductor is safely clamped.

More than one conductor per connection is not permissible. If several conductors have to be laid at a connection, then they should be laid in off-course wiring; e.g. together with WAGO transfer terminals.

Attention
If it is unavoidable to jointly connect 2 conductors, then a ferrule must be used.
Ferrule:

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

2.6 Power Supply

2.6.1 Isolation

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

- Operational voltage for the fieldbus interface.
- Electronics of the couplers / controllers and the bus modules (internal bus).
- All bus modules have an electrical isolation between the electronics (internal bus, logic) and the field electronics. Some analogue input modules have each channel electrically isolated, please see catalogue.

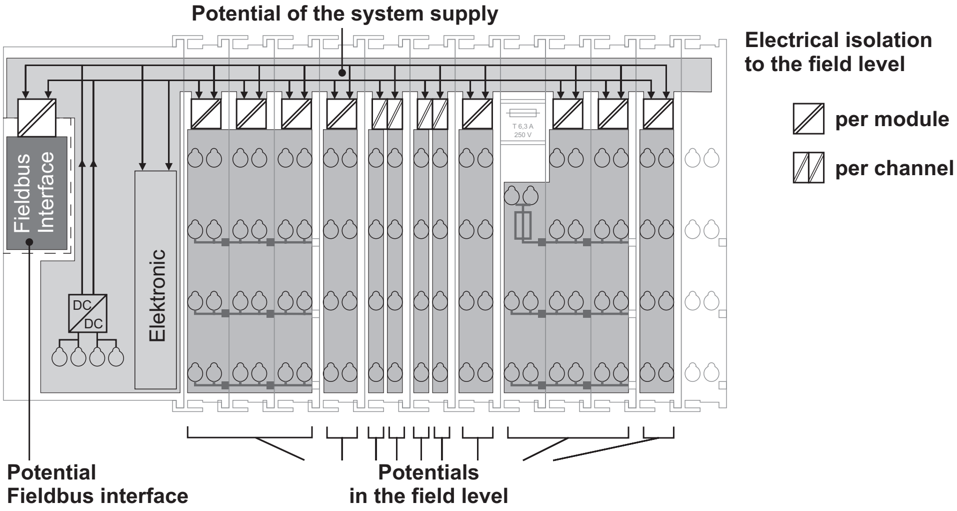

Potential of the system supply
Electrical isolation to the field level
per module
per channel
Fieldbus Interface
DC
DC
Elektronic
T 6,3 A
250 V
Potential Fieldbus interface
Potentials in the field level

Fig. 2-10: Isolation

g0xxx11e

Attention
The ground wire connection must be existent in each group. In order that all protective conductor functions are maintained under all circumstances, it is sensible to lay the connection at the beginning and end of a potential group. (ring format, please see chapter "Protective Earth"). Thus, if a bus module comes loose from a composite during servicing, then the protective conductor connection is still guaranteed for all connected field devices.

When using a joint power supply unit for the 24 V system supply and the 24 V field supply, the electrical isolation between the internal bus and the field level is disregarded for the potential group.

2.6.2 System Supply

2.6.2.1 Connection

The WAGO-I/O-SYSTEM 750 requires a 24 V direct current system supply (-15% or +20 %). The power supply is provided via the coupler / controller and, if necessary, in addition via the internal system supply modules (750-613). The voltage supply is reverse voltage protected.

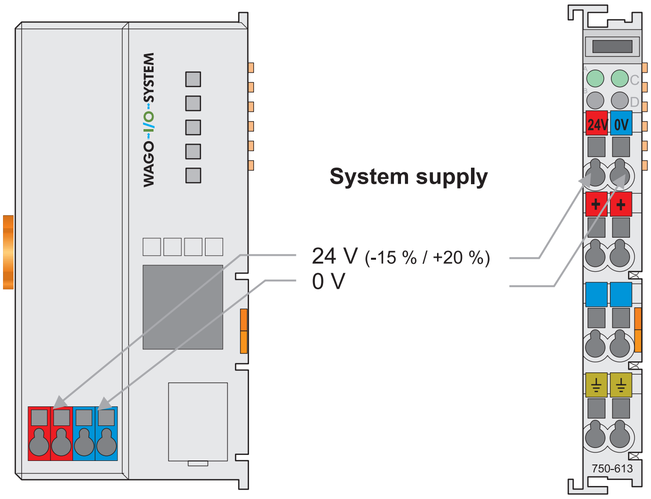


Fig. 2-11: System Supply g0xxx16e

The direct current supplies all internal system components, e.g. ECO coupler electronics, fieldbus interface and bus modules via the internal bus (5 V system voltage). The 5 V system voltage is electrically connected to the 24 V system supply.

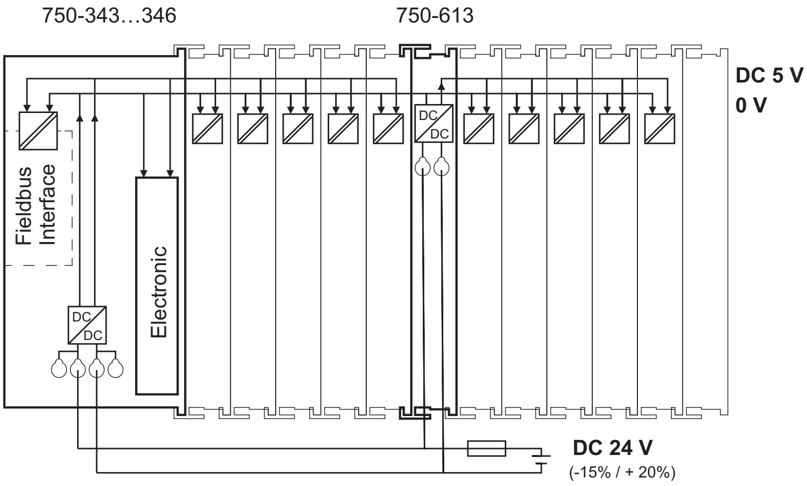


Fig. 2-12: System Voltage g0xxx12e

Attention
Resetting the system by switching on and off the system supply, must take place simultaneously for all supply modules (ECO coupler and 750-613).

2.6.2.2 Alignment

Recommendation
A stable network supply cannot be taken for granted always and everywhere. Therefore, regulated power supply units should be used in order to guarantee the quality of the supply voltage.

The supply capacity of the ECO coupler or the internal system supply module (750-613) can be taken from the technical data of the components.

Internal current consumption *)	Current consumption via system voltage: 5 V for electronics of the bus modules and ECO coupler
Residual current for bus terminals *)	Available current for the bus modules. Provided by the bus power supply unit. See ECO coupler and internal system supply module (750-613)

*) cf. catalogue W3 Volume 3, manuals or Internet

Example

ECO coupler:

internal current consumption:	350 mA at 5V
residual current for bus modules:	650 mA at 5V
sum $I_{(5V)\ ges}$:	1000 mA at 5V

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

Attention
If the sum of the internal current consumption exceeds the residual current for bus modules, then an internal system supply module (750-613) must be placed before the module where the permissible residual current was exceeded.

Example: A node with a PROFIBUS ECO Coupler consists of 20 relay modules (750-517) and 20 digital input modules (750-405).

Current consumption:

20*105 mA =	2100 mA
10* 2 mA =	20 mA
Sum	2120 mA

The coupler can provide 650 for the bus modules. Consequently, an internal system supply module (750-613), e.g. in the middle of the node, should be planned.

Recommendation
With the WAGO ProServe® Software smartDESIGNER, the assembly of a fieldbus node can be configured. The configuration can be tested via the integrated plausibility check.

The maximum input current of the 24 V system supply amounts to 260 mA. The exact electrical consumption ($I_{(24\ V)}$) can be determined with the following formulas:

ECO Coupler

$I_{(5\ V)\ ges.} =$ Sum of all current consumptions of the connected bus modules + internal current consumption of the ECO coupler

750-613

$I_{(5\ V)\ ges.} =$ Sum of all current consumptions of the connected bus modules

Input current $m\ I_{(24\ V)} =$ $5\ V / 24\ V * I_{(5\ V)\ ges.} / \eta$

$\eta = 0.87$ *(at nominal load)*

Note
If the electrical consumption of the power supply point for the 24 V-system supply exceeds 260 mA for the ECO coupler respectively 500 mA for the 750-613, then the cause may be an improperly aligned node or a defect.

During the test, all outputs, in particular those of the relay modules, must be active.

2.6.3 Field Supply

2.6.3.1 Connection

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

Power supply modules provides field side power. Power supply modules are available for other potentials, e.g. DC 24 V or AC 230 V. Likewise, with the aid of the power supply modules, various potentials can be set up. The connections are linked in pairs with a power contact.

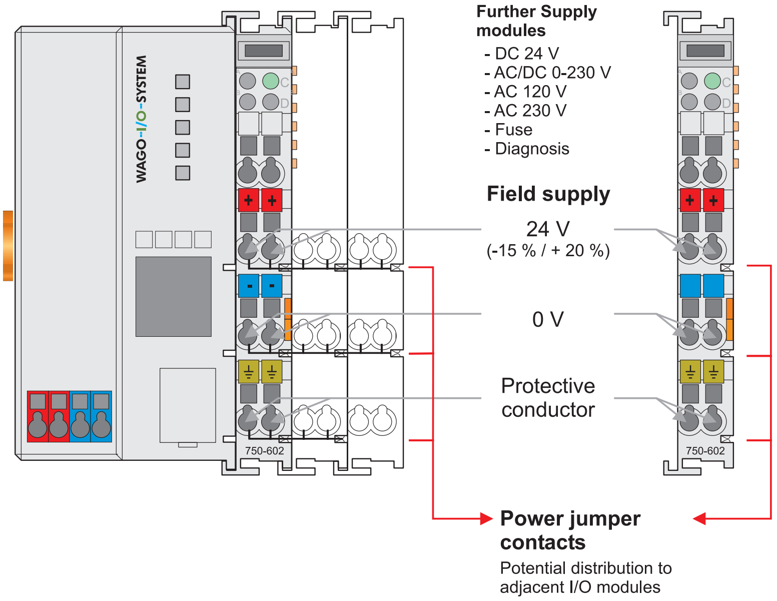


Fig. 2-13: Field Supply (Sensor / Actuator) g0xxx17e

The supply voltage for the field side is automatically passed on via the power jumper contacts when assembling the bus modules .

The current load of the power contacts must not exceed 10 A on a continual basis. The current load capacity between two connection terminals is identical to the load capacity of the connection wires.

By setting an additional power supply module, the field supply via the power contacts is disrupted. From there a new power supply occurs which also contains a potential alternation.

Attention
Some bus modules have no or very few power contacts (depends on the I/O function). Due to this, the passing on of the relevant potential is disrupted. If a field supply is required for subsequent bus modules, then a power supply module must be used.
Note the data sheets of the bus modules.

In the case of a node setup with different potentials, e.g. the alteration from DC 24 V to AC 230V, a spacer module should be used. The optical separation of the potentials acts as a warning to heed caution in the case of wiring and maintenance works. Thus, the results of wiring errors can be prevented.

2.6.3.2 Fusing

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

750-601	24 V DC, Supply / Fuse
750-609	230 V AC, Supply / Fuse
750-615	120 V AC, Supply / Fuse
750-610	24 V DC, Supply / Fuse / Diagnosis
750-611	230 V AC, Supply / Fuse / Diagnosis

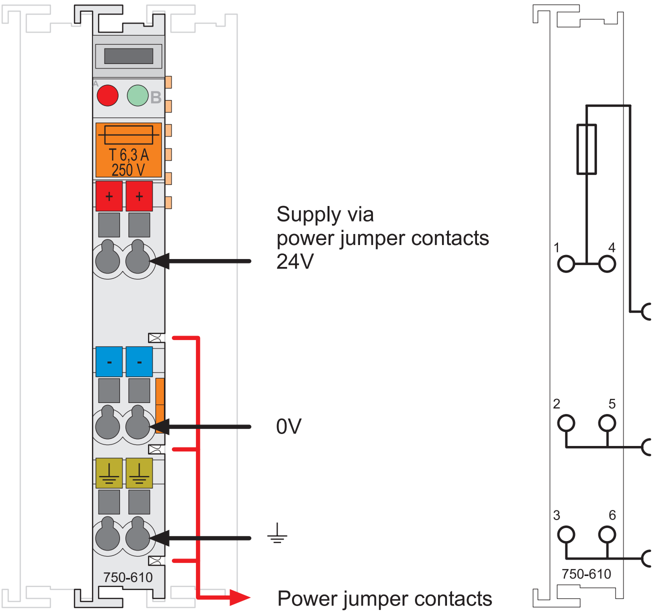


Fig. 2-14: Supply module with fuse carrier (Example 750-610) g0xxx09e

Warning
In the case of power supply modules with fuse holders, only fuses with a maximal dissipation of 1.6 W (IEC 127) must be used.

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 bus 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.

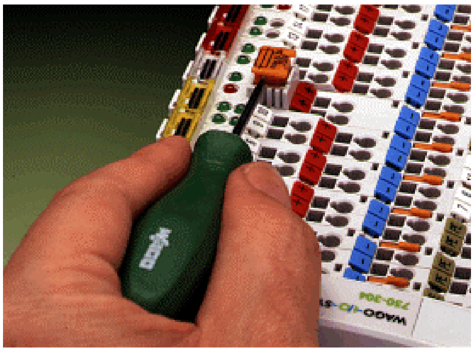


Fig. 2-15: Removing the fuse carrier p0xxx05x

Lifting the cover to the side opens the fuse carrier.

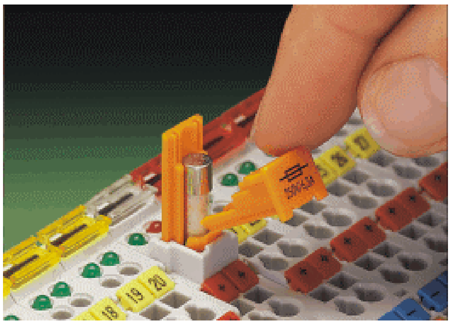


Fig. 2-16: Opening the fuse carrier p0xxx03x

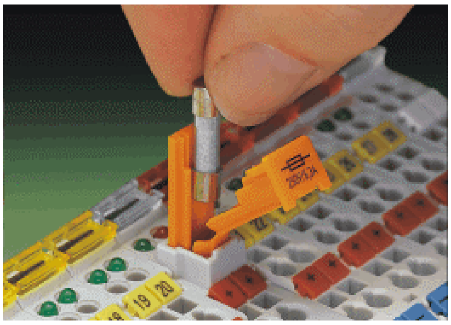


Fig. 2-17: Change fuse p0xxx04x

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.

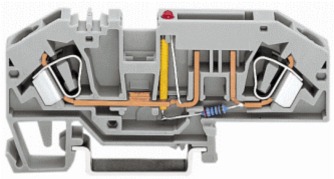


Fig. 2-18: Fuse modules for automotive fuses, Series 282 pf66800x

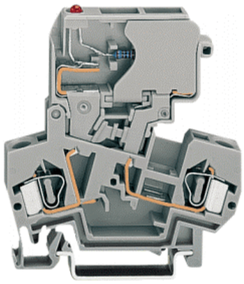


Fig. 2-19: Fuse modules with pivotable fuse carrier, Series 281 pe61100x

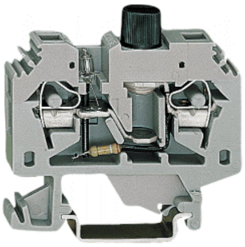
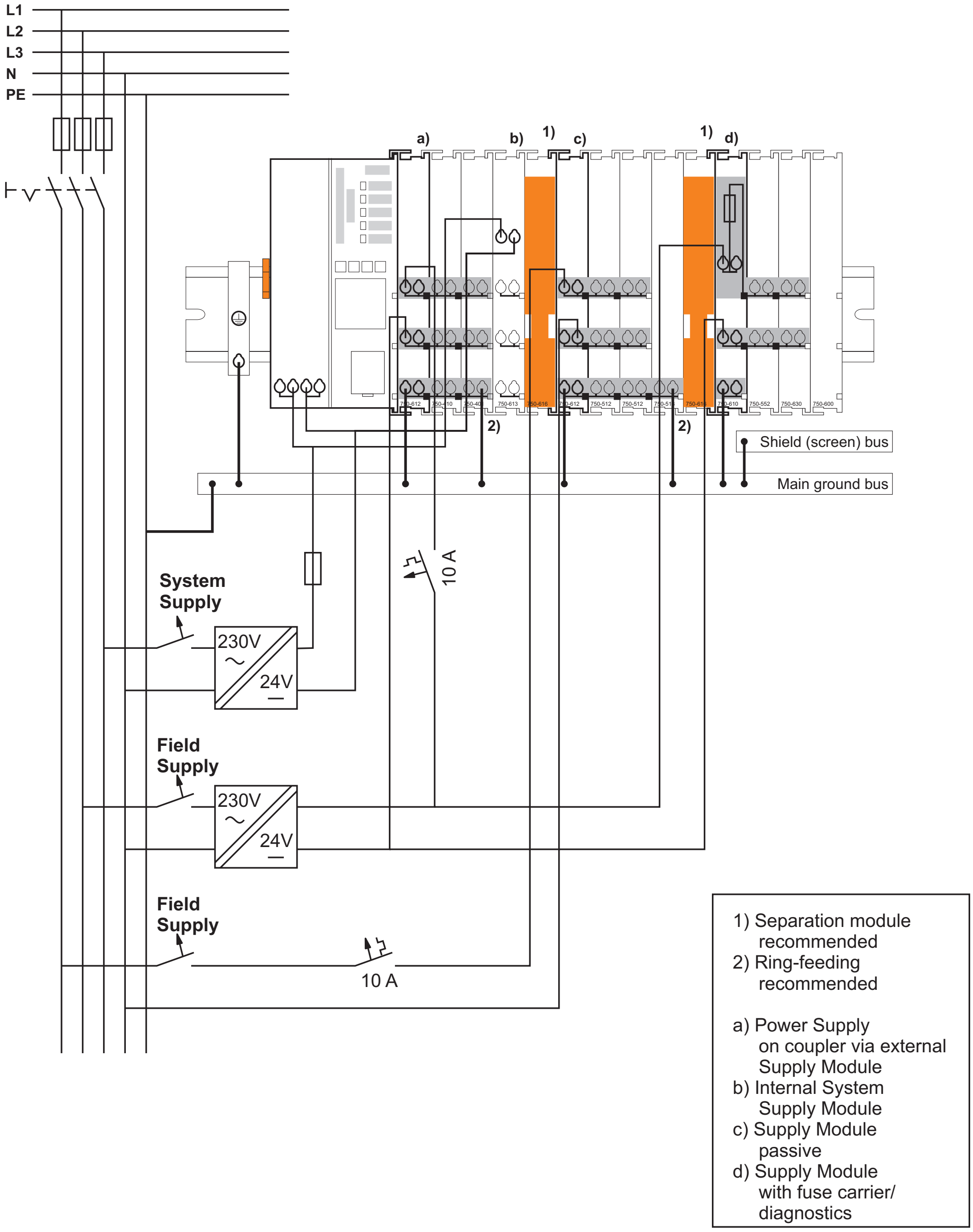


Fig. 2-20: Fuse modules, Series 282 pf12400x

2.6.4 Supply Example

Note

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



1) Separation module recommended
2) Ring-feeding recommended

a) Power Supply on coupler via external Supply Module
b) Internal System Supply Module
c) Supply Module passive
d) Supply Module with fuse carrier/ diagnostics

Fig. 2-21: Supply example

g0xxx13e

2.6.5 Power Supply Unit

The WAGO-I/O-SYSTEM 750 requires a 24 V direct current system supply with a maximum deviation of -15% or +20 %.

Recommendation
A stable network supply cannot be taken for granted always and everywhere. Therefore, regulated power supply units should be used in order to guarantee the quality of the supply voltage.

A buffer (200 µF per 1 A current load) should be provided for brief voltage dips. The I/O system buffers for ca. 1 ms.

The electrical requirement for the field supply is to be determined individually for each power supply point. Thereby all loads through the field devices and bus modules should be considered. The field supply as well influences the bus modules, as the inputs and outputs of some bus modules require the voltage of the field supply.

⚠ **Note**
The system supply and the field supply should be isolated in order to ensure bus operation in the event of short circuits on the actuator side.

WAGO products Article No.	Description
787-903	Primary switched - mode, DC 24 V, 5 A wide input voltage range AC 85-264 V PFC (Power Factor Correction)
787-904	Primary switched - mode, DC 24 V, 10 A wide input voltage range AC 85-264 V PFC (Power Factor Correction)
787-912	Primary switched - mode, DC 24 V, 2 A wide input voltage range AC 85-264 V
	Rail-mounted modules with universal mounting carrier
288-809	AC 115 V / DC 24 V; 0,5 A
288-810	AC 230 V / DC 24 V; 0,5 A
288-812	AC 230 V / DC 24 V; 2 A
288-813	AC 115 V / DC 24 V; 2 A

2.7 Grounding

2.7.1 Grounding the DIN Rail

2.7.1.1 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 electronic connection is established via the screw. Thus, the carrier rail is grounded.

⚠ **Attention**
Care must be taken to ensure the flawless electrical connection between the carrier rail and the frame or housing in order to guarantee sufficient grounding.

2.7.1.2 Insulated Assembly

Insulated assembly has been achieved when there is constructively no direct conduction connection between the cabinet frame or machine parts and the carrier rail. Here the earth must be set up via an electrical conductor.

The connected grounding conductor should have a cross section of at least 4 mm^2.

Recommendation
The optimal insulated setup is a metallic assembly plate with grounding connection with an electrical conductive link with the carrier rail.

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

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

2.7.2 Function Earth

The function earth increases the resistance capacity against disturbances from electro-magnetic influences. Some components in the I/O system have a carrier rail contact that dissipates electro-magnetic disturbances to the carrier rail.

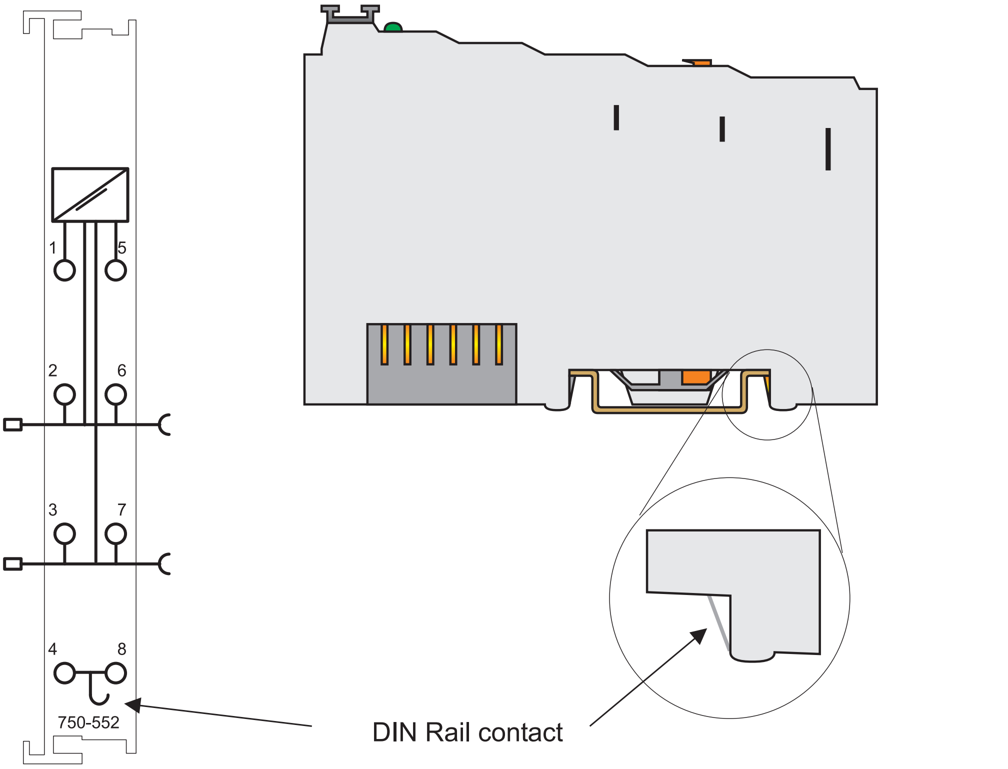


Fig. 2-22: Carrier rail contact

g0xxx10e

Attention

Care must be taken to ensure the flawless electrical connection between the carrier rail contact and the carrier rail.

The carrier rail must be grounded.

For information on carrier rail properties, please see chapter "Carrier rail properties".

2.7.3 Protective Earth

For the field level, the ground wire is placed onto the lower connection terminals of the power supply terminals and further reached through the lower power contacts to the neighboring bus terminals. If the bus terminal has the lower power contact, then the ground wire connection of the field devices can be directly connected to the lower connection terminals of the bus terminals.

⚠ **Attention**
If the connection of the power contacts for the ground wire within the node is disrupted, e.g. due to a 4-channel bus terminal, then the potential has to be re-supplied.

The ring feeding of the earth potential can increase the system security. In the event that a bus terminal is ripped out of the potential group, the earth potential is still maintained.

During the ring feeding, the ground wire is connected at the beginning and end of the potential group.

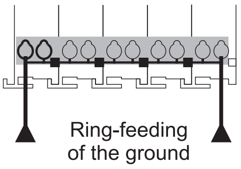


Fig. 2-23: Ring-feeding g0xxx07e

2.8 Shielding (screening)

2.8.1 General

The shielding of the data and signal conductors reduces the electromagnetic influences thereby increasing the signal quality. Measurement errors, data transmission errors and even disturbances caused by overvoltage can thus be avoided.

Attention
Constant shielding is absolutely required in order to ensure the technical specifications in terms of the measurement accuracy.

The data and signal conductors should be laid separately from all high-voltage cables.

The cable shield is to be laid over a large-scale surface onto the earth potential. With this, incoming disturbances can be easily diverted.

The shielding should be placed over the entrance of the cabinet or housing in order to already repel disturbances at the entrance.

2.8.2 Bus Conductors

The shielding of the bus conductor is described in the relevant assembly guideline of the bus system.

2.8.3 Signal Conductors

The bus terminals for analogue signals as well as some interface bus terminals possess connection terminals for the shield.

Note
Improved shielding can be achieved if the shield is previously placed over a large-scale surface. For this, we recommend the use of the WAGO shield connecting system for example.
This is particularly recommendable for systems with large-scale expansions where it cannot be ruled out that differential currents are flowing or high pulse currents, i. e. activated by atmospheric discharge, may appear.

2.8.4 WAGO Shield (Screen) Connecting System

The WAGO shield connecting system is comprised of shield terminal frames, busbars and diverse assembly feet in order to realize a multitude of constructions. Please see catalogue W3 volume 3 chapter 7.

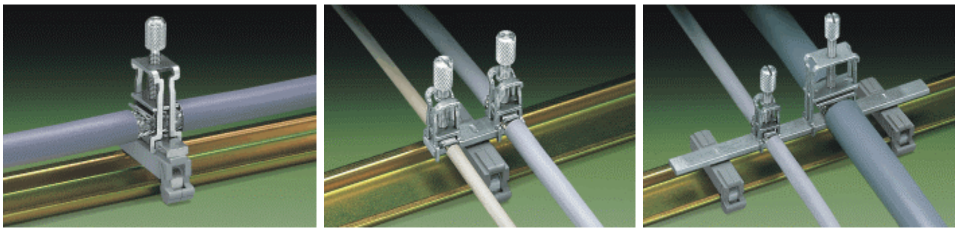


Fig. 2-24: WAGO Shield (Screen) Connecting System p0xxx08x, p0xxx09x, p0xxx10x

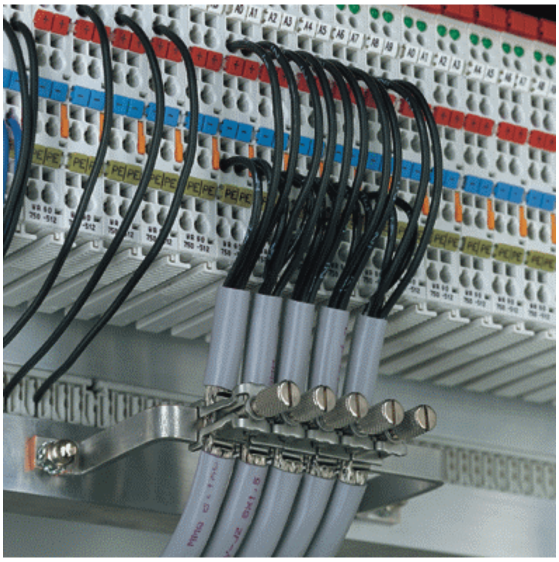


Fig. 2-25: Application of the WAGO Shield (Screen) Connecting System p0xxx11x,

2.9 Assembly Guidelines / Norms

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 – switch box combinatio

3 Fieldbus Coupler

3.1 Fieldbus Coupler 750-347 / -348

This chapter includes:

3.1.1 Description

The ECO fieldbus coupler for CANopen have different fieldbus connections.

Item no.:	Fieldbus connection:
750-347	MCS (Multi connector system)
750-348	D-SUB connection (9-pole)

Note
In this manual, the diagrams and pictures are shown from the coupler 750-348. The descriptions also apply equally to the coupler 750-347.

The ECO Fieldbus coupler are designed for applications with a reduced scale I/O requirements. By using digital only process data, or small amounts of analogs data, the ECO Fieldbus coupler retains all of the choices that is offered by the Series 750 I/O.

The couplers have an integrated supply terminal for the system voltage. The field power jumper contacts are supplied via a seperate supply module.

The CANopen buscouplers are capable of supporting all I/O modules and automatically configure, creating a local process image. CANopen allows the storing of the process image in the corresponding Master control (PLC, PC or NC). The local process image is devided into two data zones containing the data received and the data to be sent. The process data can be sent via the CANopen fieldbus to the PLC, PC or NC for futher processing, and received from the field via CANopen.

The data of analog modules is stored in the PDOs according to the order in which the modules are connected to the buscoupler. The bits of the digital modules are sent byte by byte and also mapped in the PDOs. If amount of digital information exceeds 8 bits, the buscoupler automatically starts with a new byte.

All entries of the object dictionary can be mapped – as the user prefers – in the 5 Rx PDOs and 5 Tx PDOs. The complete input and output process image can be transmitted using SDOs.

3.1.2 Hardware

3.1.2.1 View

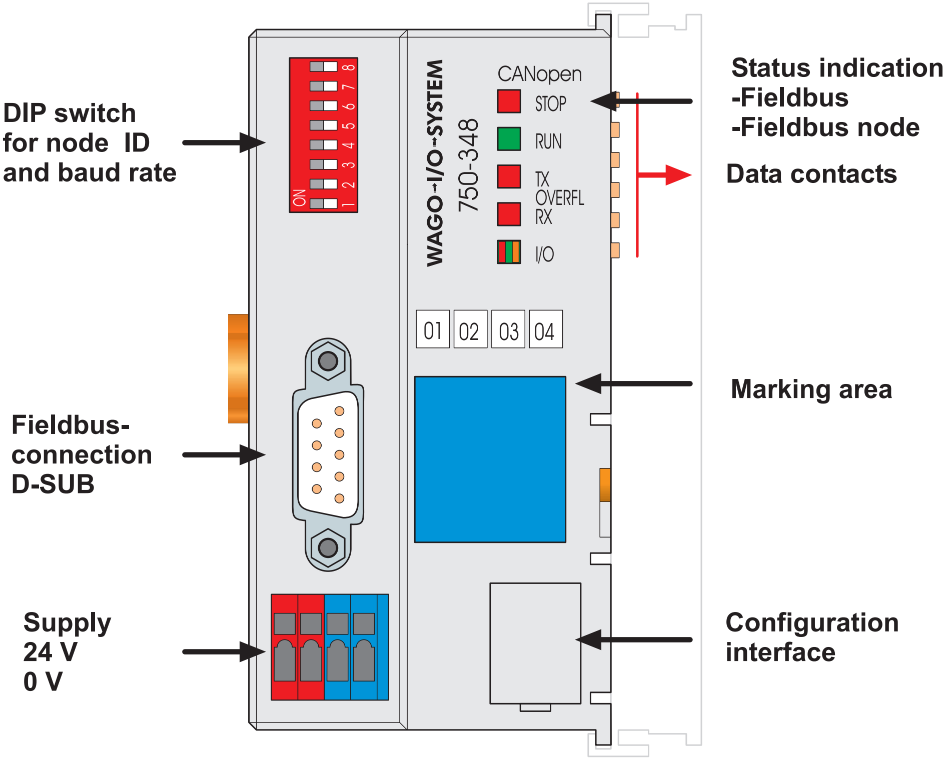


Fig. 3-1: ECO fieldbus coupler CANopen 750-348 g034800e

The fieldbus coupler is comprised of:

- Supply module with internal system sypply module for the system supply.
- Fieldbus interface with the bus connection
- DIP switch for baud rate and node ID
- Display elements (LEDs) for status display of the operation, the bus communication, as well as for fault messages and diagnosis
- Configuration Interface
- Electronics for communication with the I/O modules (internal bus) and the fieldbus interface

3.1.2.2 Device Supply

The supply is made via terminal bocks with CAGE CLAMP® connection. The device supply is intended for the system.

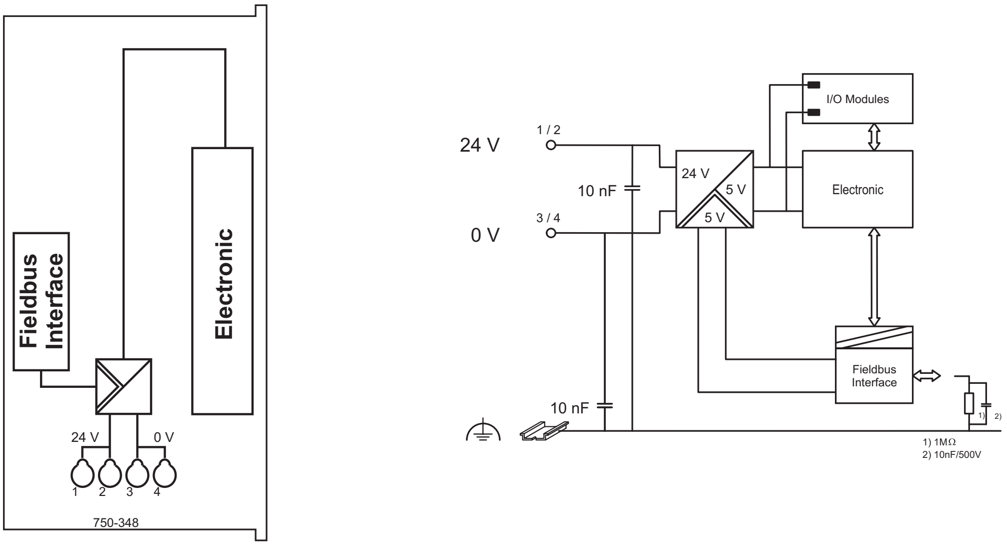


Fig. 3-2: Device supply g034801e

The integrated internal system supply module generates the necessary voltage to supply the electronics of the coupler and the connected I/O modules.

The fieldbus interface is supplied with electrically isolated voltage from the internal system supply module.

3.1.2.3 Fieldbus Connection

750-347	The CAN interface of the Fieldbus coupler 750-347 is designed as an Open Style connection, MCS (Multi Connection System; Series 231).
750-348	The Fieldbus coupler 750-348 posseses a 9-pole D-SUB-plug connector for the field bus connection.

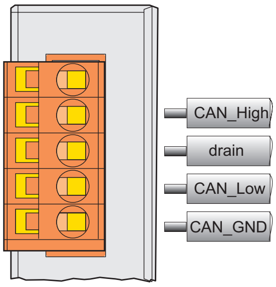
The connection point is lowered in such a way that after a connector is inserted, installation in an 80 mm high switchbox is possible.

The electrical isolation between the fieldbus system and the electronics is made via the DC/DC converter and the optocoupler in the fieldbus.

The line screen must be connected for both field bus couplers on CAN Shield.

For both fieldbus coupler the shield on the CANopen communication cable must be connected to drain. The shield is connected via a 1 MΩ resistor to ground (earth) (rail carrier contact). A connection of low impedance between shield and ground (earth) can only be made externally (for example by a supply terminal block). It is recommended to have a central ground (earth) contact for the whole CANbus shield.

3.1.2.3.1 MCS (750-347)

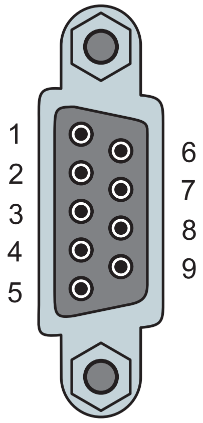


Pin	Signal	Description
1	CAN_V+	Not used
2	CAN_H	CAN Signal $_{\text{High}}$
3	Drain, Shield	Shield connection
4	CAN_L	CAN Signal $_{\text{Low}}$
5	GND	Ground

Fig. 3-3: Pin assignment for fieldbus connection of the 750-347, Series 231 (MCS) g012400x

3.1.2.3.2 D-SUB (750-348)

Signal	Pin
Not used	**1**
CAN Signal $_{\text{Low}}$	**2**
Ground (GND)	**3**
Not used	**4**
Not used	**5**



Pin	Signal
6	Not used
7	CAN Signal $_{\text{High}}$
8	Not used
9	Not used

Fig. 3-4: Pin assignment for fieldbus connection of the 750-348, D-SUB g012231d

3.1.2.4 Display Elements

The operating condition of the fieldbus coupler or node is signalled via LED's (Light diodes).

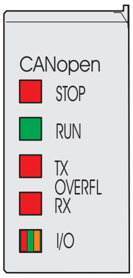


Fig. 3-5: Display elements CANopen g034802x

LED	Color	Meaning
STOP	red	The buscoupler / node is in the state STOP
RUN	green	The buscoupler / node is in the state OPERATIONAL
Tx-Overflow	red	CAN transmitter buffer is full.
Rx-Overflow	red	CAN receiver buffer is full.
IO	red /green / orange	The 'I/O'-LED indicates the operation of the node and signals faults encountered.

3.1.2.5 Configuration Interface

The configuration interface used for the communication with WAGO-I/O-CHECK or for firmware transfer is located behind the cover flap.

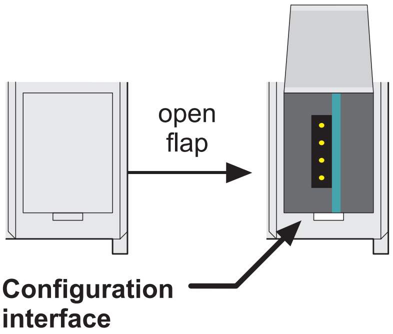


Fig. 3-6: Configuration interface g01xx06e

The communication cable (750-920) is connected to the 4 pole header.

3.1.2.6 Hardware Address (Module ID)

The DIP switches are used both for setting the baud rate of the fieldbus coupler and for setting the module ID. This module ID is necessary for calculating the COB IDs (i.e. of PDO1...4, 1. Server SDO, etc.).

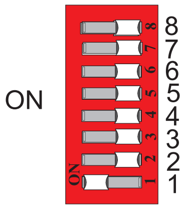


Fig. 3-7: Setting of station (node) address g012548x

The binary significance of the individual DIP switches increases according to the switch number, i.e. the module ID 1 is set by DIP1 = ON, the module ID 8 by DIP4 = ON, etc.
The nodes of the WAGO-I/O-SYSTEM can have module IDs from 1 to 127.

3.1.2.7 Setting the Baud Rate

The bus coupler supports 9 different Baud rates. DIP switches are used to set the baud rate.

The bus coupler changes to the configuration mode using the set module ID = 0 (all DIP switches off) with subsequent power On. The current set baud rate is displayed in this status. The baud rate display is shown by the top LED group (STOP, RUN, Tx-, Rx-Overflow), whereby STOP = Switch 1, RUN = Switch 2, Tx-Overflow = Switch 3 and Rx-Overflow = Switch 4. The current set baud rate is displayed by the corresponding LEDs blinking slowly. Now the new baud rate can be set using the DIP switch, by turning the corresponding DIP switches to 'ON'.

The set configuration is saved by turning DIP8 to 'ON'. Following saving, the new baud rate is displayed by the corresponding LEDs having a steady light. Except for the baud rate of 1MBaud, this is displayed by all 4 LEDs blinking/being lit.

Example: 125 kB: Tx-Overflow LED blink / are lit
250 kB: STOP und RUN LED blink / are lit

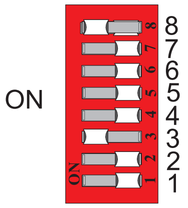


Fig. 3-8: Example: Saving the baud rate 125 kB g012449x

In this status no data exchange via CAN is possible.

Dip	Function	1 MBit	800 kB	500 kB	250 kB	125 kB	100 kB	50 kB	20 kB	10 kB	is displayed by LED
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	Acceptance	´off´ -> ´on´ : Accepting the configuration settings									

Once the baud rate setting / baud rate check is completed, switch off the operating voltage knowing that only the DIP value will be used to calculate the IDs which has been set during power ON. When switched off, the desired module ID (=1 as delivered) can be set on the DIP.

Default baud rate: 125 kB

3.1.3 Operating System

Following is the configuration of the master activation and the electrical installation of the fieldbus station.

After switching on the supply voltage, the coupler tests the I/O modules and the fieldbus interface. Following this, the I/O modules and the present configuration is determined, whereby an external not visible list is generated. This list illustrates the local input and output section on field bus RAM of the protocol chip.

In the event of a fault the coupler changes to the "Stop" condition. The "I/O" LED flashes red. After clearing the fault and cycling power, the coupler changes to the "Fieldbus start" status and the "I/O" LED lights up green.

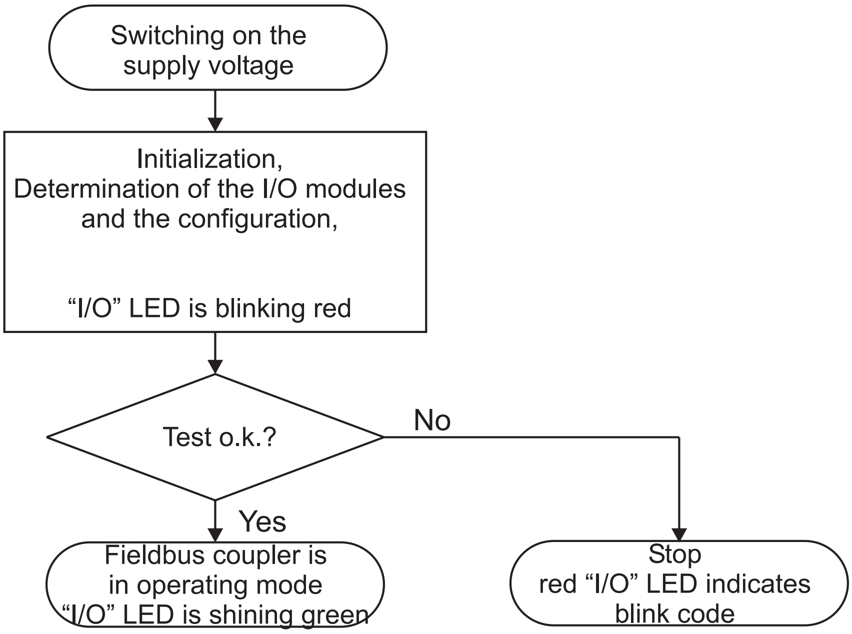

Switching on the supply voltage
Initialization, Determination of the I/O modules and the configuration,
“I/O” LED is blinking red
Test o.k.?
No
Yes
Fieldbus coupler is in operating mode “I/O” LED is shining green
Stop red “I/O” LED indicates blink code

Fig. 3-9: Operating system - Power on g012x20e

3.1.4 Process Image

After powering up, the coupler recognizes all I/O modules plugged into the node which supply or wait for data with a data width/bit width > 0. In the nodes analog and digital I/O modules can be mixed.

The coupler produces an internal process image from the data width and the type of I/O module as well as the position of the I/O modules in the node. It is divided into input and output data areas.

The data of the digital I/O modules is bit orientated, i.e. the data exchange is made bit for bit. The analog I/O modules are all byte orientated I/O modules, i.e. modules where the data exchange is made byte for byte. These I/O modules include for example the counter modules, I/O modules for angle and path measurement as well as the communication modules.

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

The data of the I/O modules is separated for the local input and output process image in the sequence of their position after the coupler in the individual process image.
In the respective I/O area, first of all analog modules are mapped, then all digital modules, even if the order of the connected analog and digital modules does not comply with this order. The digital channels are grouped, each of these groups having a data width of 1 byte. Should the number of digital I/Os exceed 8 bits, the coupler automatically starts another byte.

Note
A process image restructuring may result if a node is changed or extended. In this case the process data addresses also change in comparison with earlier ones. In the event of adding a module, take the process data of all previous modules into account.

3.1.5 Data Exchange

With CANopen, the transmission of data is made using communication objects.

Each communication object consists of a CAN telegram with 8 byte data and a unique COB-ID (Communication Object Identifier) in the network.

The transmission of data, the triggering of events, the signalling of error states etc. is made using communication objects.
Parameters for the communication objects as well as parameters and data of the CANopen subscribers are filled in an object directory.

3.1.5.1 Communication Objects for the Couplers 750-347 and 750-348

The ECO Fieldbus couplers support the following communication Objects:

- 5 Tx-PDOs,
  for process data exchange of fieldbus node input data

- 5 Rx-PDOs,
  for process data exchange of fieldbus node output data

- 2 Server SDOs,
  for exchange of configuration data and for information on the state of the node

- Synchronisations Object (SYNC),
  for network synchronisation

- Emergency Object (EMCY)

- Network Management Objects
  - Module Control Protocols
  - Error Control Protocols
  - Bootup Protocol

3.1.5.2 Communication Interfaces

For a data exchange, the CANopen fieldbus couplers are equipped with two interfaces:

- the interface to fieldbus (-master) and

- the interface to the bus modules.

Data exchange takes place between the fieldbus master and the bus modules.

Access from the fieldbus side is fieldbus specific.

3.1.5.3 Memory Areas

The couplers use a memory space of 32 Bytes for the physical input and output data.

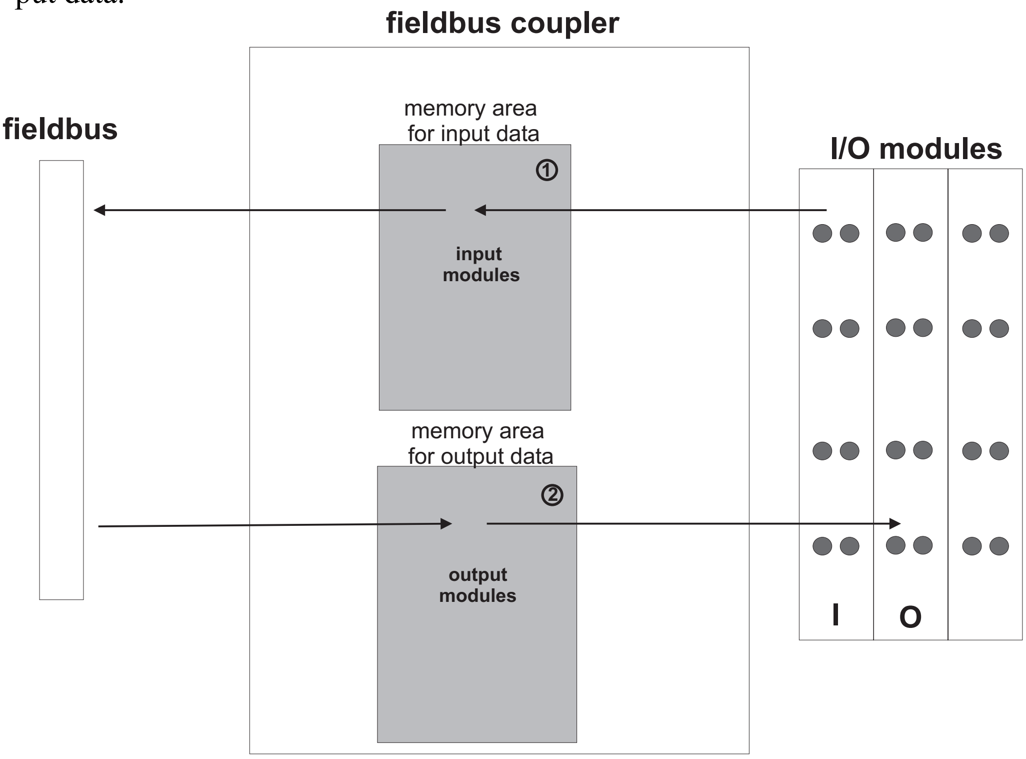


Fig. 3-10: Memory areas and data exchange for a fieldbus coupler g012447e

The coupler process image contains the physical data of the bus modules in a storage area for input data and in a storage area for output data (each 32 Bytes).

① The input module data can be read from the fieldbus side.

② In the same manner, writing to the output modules is possible from the fieldbus side.

3.1.5.4 Addressing

Upon switching on the supply voltage, the data is mapped from the process image to an object directory (initialization). A CANopen fieldbus master uses the 16 bit indexes and 8 bit sub-indexes of the object directory in order to address the data via the PDOs or SDOs and for access purposes.
Therefore, the position of the data in the process image has no direct meaning for the CANopen user.

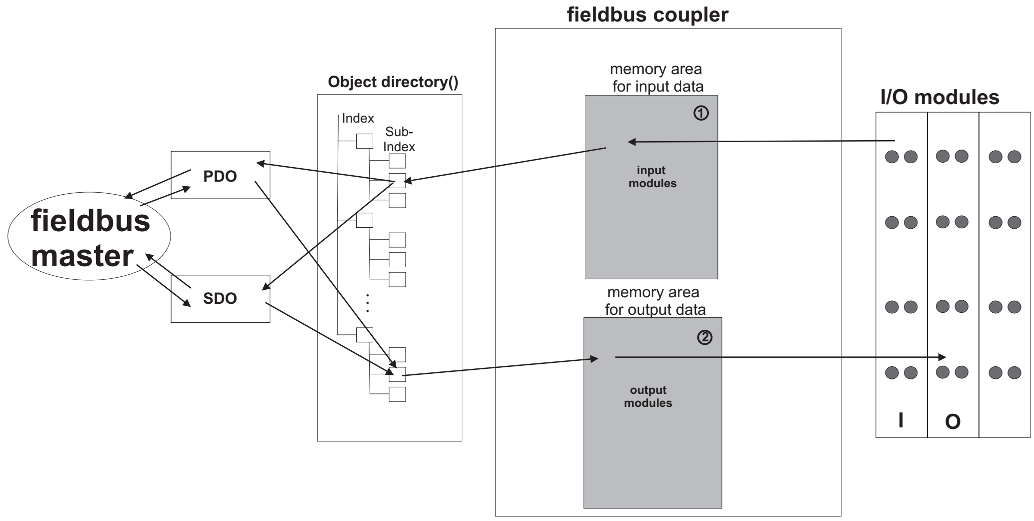


Fig. 3-1: Fieldbus specific data exchange for a CANopen fieldbus coupler g012446e

3.1.5.4.1 Indexing the Bus Module Data

If a customer specific configuration was stored prior to the initialization, and if the currently connected module configuration coincides with the configuration stored before, initialization takes place with this configuration.

Note
For an example for the initialization of the customer specific configuration, please refer to chapter 3.1.6 "Starting up CANopen Fieldbus Nodes".

In every other case, when initializing, the object directory is assigned a default configuration according to the device profile DS 401.

The entry into the object directory is then made separately according to data width (1 bit, 1 byte, 2 bytes, 3 bytes, etc...) and input and output.
The physical bus module arrangement within a node is optional.

Data width = 1 Word / channel	Data width = 1 Bit / channel
Analog input modules	Digital input modules
Analog output modules	Digital output modules
Input modules for thermal elements	Digital output modules with diagnostics (2 Bit / channel)
Input modules for resistance sensors	Power supply modules with fuse holder / diagnostics
Pulse width output modules	Solid State power relay
Interface module	Relay output modules
Up/down counter	
I/O modules for angle and path measurement	

Table 3-1: I/O module data width

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

The digital module data is taken into consideration first.
Knowing that CANopen does not transmit the data bit by bit, the digital module data is grouped to form bytes and assigned to the corresponding index, digital input data to index 0x2000, digital output data to index 0x2100.
The assignment of bus module data of a data width of 1 byte or more is made in relation to the individual indices.

The table reviews the indices of the bus module data.

Data width	input modules	output modules
	Index	
1 Bit digital	0x2000	0x2100
1 Byte specialty modules	0x2200	0x2300
2 Byte specialty modules	0x2400	0x2500
3 Byte specialty modules	0x2600	0x2700
4 Byte specialty modules	0x2800	0x2900
5 Byte specialty modules	0x3000	0x3100
6 Byte specialty modules	0x3200	0x3300
7 Byte specialty modules	0x3400	0x3500
8 Byte specialty modules	0x3600	0x3700

Table 3-2: Indexing the bus module data in the object directory

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

The number of data inputs is quoted in sub-index 0, whereas in the following sub-indices the data is filled in blocks.
The block size depends on the data width of the bus module.

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

Table 3-3: Sub-indices of the bus module data in the object directory

More information
For a detailed description of setting the default configuration please refer to the chapter 5.3.4 "Object Directory".

Attention
A process image restructuring may result if a node is changed or extended. In this case the process data addresses also change in comparison with earlier ones. In the event of adding modules, take the process data of all previous modules into account.

Example:

The bus module configuration contains :

1) 5 digital 2 channel input modules (i.e. 750-400),

2) one digital 4 channel output module (i.e. 750-504) and

3) two 2 channel analog output modules with output modules having 2 bytes per channel (i.e. 750-552).

To 1) Index the data of the 5 digital 2 channel input modules:

Index:	**Sub-Index:**	**Contents:**	**Description:**
0x2000	0	2	number 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 1channel 2, etc.

To 2) Index the data of the digital 4 channel output module:

Index:	**Sub-Index:**	**Contents:**	**Description:**
0x2100	0	1	number of dig. 8 Bit input 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.

To 3) Index the data of the 2 analog 2 channel output modules:

Index:	**Sub-Index:**	**Contents:**	**Description:**
0x2500	0	4	number of 2 Byte specialty 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 module 1 channel 1, D1.2 = Data word module 1 channel 2, etc.

3.1.6 Starting up CANopen Fieldbus Nodes

This chapter shows the step-by-step procedure for starting up a WAGO CANopen fieldbus node.

Attention
This description is given as an example and is limited to the execution of a local start-up of an individual CANopen fieldbus node.

The procedure contains the following steps:

1. Connecting the PC and fieldbus node
2. Checking and setting the Baud rate
3. Setting the module ID
4. Changing to the OPERATIONAL status
5. Releasing the analog input data
6. Application specific mapping

3.1.6.1 Connecting the PC and Fieldbus Node

Connect the fitted CANopen fieldbus node to the CANopen fieldbus PCB in your PC via a fieldbus cable and start your PC.

3.1.6.2 Checking and Setting the Baud Rate

First of all, turn all DIP switches to the “OFF” position (module ID = 0), then apply the supply voltage (DC 24 V power pack) to the fieldbus coupler.

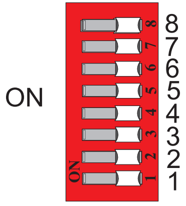


Fig. 3-11: All DIP switches to “OFF” for checking and setting the Baud rate g012450x

Now the currently set Baud rate is checked and displayed by the blinking of the top group of LED's.

Note
If applying voltage when not all of the DIP switches are in the “OFF” position, the existing setting will be written as a module ID.

Now push the corresponding DIP switches to the desired Baud rate to 'ON', i.e. DIP switch 3 for the Baud rate 125 kB.
Wait until the new baud rate is displayed by LED blinking.

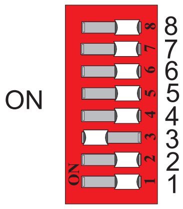


Fig. 3-12: Setting the Baud rate 125 kB g012550x

To be able to store the new setting, push DIP switch 8 also to 'ON'.
Then switch off the coupler supply voltage.

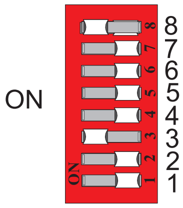


Fig. 3-13: : Storing the Baud rate 125 kB g012449x

3.1.6.3 Setting the Module ID

The module ID is set with the supply voltage switched off. Then push all DIP switches to the “OFF” position again. Next, push the DIP switch intended for the desired module ID to “ON”, i.e. DIP switch 1 for the module ID 1.

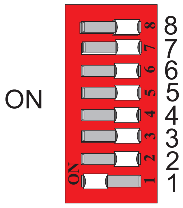


Fig. 3-14: Setting the module ID 1 g012548x

As soon as you switch on the supply voltage, the coupler is in the INITIALIZATION state.
At the same time, the process image is created by means of the connected bus modules and the object directory initialized following the default mapping, if no application specific configuration was stored.
After a fault-free termination of the initialization phase, the coupler automatically changes to the PRE-OPERATIONAL status.
In this status, communication is possible via SDOs, which you can now use to proceed with various settings via your CAN Master software:

- You can set the coupler directly to its OPERATIONAL status

Note
Due to the fact that, as a default setting, the PDO transmission of the analog input data is switched off, the analog input data will not be taken into consideration.

- You can release the switched off transmission of the analog input data or
- Select an application specific mapping.

3.1.6.4 Changing to the OPERATIONAL Status

You can change the coupler PRE-OPERATIONAL status to the OPERATIONAL status using the **Start_Remote_Node** command from the network management objects. This creates the communication readiness of the fieldbus node for PDOs (see chapter 5.3.3.1.1 "Start Remote Node").

Note
As a default setting, the PDO transmission of the analog input data is switched off. For this reason, this data is read out only once and subsequently never updated. To be able to use this data via the PDOs, switch the analog input data on in the PRE-OPERATIONAL status. Access via the SDOs is possible at any time.

If no further settings are made, the coupler is operational. Communication is possible according to the Default-Mapping (refer to chapter 5.3.4.1 "Initialization").

3.1.6.5 Switching on the Analog Input Data

To avoid the CAN bus from overflowing with CAN messages, the transmission of analog input data via PDOs is deactivated in the default setting. This means that object 0x6423 "Analoge Input Global Interrupt Enable" has the default value 'FALSE' (= '0') (refer to chapter 5.3.4.4.12 "Object 0x6423, Analogue Input Global Interrupt Enable").
When the coupler has the PRE-OPERATIONAL status, you can generally release the transmission by setting the object 0x6423 to the TRUE (= '1') value. Subsequently, the "**Start Remote Node**" command can be used to change the coupler status from PRE-OPERATIONAL to OPERATIONAL. This process allows communication via PDOs and the transmission of analog input data.

If no further settings are made, the coupler is operational and communication can occur according to the Default Mapping (refer to chapter 5.3.4.1 "Initialization").

3.1.6.6 Application Specific Mapping

An alternative to the use of the default mapping is to define the data to be transmitted by PDOs in an application specific PDO mapping. For this purpose, the coupler has to be in the PRE-OPERATIONAL status or has to be transferred in this status by using the NMT service "**Enter Pre-Operational**".

Details of how to proceed with an application specific mapping are explained below.

Example:
The 3rd and the 5th 2 byte analog input channel and the first 8 bit digital input group are to be read using the TxPDO 2. For transmission purposes, the CAN identifier 0x432 is to be used. Transmission must be synchronous with each 3rd SYNC object.

The default CAN IDs are used for the SDOs. The setting is made at node 8.

xx... is not evaluated

1. First of all, deactivate the PDO you wish to map.
   In the present example, this is the TxPDO2.
   To this effect, write value 0x80000000 into the object having the index 0x1801, sub-index 01 (Transmit PDO Communication Parameter).

Deactivating PDO:		
	CAN ID	**Data**
Transmit	608	0x23 01 18 01 00 00 00 80
Receive	588	0x60 01 18 01 xx xx xx xx

2. Then deactivate the PDO mapping by zeroing the number of mapping objects in index 0x1A01, sub-index 0 (Transmit PDO Mapping Parameter).

Deactivating PDO mapping:		
	CAN ID	**Data**
Transmit	608	0x2F 01 1A 00 00 xx xx xx
Receive	588	0x60 01 1A 00 xx xx xx xx

3. Enter into the TxPDO mapping parameter structure (Index 0x1A01) the Index, Sub-Index and the Object length of the application object.
   Max. 8 bytes of data can be assigned per PDO.

Writing into the mapping parameter structure:		
Application object	**Index**	**Sub-Index**
3. analog input channel	0x2400	3
5. analog input channel	0x2400	5
1. digital input group	0x2000	1

The following structure must be reached in the mapping parameters of the 2nd TxPDO in order to ensure the task set:

TxPDO Mapping Parameter Structure, Index 0x1A01			
	Application object		
Sub-Index:	Index:	Sub-Index:	Object length in Bit
0	3		
1	0x2400	3	0x10
2	0x2400	5	0x10
3	0x2000	1	0x08

→

Note
First of all enter the mapping parameter sub-index 1 ... 8 in the sub-index 0, followed by the number of valid sub-indexes.

These objects are stored with the aid of SDO transmissions:

Mapping 3. analog input channel		
	CAN ID	Data
Transmit	0x608	0x23 01 1A 01 10 03 00 24 23 0 data bytes invalid 011A Index(Lowbyte 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 00 24 Index (Lowbyte first) where the 3rd analog channel is in the manufacturer device profile
Receive	0x588	0x60 01 1A 01 xx xx xx xx 60 OK 011A Index (Lowbyte first) 01 Sub-Index

Mapping 5. analog input channel		
	CAN ID	Data
Transmit	0x608	0x23 01 1A 02 10 05 00 24
Receive	0x588	0x60 01 1A 02 xx xx xx xx

Mapping 1. digital input group		
	CAN ID	Data
Transmit	0x608	0x23 01 1A 03 08 01 00 20
Receive	0x588	0x60 01 1A 03 xx xx xx xx

Number of mapping objects = 3, enter on Sub-Index 0		
	CAN ID	**Data**
Transmit	0x608	0x2F 01 1A 00 03 xx xx xx
Receive	0x588	0x60 01 1A 00 xx xx xx xx

4. Now write into the object with Index 0x1801, Sub-Index 1 to 3 (Transmit PDO Communication Parameter) the communication parameters in the structure.
   Thereby the Transmission Type is 3 (Synchronous transmission with every 3. SYNC object).

Enter the Communication Parameter:		
TxPDO Communication Parameter, Index 0x1801		
Sub-Index:	**Value:**	**Meaning:**
0	3	Number of supported entries in the record
1	0x432	COB-ID used by PDO
2	3	Transmission Type
3	0	Inhibit Time

Sub-Index 3: Inhibit Time = 0		
	CAN ID	**Data**
Transmit	0x608	0x2B 01 18 03 00 00 xx xx
Receive	0x588	0x60 01 18 03 xx xx xx xx

Sub-Index 2: Transmission Type = 3		
	CAN ID	**Data**
Transmit	0x608	0x2F 01 18 02 03 xx xx xx
Receive	0x588	0x60 01 18 02 xx xx xx xx

Sub-Index 1: Change COB-ID = 432 on PDO and PDO from invalid to valid		
	CAN ID	**Data**
Transmit	0x608	0x23 01 18 01 32 04 00 00
Receive	0x588	0x60 01 18 01 xx xx xx xx

5. When you change the bus coupler to OPERATIONAL using **the ”Start Remote Node**" message, the PDOs are activated and the TxPDO object can now be used for data transmission.

3.1.7 LED Display

The coupler possesses several LEDs for on site display of the coupler operating status or the complete node.

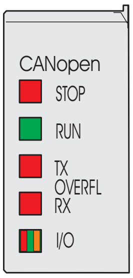


Fig. 3-15: Display elements ECO Fieldbus coupler CANopen g034802xx

Here is a description of the two LED groups.

The first group = fieldbus contains the solid colored LEDs with the denotation STOP (red), RUN (green), Tx-Overflow (red) and Rx-Overflow (red) signalling the operating status of the communication via CAN.

The second group = module bus consists of the three-color I/O LED (red/green/orange). The module bus status and the software exception codes are signalled by this LED.

3.1.7.1 Blink Code

Detailed fault messages are displayed with the aid of a blink code. A fault is cyclically displayed with up to 3 blink sequences.

- The first blink sequence (approx. 10 Hz) starts the fault display.
- The second blink sequence (approx. 1 Hz) following a pause. The number of blink pulses indicates the **fault code**.
- The third blink sequence (approx. 1 Hz) follows after a further pause. The number of blink pulses indicates the **fault argument**.

3.1.7.2 Fieldbus Status

The upper four LED´s (STOP, RUN, Tx- und Rx-Overflow) signal the operating conditions of the CAN communication.

STOP	RUN	TXOVERF	RXOVERF	Meaning	Remedy
OFF	OFF	OFF	OFF	No function or self-test	check supply (24V and 0V), wait for self-test
OFF	SLOW FLASHING	X	X	Module is in the state PRE-OPERATIONAL	
OFF	ON	X	X	Module is in the state OPERATIONAL	
ON	OFF	X	X	Module is in the STOP state or fatal fieldbus independent error (i.e. a module was removed), incorrect configuration	Check in the event of a fieldbus independent error, reset the node, check same in the event of a configuration error
X	X	X	ON	CAN receiver buffer is full. Data loss is likely.	Increase the time span between 2 protocols.
X	X	ON	X	CAN transmitter buffer is full. Data loss is likely.	Check the data sizes of the bus system. Increase the transmit priority of the module.
X	X	FAST FLASHING in turns with RXOVERF	FAST FLASHING in turns with TXOVERF	CAN Controller exceeded the Warning Level, to many error messages	Check baud rate and bus connection, install min. 2 modules in the network.
OFF	FAST FLASHING	X	X	Module is in the state PRE-OPERATIONAL, Sync/Guard Message/Heartbeat failed	Change into the state OPERATIONAL and restart Sync/Guard message/Heartbeat
FAST FLASHING	FAST FLASHING	X	X	Module is in the state OPERATIONAL, Sync/Guard Message/Heartbeat failed	Restart Sync/Guard message/Heartbeat
FAST FLASHING	OFF	X	X	Module is in the state STOP, Sync/Guard Message/Heartbeat failed	Change into the state OPERATIONAL and restart Sync/Guard message/Heartbeat

3.1.7.3 Node Status

LED	Color	Meaning
IO	red /green / orange	The 'I/O' LED indicates the node operation and signals faults occurring.

The coupler starts after switching on the supply voltage. The "I/O" LED flashes red. Following an error free start up the "I/O" LED changes to green steady light.
In the case of a fault the "I/O" LED continues blinking red. The fault is cyclically displayed with the blink code.

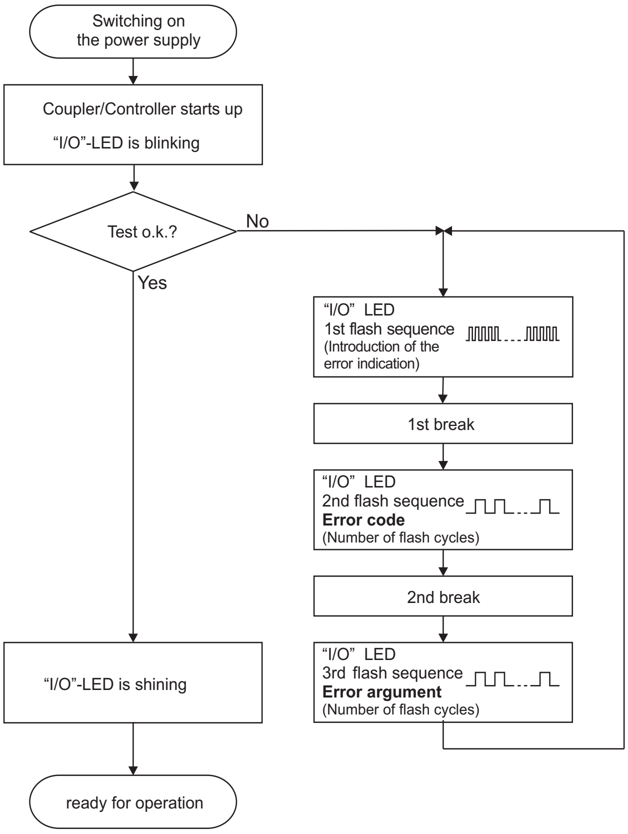


Fig. 3-16: Signalling the LED's node status g012911e

After overcoming a fault, restart the coupler by cycling the power.

I/O	Meaning
green	Data cycle on the internal bus
off	No data cycle on the internal bus
red	Coupler hardware defective
red blinks	When starting: internal bus is initialized During operation: general internal bus fault
red blinks cyclically	Fault message during internal bus reset and internal fault:

3.1.7.4 Fault Message via the Blink Code of the I/O LED

Fault argument	Fault description
Fault code 1: Configuration fault	
0	Reserved
1	Overflow of the internal buffer memory for the inline code
2	Reserved
3	Check sum fault in the parameter range of the EEPROM
4	Fault when writing in the serial EEPROM
5	Fault when reading in the serial EEPROM
6	Changed I/O module configuration found after AUTORESET
7	Reserved
8	Timeout error on the serial EEPROM
Fault code 2: Another configuration faults	
0	-
1	Reserved
2	Process image is larger than the available buffer
3	Process image length is zero
Fault code 3: Internal bus command fault	
0	I/O module(s) has (have) identified internal bus command as incorrect
Fault code 4: Internal bus data fault	
0	Data fault on internal bus or Internal bus interruption on coupler
n* (n>0)	Internal bus interrupted after I/O module n, passive clamps such as supply modules are not counted
Fault code 5: Register communication fault	
n*	Internal bus fault during register communication with the I/O module n, passive clamps such as supply modules are not counted
Fault code 6: Fieldbus fault	
0	Maximum Process image size is exceeded, too many I/O modules in the configuration
n* (n>0)	Programmed configuration does not agree with actual hardware configuration, fault at I/O module n
Fault code 7: not used	
0	Reserved
Fault code 8: not used	
0	Reserved

* The number of blink pulses (n) indicates the position of the I/O module. I/O modules without data are not counted (i.e. supply module without diagnosis)

Example: the 13th I/O module is removed.	
1.	The "I/O" LED generates a fault display with the first blink sequence (approx. 10 Hz).
2.	The first pause is followed by the second blink sequence (approx. 1 Hz). The "I/O" LED blinks four times and thus signals the fault code 4 (internal bus data fault).
3.	The third blink sequence follows the second pause. The "I/O ERR" LED blinks twelve times. The fault argument 12 means that the internal bus is interrupted after the 12th I/O module.

3.1.8 Technical Data

System data	
Number of nodes	110
Transmission medium	shielded Cu cable3 x 0,25 mm^2
Max. length of bus line	40 m ... 1000 m (baudratenabhängig / kabelabhängig)
Baud rate	10 kBaud ... 1 Mbaud
Buscoupler connection	- 5-pole male connector, series 231 (MCS) female connector 231-305/010-000 is included (750-347) - 1 x D-SUB 9; plug connector (750-348)
Standards and approvals	
UL	E175199, UL508
Conformity marking	CE
Technical data	
Max. number of I/O modules	64
Fieldbus	
Input process image	max. 32 bytes
Output process image	max. 32 bytes
No. of PDO	5 Tx / 5 Rx
No. of SDO	2 Server SDO
Communication profile	DS-301 V4.0
Device profile	DS-401 V2.0
	programmable error response
COB ID Distribution	SDO, standard
Node ID Distribution	DIP switch
Other CANopen Features	NMT Slave
	Minimum Boot-up
	Variable PDO Mapping
	Emergency Message
	Life Guarding
Configuration	via PC or PLC
Voltage supply	DC 24 V (-15 % ... + 20 %)
Input current$_{typ.}$ at nominal load	260 mA at 24 V
Efficiency of the power supply $_{Type.}$ at nominal load	80 %
Internal power consumption	350 mA at 5 V
Total current for I/O modules	650 mA at 5 V

Dimensions (mm) W x H x L	50 x 65* x 97 (* from top edge of mounting rail)
Weight	ca. 120 g
EMC interference resistance	acc. EN 50082-2 (96)
EMC interference transmission	acc. EN 50081-2 (94)

4 I/O modules

The detailed description of the fieldbus independent I/O modules of the WAGO-I/O-SYSTEM 750 is not contained in the excerpt of the present manual.

Further information
For information on the bus modules, please refer to the manual of the standard coupler or to the corresponding data sheets.
These data sheets are available in the Internet under:
www.wago.com/wagoweb/documentation/navigate/nm0d___d.htm.

5 CANopen

5.1 Description

CAN (Controller Area Network) was developed in the mid-eighties for data transmission in automobiles. The CAN specification defines the Data Link Layer which is the physical and data backup layer. The message structure is exactly described, however, nothing is said regarding the Application Layer. CAL, in the contrary, describes the Application Layer or the Meaning of the transmitted data. CAL is a general descriptive language for CAN networks and provides a large number of communication services.

CANopen is a networking concept based on the serial bus system CAN. CANopen is defined as a uniform application layer by the DS 301 specifications of the CIA (CAN in automation).

The network management provides a simplified start-up of the network. This network can be extended by the user as desired.

CAN is a Multimaster bus system. In contrast to other fieldbus systems, the modules connected to the bus are not addressed but the messages identified. Whenever the bus is free, the subscribers are allowed to send messages. Bus conflicts are solved in that the messages are assigned a certain priority. This priority is defined by the COB ID (Communication Object Identifier) and is clearly assigned to a communication object. The smaller the assigned identifier, the higher the priority. This also allows communication without the bus master group.

Each bus subscriber is solely decisive as to the point in time of data transmission. However, there is also a possibility to request other bus subscribers to send data. This request is performed via the so-called remote frame.

The CANopen specification (DS 301) defines the technical and functional features used to network distributed field automation devices.

Further information
CAN in Automation (CiA) provides further documents for their members in the Internet under:
can-cia.de

5.2 Network Architecture

5.2.1 Transmission Media

5.2.1.1 Type of Cable

A bus medium forms the basis for the physical connection of CAN.
With CAN, both the bus coupling and the bus medium are specified according to ISO 11898 (CAN High-Speed).

According to the cable specification, the Twisted-Pair medium (shielded cables twisted in pairs) with a wave resistance of 108...132 Ohm is recommended.
Twisted-Pair is low priced, convenient to use and permits simple bus type wiring.

The WAGO CANopen fieldbus nodes are intended for wiring using shielded copper wire (3x0.25 mm²).

Two important points have to be taken into consideration when designing the electrical bus medium:

- the maximum bus length and
- the required conductor cross section.

5.2.1.2 Maximum Bus Length

The length of the bus is mainly limited by the signal running time and must, therefore, be adapted to the Baud rate:

Baud rate	Bus length
1 Mbit/s	30 m
800 kbit/s	50 m
500 kbit/s	100 m
250 kbit/s	250 m
125 kbit/s	500 m
≤ 50 kbit/s	1000 m

Table 5-1: Maximum bus length dependent on the set Baud rate

5.2.1.3 Required Conductor Cross Section

The conductor cross section depends on the conductor length and has to be selected according to the number of nodes connected.

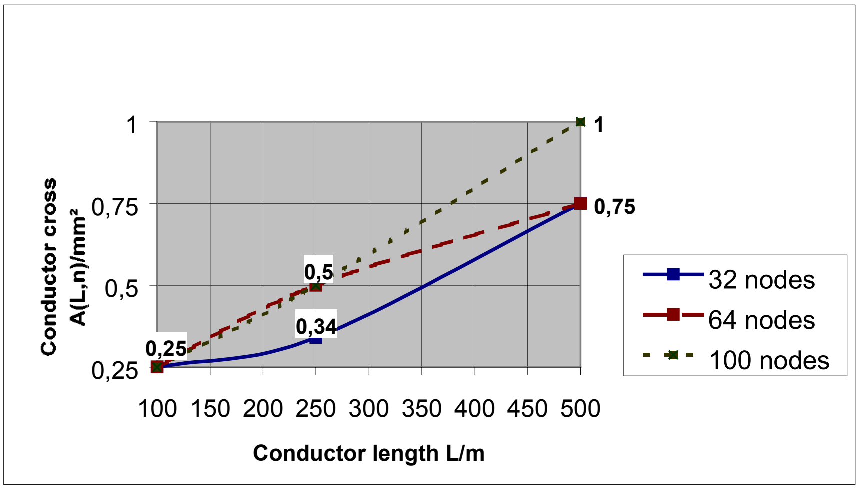


Fig. 5-1: Conductor cross section depending on the conductor length and the number of nodes

5.2.2 Cabling

The connection of a WAGO fieldbus node to the CANopen bus cable is made by the corresponding supplied plug connector.

750-347	5-pole plug (Multi Connector 231) (is included in delivery)
750-348	9-pole D-SUB plug connector (The connector 750-963 is not included in delivery).

Further information
For the respective plug assignment for the fieldbus connection see chapter: 3.1.2.3 "Fieldbus Connection".

For cabling with a shielded copper cable (3x0.25 mm²), the respective plug is assigned with the CAN_High, CAN_Low and CAN_GND connections. CAN_High and CAN_Low are two physically different bus levels. CAN_GND is the common reference potential..

The conductor shield of the cable can be routed on the connection drain, which is terminated with 1 MΩ as against the ground or PE (carrier rail contact). A low ohmic connection of the shield to the PE can only be made externally (i.e. by means of a supply module). The aim is for a central PE contact for the entire CANopen bus conductor screening.

Note
WAGO offers the screen connection system (series 790) for an optimum connection between fieldbus cable screening and functional earth.

Each CAN node forms the differential voltage U_{Diff} with: $U_{Diff} = U_{CAN\_High} - U_{CAN\_Low}$. from the bus levels CAN_High and CAN_Low.
The different signal transmission offers the advantage of being immune to common mode interference and ground offset between nodes.

If the bus level is in the recessive status, the voltage between CAN_Low and CAN_GND is 2.5 V and also 2.5°V between CAN_High and CAN_GND. This means that the differential voltage is 0 V.

If the bus level is in the dominant status, the voltage between CAN_Low and CAN_GND is 1.5 V and 3.5°V between CAN_High and CAN_GND. Then differential voltage is approx. 2 V.

Note

When connecting subscribers, ensure that the data lines are not mixed up.

At its conductor ends, the bus cable must always be connected with a matching resistor of 120 Ohm to avoid reflections and, as a result, transmission problems.
This is also required for very short conductor lengths.

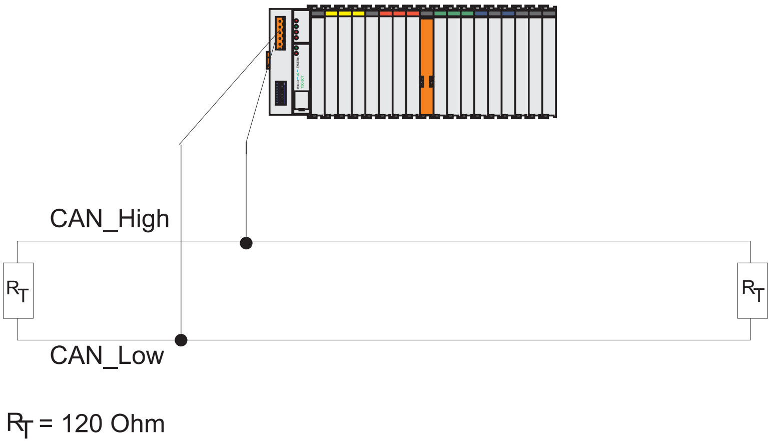


Fig. 5-1: Connection principle of a WAGO fieldbus node to the CAN bus g012402x

Before starting the buscoupler on the network, the installation should be checked. The physical connection can be checked in the CAN fieldbus with an ohmmeter at any place. You have to remove all connections to other devices except for the terminating resistors.

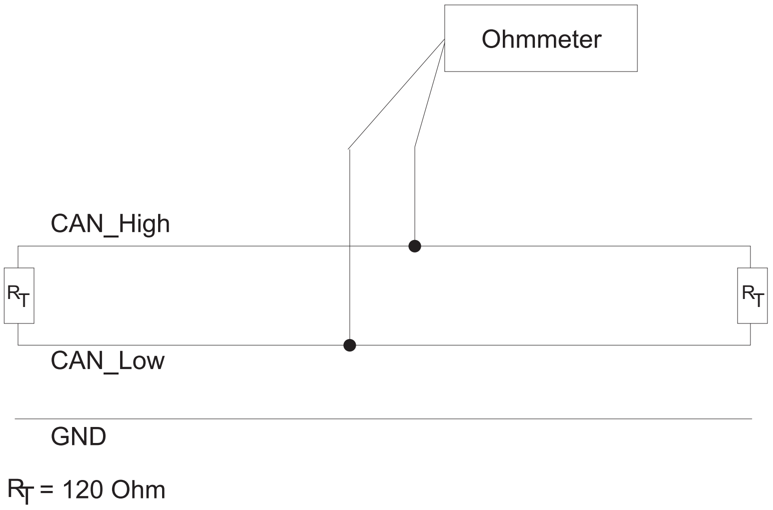


Fig. 5-2: Measuring principle to check the CAN bus prior to wiring g012445x

Measurement between	Value	Meaning
GND and CAN_L	infinite	ok.
	0	Short-circuit between GND and CAN_L
GND and CAN_H	infinite	o.k.
	0	Short-circuit between GND and CAN_H
CAN_L and CAN_H	ca. 60 Ω	o.k., 2 terminal resistors in the bus
	ca. 120 Ω	Only 1 terminal resistor in the bus
	< 50 Ω	More than 2 terminal resistors in the bus

The CAN bus is 2-wire bus and bus error management can detect a cable break or a short-circuit by the asymmetric operation.

Further information
The CiA provides documents regarding specifications, especially cable specifications in the Internet under:

http://www.can-cia.de

5.2.3 Network Topology

To build a simple CANopen network, you need a master (PC with a CANopen fieldbus PCB card), a connection cable and a DC 24 V power pack to ensure the power supply in addition to a CANopen fieldbus node.

The CANopen network is constructed as a line structure with matching resistors (120 Ohm).

In systems having more than two stations, all subscribers are wired in parallel. The maximum length for a conductor branch should not exceed 0.3 m.

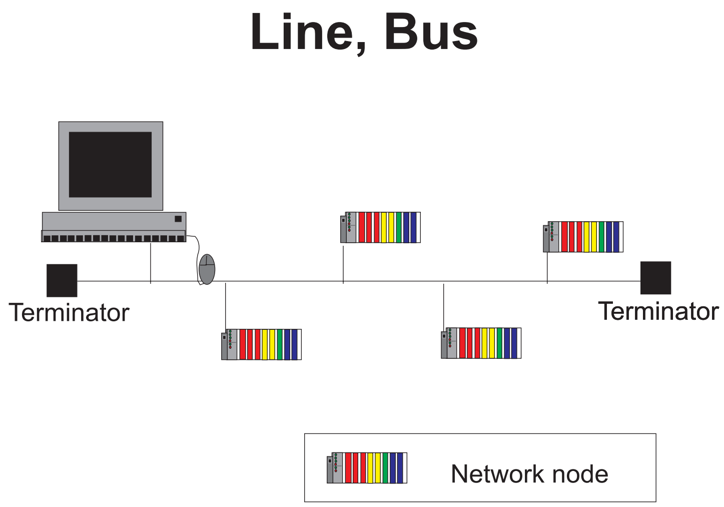


Fig. 5-2: Bus topology of a CANopen network g012430e

All net subscribers communicate at the same Baud rate. The bus structure permits coupling in and out without side effect of stations, or the step-by-step start-up of the system.
Later extensions have no influence on stations already in operation. The system automatically detects when a subscriber fails or is newly added to the net.

Also branches from the line-shaped bus and as such the establishment of hierarchic net structures are possible via router nodes.

Repeaters can be used to increase the maximum possible number of nodes to 110 and to enlarge the network spatial extension (bus length). Although the network spatial extension depends on the transmission rate, CAN can also be used for spatially extended networks. The data rates achievable are of the same order as with other bus systems. However, the maximum possible cable length is reduced per repeater by 20 - 30 m due to the signal delay.

5.2.4 Interface Modules

In a network, all WAGO CANopen fieldbus nodes operate as slaves. The master operation is taken over by a central control system, such as PLC, NC or RC. The connection to fieldbus devices is made via interface modules.

Interface modules for programmable logic controls (PLCs) and PC interface PCBs are offered by various manufacturers.

5.2.5 Configuration Software

So that a connection between SPS and the field bus devices becomes possible, the interface modules must be configured with the individual station data.

The software to project engineering, start-up and diagnosis is included in delivery of the interface modules or PC interface PCBs or as accessories of other manufacturers available.

5.3 Network Communication

With CANopen, data transmission, the triggering of events, signalling of error states etc. takes place by means of communication objects. For this purpose, each communication object is assigned a clear COB-ID (Communication Object Identifier) in the network.

The parameters required for the communication objects as well as the parameters and data of the CANopen subscribers are filled in the object directory.

Type and number of objects supported by the node depend on the individual fieldbus coupler.

In addition to several special objects, i.e. for the network management (NMT), for synchronization (SYNC) or for error messages (EMCY), the communication profile contains the two object types PDO and SDO.

The PDOs (process data objects) are used for the transmission of real time data, and the SDOs (service data objects) permit access to the object directory both for reading and writing.

5.3.1 Communication Objects

5.3.1.1 Process Data Object - PDO

PDOs contain real time data with high priority identifiers. The data telegrams consist of a maximum of 8 bytes and can be interchanged among the individual sub-assemblies, as required. This data exchange can be optionally event controlled or performed in a synchronized manner. The event controlled mode allows the bus load to be drastically reduced permitting a high communication capacity at a low Baud rate. However, the various modes can also be processed as a mix (see chapter 5.3.4.2.20 "Object 0x1400– 0x1404, Receive PDO Communication Parameter").

5.3.1.1.1 PDO Protocol

This protocol is used to transmit data from/to the bus coupler without protocol overhead. PDOs consist only of the CAN identifier and the data field. No further protocol information is contained in a PDO. The contents of the data are defined by the mapping parameters and the transmission type by the communication parameters.

A differentiation is made between RxPDO (receipt PDO) and TxPDO (transmit PDO).

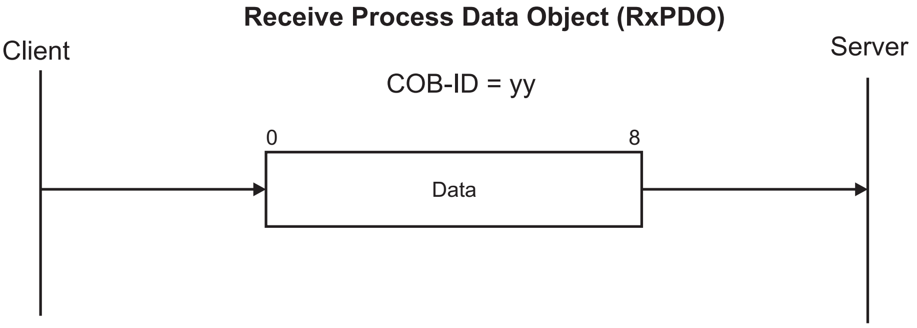


Fig. 5-1: RxPDO g012403x

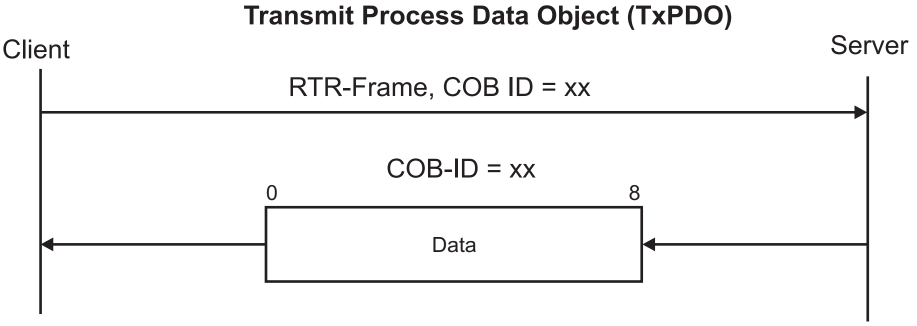


Fig. 5-2: TxPDO g012404x

5.3.1.2 Service Data Object - SDO

The SDOs can be used to read and/or write entries in the object directory. In this manner, a CANopen subscriber can be fully configured. The default SDO is pre-assigned with a low priority identifier. The transmitted data has to be distributed to several messages if it exceeds 4 bytes.

5.3.1.2.1 SDO Protocol

A specific protocol overhead that is indispensable for transmission and contains the command specifier, the index and the sub-index of the entry to be read/written.

5.3.1.2.1.1General Design

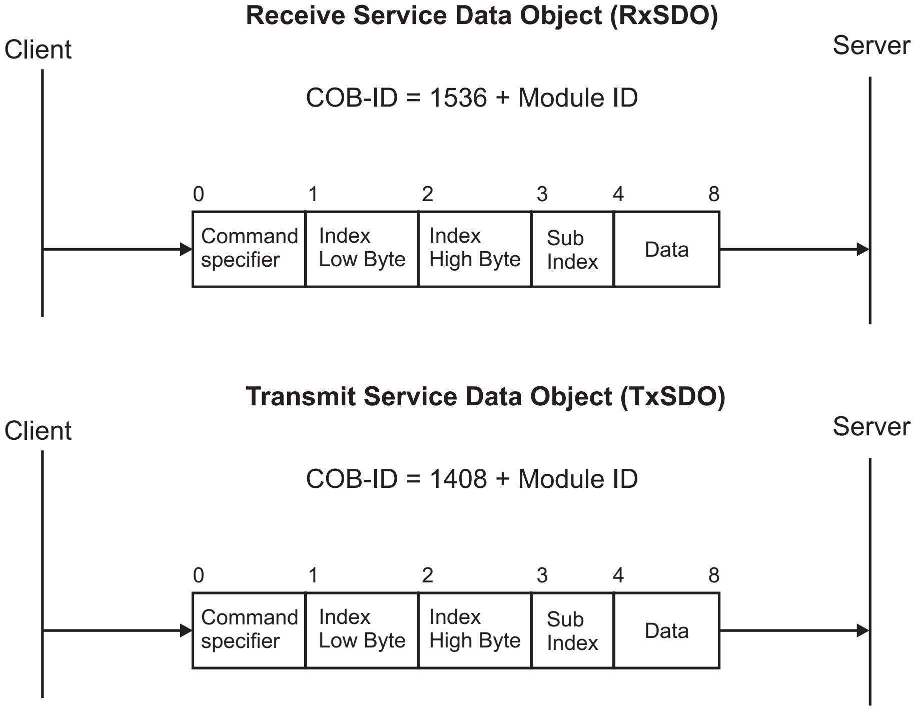


Fig. 5-3: SDO Protocol g012405x

5.3.1.2.1.2Download SDO Protocol

This protocol is used to write data from the master into the bus coupler.

5.3.1.2.1.2.1 Initiate SDO Download

This protocol is used to initiate the data transmission from the master to the bus coupler. When transmitting data of max. 4 bytes, these are also transmitted within the protocol.

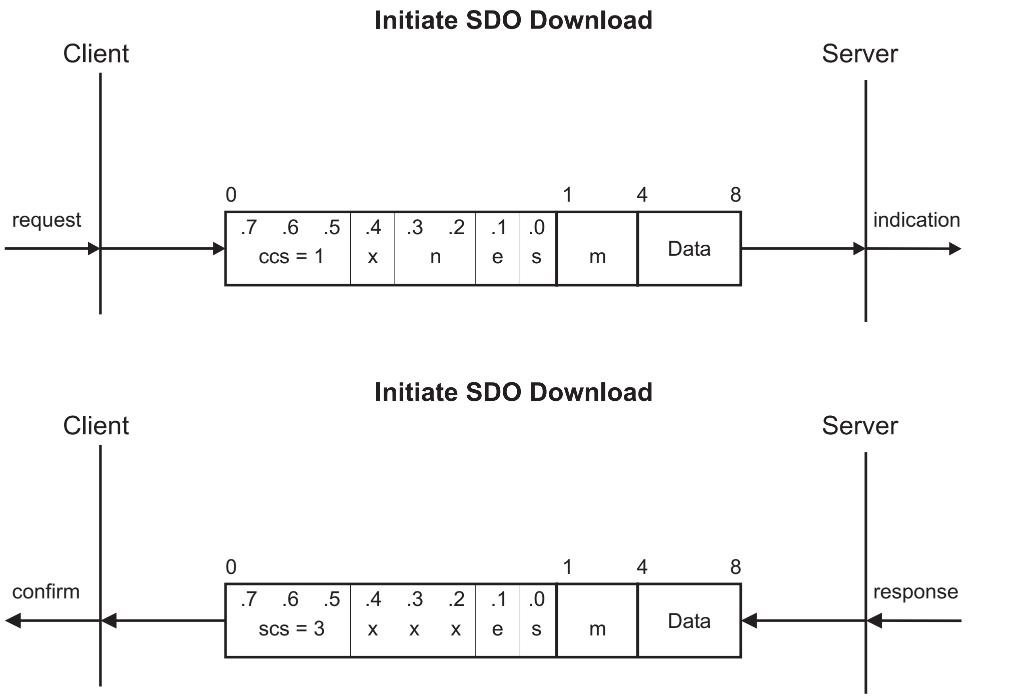


Fig. 5-4: Initiate SDO

g012406x

ccs:	Client command specifier	1: initiate download request
scs:	Server command specifier	3: initiate download response
n:	only valid, if e =1 and s =1, otherwise 0.	If n is valid, it displays the number of bytes which do not contain any data. Example: 3 data bytes, e = 1 and s = 1, n = 4 - 3 = 1
e:	transfer type	0: normal transfer, Anzahl der zu schreibenden Bytes >= 5 Byte
		1: expedited transfer, , number of the bytes to be written >= 5 byte
s:	size indicator	0: data set size is not displayed
		1: data set size is displayed s is always 1
m:	multiplexor	Index und 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 further use of CiA
		e = 0, s = 1: d contains the number of bytes for download
		Byte 4 contains the LSB and byte 7 contains the MSB.
		e = 1: d contains the data
X:		not used, always 0
reser- ved:		Reserved for further use of CiA

5.3.1.2.1.2.2 Download SDO Segment

This protocol is used to transmit more than 4 data, i. e. this follows after fully processing the „Initiate SDO Download Protocol" which initiates the data transmissions.

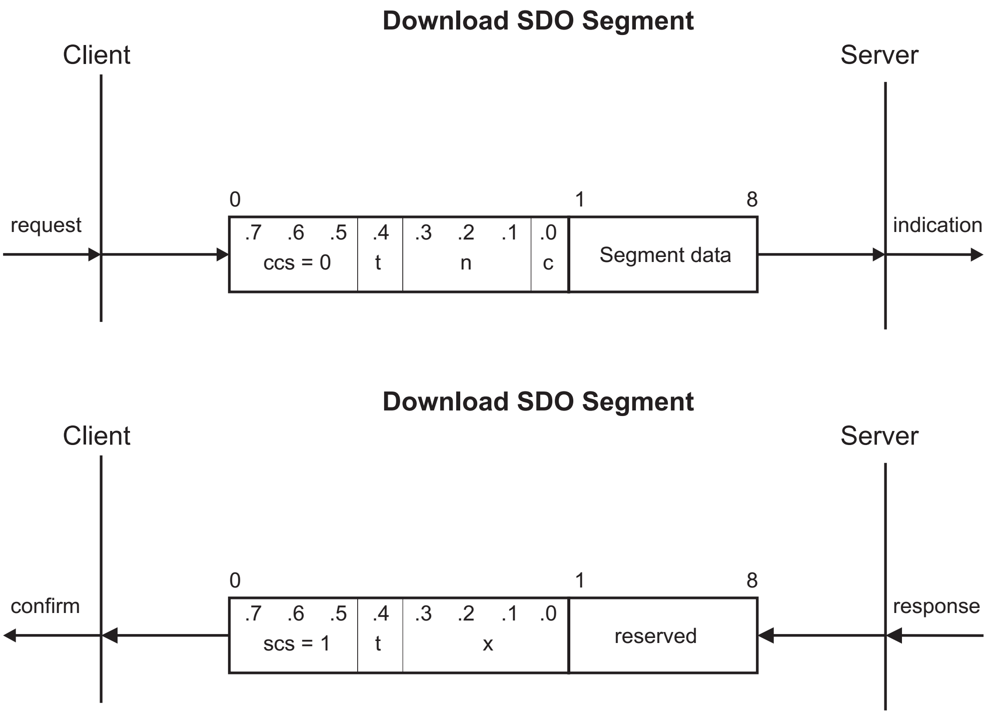


Fig. 5-5: Download SDO Segment g012409x

ccs:	client command specifier	0: download segment request
scs:	server command specifier	1: download segment response
seg-data	Contains the data to be transmitted.	The meaning of the data is determined by the application.
n:		Displays the number of bytes not containing any data. n is 0 if no segment size is displayed.
c:	Indicates whether or not a download is necessary for further data.	0: There is more data to be downloaded.
		1: There is no more data to be downloaded.
t:	Toggle Bit	This bit must be able to toggle for each segment for which a download is made. The first segment zeroes the toggle bit. The toggle bit is identical both for the enquiry and the reply message.
X:		Not used, always 0
reserved:		Reserved for further use of CiA

5.3.1.2.1.3Upload SDO Protocol

This protocol is used to read data out of the bus coupler.

5.3.1.2.1.3.1 Initiate SDO Upload

The data transmission from the bus coupler to the master is initiated with this protocol. When transmitting data of max. 4 bytes, these are also transmitted within the protocol.

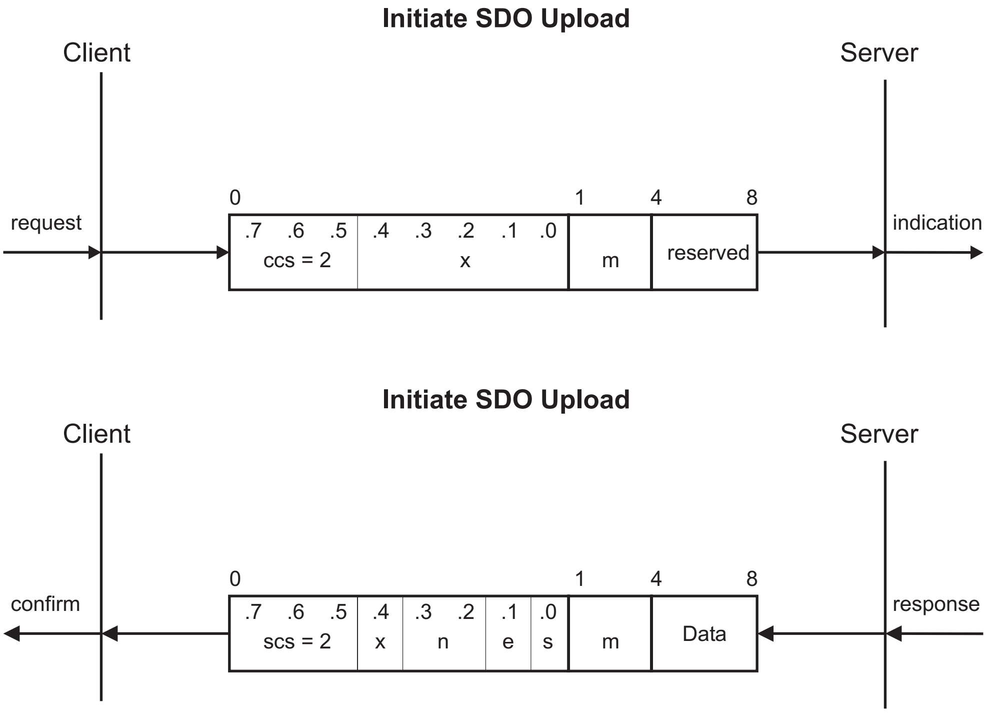


Fig. 5-6: Initiate SDO Upload

g012410x

ccs:	Client command specifier	2: initiate upload request
scs:	Server command specifier	2: initiate upload response
n:	is only valid if e = 1 and s = 1, otherwise 0.	If n is valid, it displays the number of bytes in d which do not contain any data. The bytes [8-n, 7] do not contain segment data.
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: the number of bytes to be transmitted is not displayed 1: the number of bytes to be transmitted is displayed (depending on the 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 further use of CiA
		e = 0, s = 1: d contains the number of bytes for download
		Byte 4 contains the LSB and Byte 7 contains the MSB.
		e = 1: d contains the data
X:		Not used, always 0
reserved:		Reserved for further use of CiA

5.3.1.2.1.3.2 Upload SDO Segment

This protocol is used if more than 4 data is transmitted, i.e. this follows after fully processing the „Initiate Upload Protocol" which initiates the data transmissions.

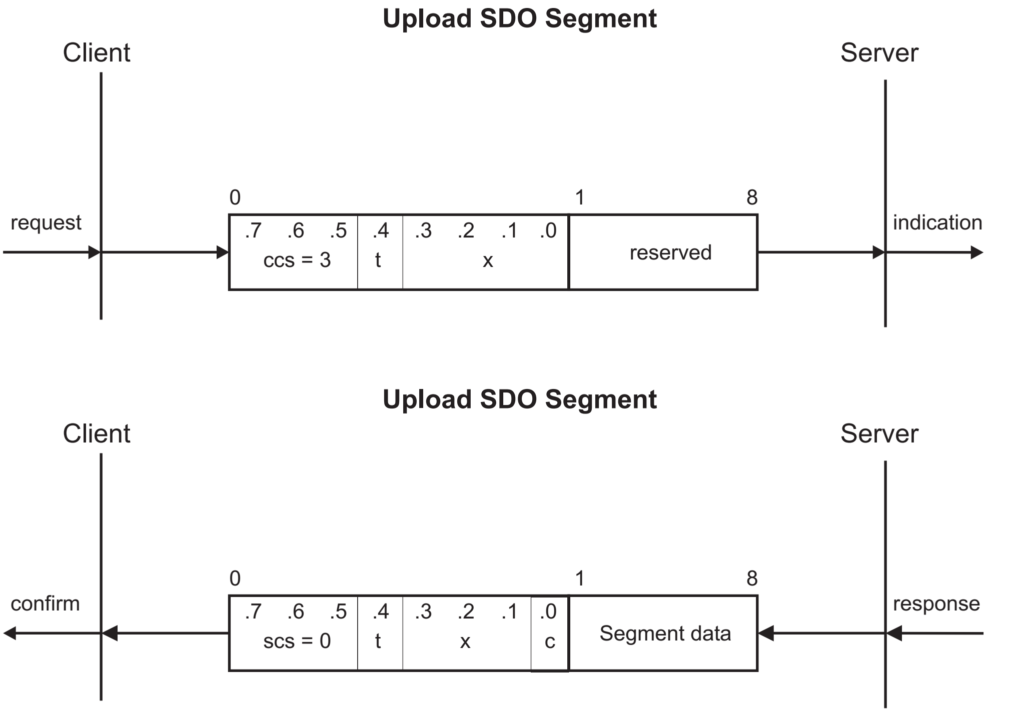


Fig. 5-7: Upload SDO Segment

g012411x

ccs:	Client command specifier	3: download segment request
scs:	Server command specifier	0: download segment response
t:	Toggle bit	This bit has to change for each segment for which an upload is made. The toggle bit has to be zeroed for the first segment. The toggle bit is identical both for the enquiry and the reply.
c:	Indicates whether further segments are present for the upload	0: there are more segments to be uploaded
		1:there are no more segments for uploading

seg-data	Contains the data to be transmitted.	The meaning of the data is determined by the application.
n:		Displays the number of bytes which do not contain data. Bytes [8-n, 7] do not contained data. N is 0 if no segment size is displayed.
X:		Not used, always 0
reserved:		Reserved for further use of CiA

5.3.1.2.1.4Abort SDO Transfer

This protocol is used in the event of errors occurring during transmission.

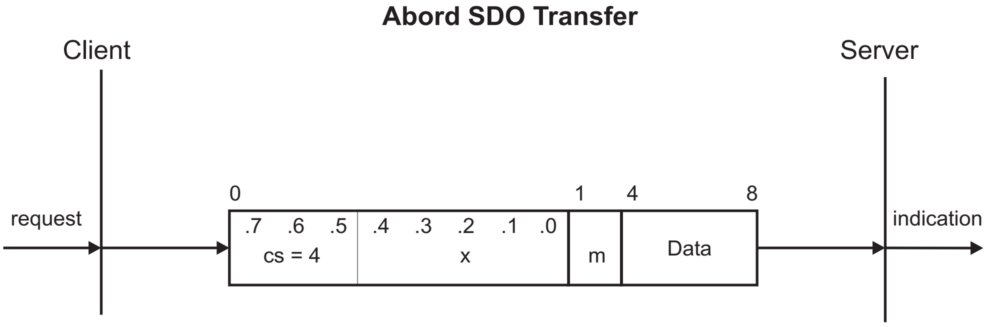


Fig. 5-8: Abort SDO Transfer

g012412x

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 about the reasons for the abort.

5.3.1.2.2 SDO Examples

The following are 4 SDO examples, the data is being displayed in hexadecimal. These examples show the handling of SDOs on the CAN message level and can be used if the SDO protocol is to be implemented on a CAN card.

A message is subdivided into 4 columns:

1. column	Direction	M->BC = message is sent by the master to the bus coupler. BC->M = message is sent by the bus coupler to the master.
2. column	CAN Identifier	

3. column	Frame Type	D = Data frame
		R = RTR frame

4. column Data Data bytes of the CAN message
A maximum of 8 data bytes can be transmitted in a CAN message. The individual bytes are separated by spaces. Entries having the value XX have no meaning, but must be existing. The values should be zeroed for a better understanding. Entries in the reply from the bus coupler having the value DD contain data, which are dependent on the configuration.

5.3.1.2.2.1Example 1:
Read Index 0x1000 Sub-Index 0; Device Type

Index 0x1000 returns 4 bytes. The expedited transfer mode is used for transmission.

Direction	CAN Id	Frame Type	Data byte 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 rotation:
0x0191 = 401 Device Profile Number

Data bytes 6 and 7: DD 00 Sequence Low Byte, High Byte rotation

5.3.1.2.2.2Example 2:
Read Index 0x1008 Sub-Index 0; Manufacturer Device Name

Index 0x1008 returns more than 4 bytes. The normal transfer mode is used for transmission in which case 2 messages per mode are transmitted.

Direction	CAN Id	Frame Type	Data byte 0-7
M->BK	0x601	D	0x40 08 10 00 XX XX XX XX
BK->M	0x581	D	0x41 08 10 00 07 00 00 00
M->BK	0x601	D	0x60 XX XX XX XX XX XX XX
BK->M	0x581	D	0x01 37 35 30 2D 33 34 37

Result:

The first reply from the bus coupler informs the master of the number of data to be transmitted (0x00000007 Byte). In the second message, the bus coupler supplies the article number in the ASCII format (hex representation) „750-347“.

5.3.1.2.2.3Example 3: Read Index 0x2000 Sub-Index 1; First 8 bit digital input block

The signals of the digital input modules are saved in index 0x2000. 8 bits each are assigned to a group and can be read as from sub index 1. In this example, the input value of the first 8 bit group is read via an SDO message.

Direction	CAN Id	Frame Type	Data byte 0-7
M->BK	0x601	D	0x40 00 20 01 XX XX XX XX
BK->M	0x581	D	0x4F 00 20 01 02 XX XX XX

Result:

In the 5th byte of the CAN message, the bus coupler returns the status of the first group of 8 bits. In this case the 2nd bit is set. Bytes 5-7 are without meaning.

5.3.1.2.2.4Example 4: Write Index 0x2100 Sub-Index 1; First 8 bit digital output block

The output values of the digital output modules are saved in index 0x2100. 8 bits each are assigned to a group and can be read and written as from sub index 1. In this example, the value 0xFF is written into the outputs of the first 8 bit digital output group.

Direction	CAN Id	Frame Type	Data byte 0-7
M->BK	0x601	D	0x2F 00 21 01 FF XX XX XX
BK->M	0x581	D	0x60 00 21 01 XX XX XX XX

Result:

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

5.3.1.3 Synchronization Object - SYNC

These objects allow the synchronization of all network subscribers. Corresponding configuration of the PDOs can initiate the network subscribers to process their input data or to update the outputs upon the arrival of a SYNC object.

In this manner cyclical transmission of a SYNC object ensures that all network subscribers will process their process data simultaneously.

5.3.1.3.1 SYNC Protocol

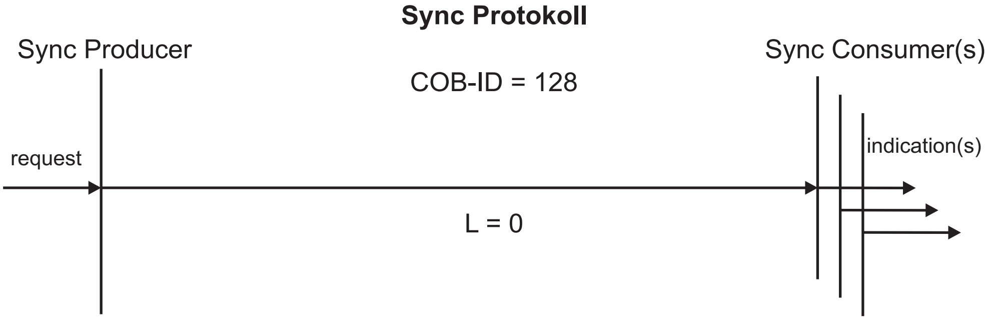


Fig. 5-9: SYNC Protocol

g012413x

5.3.1.4 Emergency Object (EMCY)

Emergency objects are triggered by an internal error situation such as i.e. a module is removed during operation, or a module signals an error. The bus coupler then sends an emergency object to all connected devices (Broadcast), to broadcast the error occurred. The informed bus subscribers can then react accordingly by suitable error correction measures.

5.3.1.4.1 EMCY Protocol

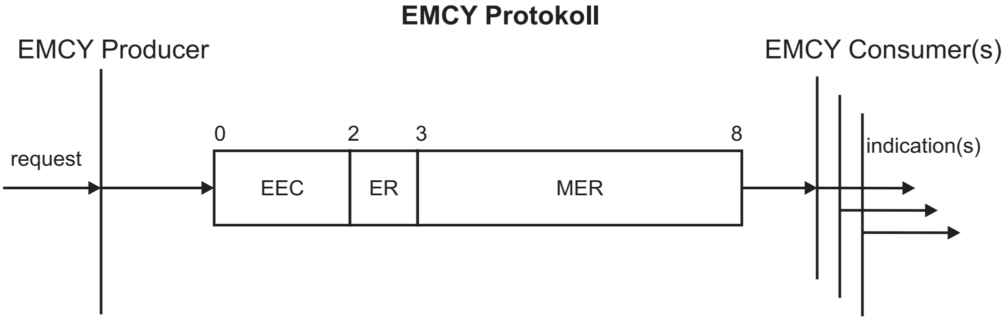


Fig. 5-10: EMCY Protocol

g012414x

5.3.2 Communication States of a CANopen Coupler

5.3.2.1 CANopen State Diagram

The status diagram described in the following figure shows the individual communication states and possible transitions related to the CAN communication.

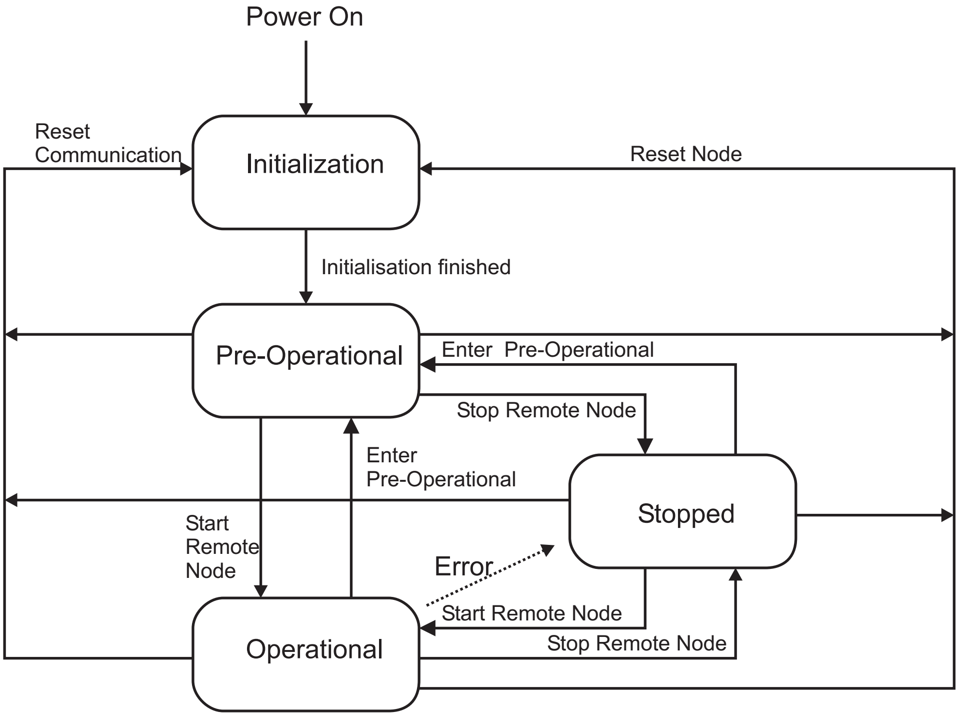


Fig. 5-11: State diagram of the fieldbus coupler g012422x

5.3.2.2 INITIALIZATION

Following a power On or a reset (module ID unequal 0), the bus coupler is automatically in the INITIALIZATION status. In this status, the bus coupler performs a self-check to check all functions of its components and the communication interface. The process image is created on the basis of the connected modules and a possibly stored configuration, and the object directory initialized. If no errors are detected during the initialization phase the bus coupler automatically changes to the pre-operational status. If errors were found, a change to the STOP status takes place.

During initialization, the I/O-LED starts blinking orange and changes to red at its higher frequency. If initialization and the change to the pre-operational status have been successfully completed, the I/O LED is lit green and the RUN LED blinks. If errors have occurred (i.e. no end module connected) the I/O LED indicates the error type by a red blinking sequence (see LED status display). In this case, the STOP LED is lit red.

5.3.2.3 PRE-OPERATIONAL

In this status, communication can be made via SDOs. Communication via PDOs is not possible. The SDOs allow for reading and writing in the entries of the object directories permitting for instance to re-configure the bus coupler by means of the configuration tool. Mapping, bus coupler parameters, IDs etc. can in this manner be adapted to the required conditions. The newly configured configuration can be saved in flash.

A change from the pre-operational status to the operational status is performed by means of the NMT service Start_Remote_Node.

In the pre-operational status the I/O LED is lit green and the RUN LED blinks.

5.3.2.4 OPERATIONAL

This status allows communication via SDOs and PDOs, it does, however, not allow different configurations. It is, for instance, not allowed to change the COB ID in the presence of a valid PDO. For a detailed description, please refer to the corresponding entries in the object directory.

The change from the operational status to the pre-operational status is performed with the NMT service Enter_Pre_Operational_State.

In the operational status, the I/O and the RUN LED are lit.

5.3.2.5 STOPPED

The Stopped status reflects an error status. This is the case if the NMT service Stop_Remote_Node was received or if a fatal internal error has occurred (i.e. module was removed during operation).

This status does not allow communication via SDOs or PDOs. Only the NMT services and the Node Guarding/Heartbeat (if activated) are performed.

You can quit the Stopped status via the NMT services Start_Remote_Node_Indication, Enter_Pre_Operational_State and Reset_Node.

The Stop LED is lit in the Stopped status.

5.3.3 Network Management Objects

5.3.3.1 Module Control Protocols

The NMT master can use these protocols to check the status of the NMT slave. The following states are defined: INITIALIZING, PRE-OPERATIONAL, OPERATIONAL and STOPPED. It is possible to change the status of all nodes with one command or to change the status of each node individually.

5.3.3.1.1 Start Remote Node

This service is used to change the NMT Slave (bus coupler) status to OPERATIONAL.

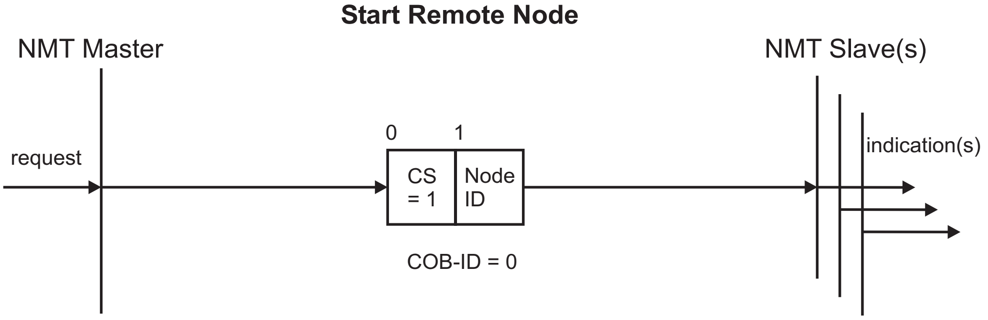


Fig. 5-12: Start Remote Node g012415x

Node ID = 0: The state of all existing nodes is changed to OPERATIONAL.

5.3.3.1.2 Stop Remote Node

This service is used to change the NMT Slave (bus coupler) status to STOPPED.

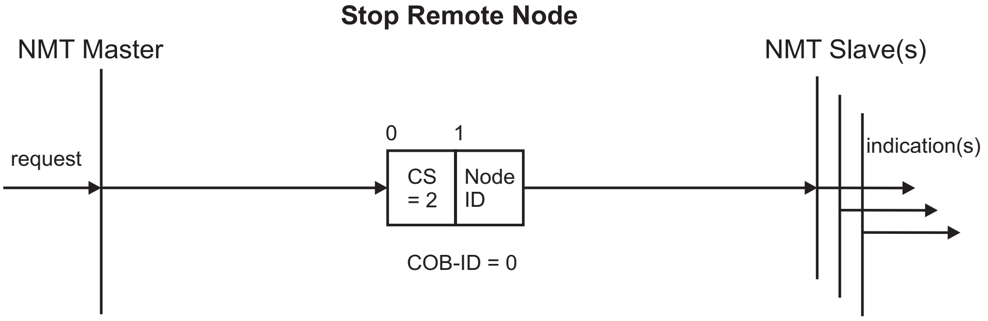


Fig. 5-13: Stop Remote Node g012416x

Node ID = 0: The state of all existing nodes is changed to STOPPED.

5.3.3.1.3 Enter Pre-Operational

This service is used to change the status of the NMT Slave (bus coupler) to PRE-OPERATIONAL.

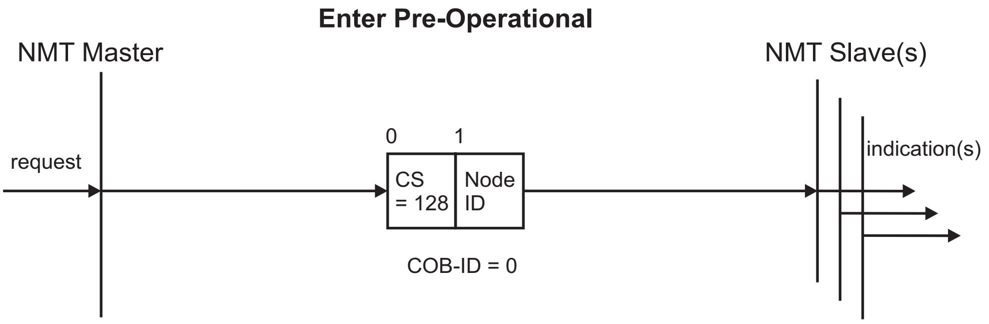


Fig. 5-14: Enter PRE-OPERATIONAL g012417x

Node ID = 0: The state of all existing nodes is changed to PRE-OPERATIONAL.

5.3.3.1.4 Reset Node

In this service a reset is performed with the NMT Slave (bus coupler).

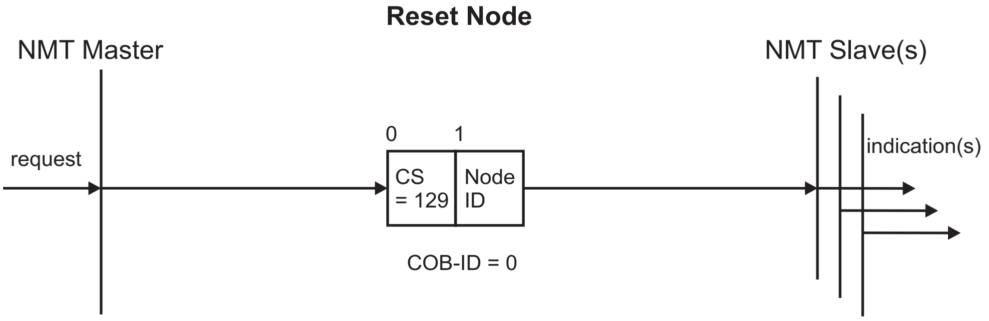


Fig. 5-15: Reset Node g012418x

Node ID = 0: a reset of all existing nodes is performed.

5.3.3.2 Error Control Protocols

These protocols permit the detection of possible errors in the network. In this manner the master can check whether a node is still in the status defined by it or if it has changed to a different status, for instance following a reset.

5.3.3.3 Node Guarding Protocol

By means of Node Guarding, the NMT slave is cyclically requested via an RTR frame to send its current status. Additional toggling of a bit detects whether or not the NMT slave still operates correctly.

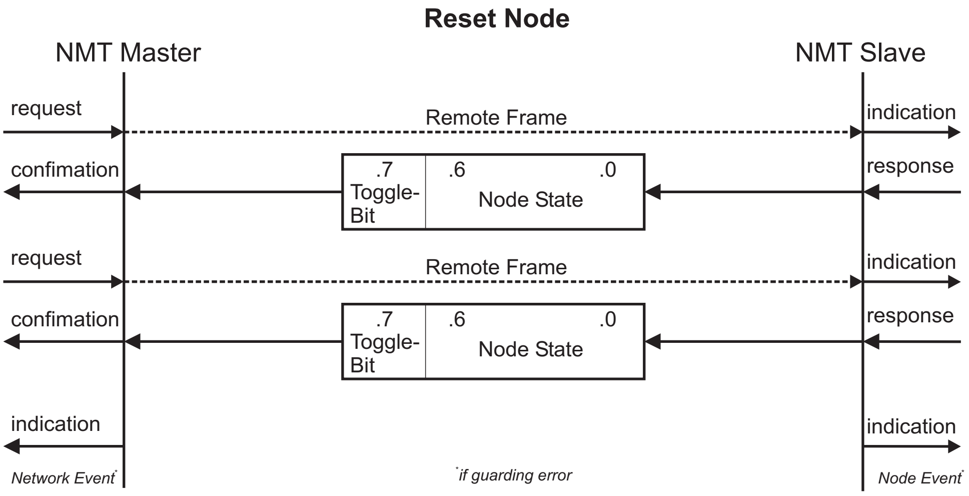


Fig. 5-16: Node Guarding Protocol g012419x

5.3.3.4 Heartbeat Protocol

This protocol allows monitoring without RTR frames. A heartbeat generator cyclically generates a heartbeat message received by n subscribers. The heartbeat message contains the coding of the current generator status.

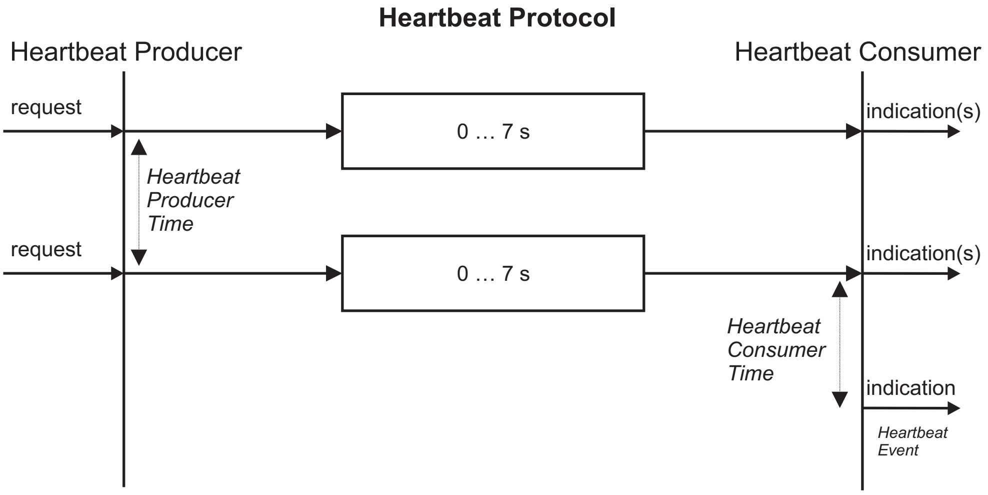


Fig. 5-17: Heartbeat Protocol g012420x

5.3.3.5 Bootup Protocol

This protocol shows that the NMT slave has changed its status from INITIALIZING to PRE-OPERATIONAL. This is performed after a hardware/software reset or following the service reset code.

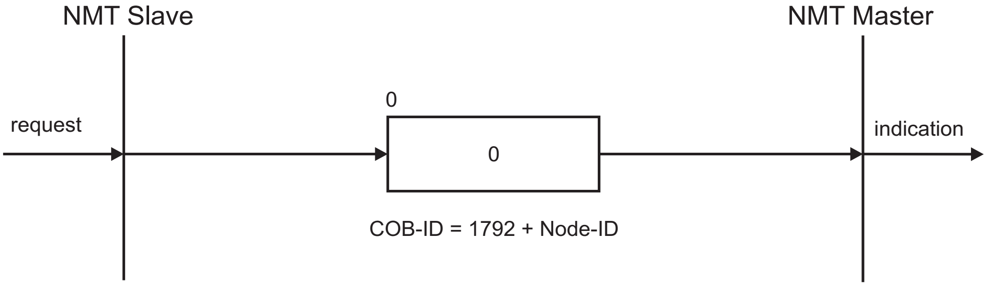


Fig. 5-18: Bootup Protocol g012421x

5.3.4 Object Directory

The object directory is the central point of a CANopen subscriber where all configuration information and data is stored and can be polled. The directory organization is by means of tables and contains three areas of CANopen objects:

- Communication Profile Area (Index 0x1000 – 0x1FFF)
  This profile contains all parameters relevant for CANopen communication. This area is identical for all CANopen subscribers.

- Manufacturer Specific Profile Area (Index 0x2000 – 0x5FFF)
  In this profile, each manufacturer can implement his own company specific objects.

- Standardised Device Profile Area (Index 0x6000 – 0x9FFF)
  This profile contains all objects which are assisted by a certain device profile. The bus coupler assists the device profile DS-401 (Device Profile for Generic I/O Modules)

In the object directory, a logical addressing scheme is used for the access to communication and device parameters, data and functions. Each entry into the directory is identified by a 16 bit index which indicates the row address of the table. A maximum of 65536 entries are permitted.

If an object is composed of several components, the components are identified by means of an 8 bit sub-index. The sub-index indicates the individual column address of the table allowing a maximum of 256 entries.

If the index only consists of simple variables (UNSIGNED8, UNSIGNED16,...) the sub-index is always zero.
For more complex developed objects (Array, Records,...) the sub-index 0 indicates the max. number of the following sub-indexes. The data is coded in the following sub-indexes.

Each entry consists of:

- an object name describing the object function,
- a data type attribute defining the data type of the entry, and
- an access attribute indicating whether the entry is only read, only write or read and write.

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 standardized device profiles
0xA000 – 0xFFFF	Reserved

Tab. 5-1: Structure of the CANopen object directory

The object directory structure is designed for the worst case. Object entries that cannot be used because of the connected module configuration are deactivated.

5.3.4.1 Initialization

The connected module configuration is determined following power On.

If a customer-specific configuration was saved and if the currently connected module configuration coincides with the one last saved, the object directory with this saved configuration will be initialized.

In every other case, object directory will be assigned a default configuration.

5.3.4.1.1 Default Configuration

5.3.4.1.1.1Initialization Communication Profile Area

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

- **Entry of the default mapping parameters:**
  Pre-assignment of the mapping parameters depends on the device profile used. The bus coupler assists the DS 401 profile, and as such the process described there is used. The first 4 Rx-/TxPDOs are defined as default PDOs. If more inputs/outputs exist at the bus coupler than can be covered with the default PDOs, from Rx-/TxPDO 5 all remaining I/Os are entered. First all digital, then all analog I/Os are entered. If more than 64 digital I/Os per input/output are available, a continuation is only made with PDO 5, even if no analog modules exist. PDO 2 to 4 will then remain unused. Furthermore, only one data type is used as a default entry for a PDO, in other words, if a 3 byte and a 4 byte module exists, then a default is entered for each of the 2 PDOs.

- **1. RxPDO:**
  contains maximum the first 8x8 digital outputs. If no digital outputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

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

-

- **2. RxPDO:**
  contains max. the 1st to 4th 16 bit analog output. If no 16 bit analog outputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

Idx	S-Idx	Description	Default Value
0x1601	0	Number of mapped objects	None, possible values: 0: no analog output 1..4: 1..4 analog outputs
	1	1. mapped 16 bit analog output	0x6411 01 10
	2	2. mapped 16 bit analog output	0x6411 02 10
	3	3. mapped 16 bit analog output	0x6411 03 10
	4	4. mapped 16 bit analog output	0x6411 04 10

- **3. RxPDO:**
  contains maximum the 5th to 8th bit analog output. If more than 4 16 bit analog outputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

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

- **4. RxPDO:**
  contains maximum the 9th to 12th 16 bit analog output. If not more than 8 16 bit analog outputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

Idx	S-Idx	Description	Default Value
0x1603	0	Number of mapped objects	None, possible values: 0: no analog output 1..4: 1..4 analog outputs
	1	9. mapped 16 bit analog output	0x6411 09 10
	2	10. mapped 16 bit analog output	0x6411 0A 10
	3	11. mapped 16 bit analog output	0x6411 0B 10
	4	12. mapped 16 bit analog output	0x6411 0C 10

- **1. TxPDO:**
  contains maximum the first 8x8 digital inputs. If no digital inputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

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

- **2. TxPDO:**
  contains maximum the 1st to 4th 16 bit analog input. If no 16 bit analog inputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

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

- **3. TxPDO:**
  be contains maximum the 5th to 8th 16 bit analog input. If not more than 4 16 bit analog inputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

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

- **4. TxPDO:**
  contains maximum the 9th to 12th 16 bit analog input. If not more than 8 16 bit analog inputs exist, the sub index 0 has the default value 0, and this PDO is not used with its default value.

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

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

- **Initialization Standardized Device Profile Area**
  All supported objects are initialized, as defined in the DS 401 standard.

5.3.4.2 Communication Profile Area

The following table contains all objects of the communication profile supported by the bus coupler.

Idx	Name	Type	Meaning	See on page
0x1000	Device Type	Unsigned32	Device Profile	92
0x1001	Error Register	Unsigned8	Errors are bit coded (DS401)	92
0x1003	Pre-defined Error Field	Array Unsigned32	Storage of the last 20 errors occurred	93
0x1005	COB-ID SYNC message	Unsigned32	COB-ID of the SYNC object	93
0x1006	Communication Cycle Period	Unsigned32	Max. time between 2 SYNC messages	**Fehler! Textmarke nicht definiert.**
0x1008	Manufacturer Device Name	Visible String	Device name	93
0x1009	Manufacturer Hardware Version	Visible String	Hardware version	94
0x100A	Manufacturer Software Version	Visible String	Software version	94
0x100C	Guard Time	Unsigned16	Time for "Life Guarding Protocol"	94
0x100D	Life Time Factor	Unsigned8	Life Time Factor	94
0x1010	Store Parameters	Array Unsigned32	max. number of store parameters	94
0x1011	Restore default Parameter	Array Unsigned32	Parameter to restore the default configuration	95
0x1014	COB-ID Emergency Object	Unsigned32	COB-ID for the emergency Object	96
0x1015	Inhibit Time EMCY	Unsigned32	Min. time between 2 EMCY messages	96
0x1016	Consumer Heartbeat Time	Array Unsigned32	Heartbeat monitoring time	97
0x1017	Producer Heartbeat Time	Unsigned16	Time between 2 generated Heartbeat messages	97
0x1018	Identiy Object	Record Identity	Device information	97
0x1029	Error Behaviour	Array Unsigned8	Status change in the event of an error	98
0x1200 to 0x1201	Server SDO Parameter	Record SDO Parameter	Parameter for the Server SDO	98
0x1400 to 0x1404	Receive PDO Communication Parameter	Record PDO Paramter	Communication parameter for the Receive PDO	98
0x1600 to 0x1604	Receive PDO Mapping Parameter	Record PDO Mapping	Mapping parameter for the Receive PDO	99
0x1800 to 0x1804	Transmit PDO Communication Parameter	Record PDO Paramter	Communication parameter for the Transmit PDO	100
0x1A00 to 0x1A04	Transmit PDO Mapping Parameter	Record PDO Mapping	Mapping parameter for the Transmit PDO	102

5.3.4.2.1 Object 0x1000, Device Type

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

The object indicates the implemented device profile. The CANopen bus coupler has implemented the „Device Profile for Generic I/O Modules" (device profile No. 401). Moreover, in the index 0x1000 the value informs about the type of modules connected.

Design:

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

Mit Bit 1 = 1, if at least one digital input is connected.
2 = 1, if at least one digital output is connected.
3 = 1, if at least one analog input is connected.
4 = 1, if at least one analog output is connected.

5.3.4.2.2 Object 0x1001, Error Register

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

This register contains internal errors. This register is also part of the emergency message.

Design:

Bit	Meaning
0	General Error
1	Current
2	Voltage
3	Temperature
4	Communication
5	Device profile specific
6	Reserved
7	Manufacturer specific

In the event of an error, bit 0 is always set. Additional bits used specify the error in more detail.

5.3.4.2.3 Object 0x1003, Pre-defined 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	-

The sub-index 0 contains the errors currently stored in the field. If a new error occurs, it will be entered in sub-index 1, and all errors already existing moved down by one sub-index. A max. of 20 error entries are supported. Should more than 20 errors occur, each time the error contained in sub-index 20 is written over.

Design Standard Error Field:

Bit31	Bit16	Bit15	Bit0
Additional Information		Error code	

The additional information corresponds to the first 2 bytes of the additional code of the Emergency telegram. The error code coincides with the error code in the Emergency telegram.

The complete error memory is deleted by writing a „0" in sub-index 0.

5.3.4.2.4 Object 0x1005, COB-ID SYNC message

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

The object defines the COB ID for the synchronization message.

Design:

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

5.3.4.2.5 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 max. time in µs for two subsequent SYNC messages. The internal resolution is 2ms. If the value is 0, no SYNC monitoring is performed.

5.3.4.2.6 Object 0x1008, Manufacturer Device Name

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1008	0	Manufacturer Device Name	Visible String	RO	i. e. 750-347

The object indicates the device name of the bus coupler.

5.3.4.2.7 Object 0x1009, Manufacturer Hardware Version

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1009	0	Manufacturer Hardware Version	Visible String	RO	Current HW-Version

The object indicates the current hardware version of the bus coupler.

5.3.4.2.8 Object 0x100A, Manufacturer Software Version

Idx	S-Idx	Name	Type	Attribute	Default Value
0x100A	0	Manufacturer Software Version	Visible String	RO	Current SW-Version

The object indicates the current software version of the bus coupler.

5.3.4.2.9 Object 0x100C, Guard Time

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

The object indicates the *Guarding Time* in milli-seconds. An NMT master cyclically interrogates the NMT slave for its status. The time between two interrogations is termed *Guard Time*.

5.3.4.2.10 Object 0x100D, Life Time Factor

Idx	S-Idx	Name	Type	Attribute	Default Value
0x100D	0	Lifetime Factor	Unsigned8	RW	0

The life *Time Factor* is part of the *Node Guarding Protocol*. The NMT slave checks if it was interrogated within the *Node Life Time* (Guardtime multiplied with the life time factor). If not, the slave works on the basis that the NMT master is no longer in its normal operation. It then triggers a *Life Guarding Event*.

If the node life time is zero, no monitoring will take place.

5.3.4.2.11 Object 0x1010, Store Parameters

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1010	0	Max. supported Sub-Index	Unsigned8	RO	1
	1	Store all Parameter	Unsigned32	RW	1

This object allows to permanently store the settings made by the user. For this purpose, the signature „save" (lower case letters ASCII - MSB – 0x65 76 61 73 - LSB) must be written into the index 0x1010 sub index 1. The storing process runs in the background and takes approx. 2-3 seconds. When the storing process is finished, the SDO reply telegram is sent. Communication remains possible during storage by means of SDOs. An error message as a result of a new storage attempt only occurs, when the previous one was not yet finished. It is also not possible to trigger the storage function for as long as „Restore" is active.

As soon as a setting is stored, the Emergency “Changed HW configuration” is not sent any longer if the bus coupler is started up again without changing the module configuration.

Attention
If following the storage of a configuration only the module ID is changed via the DIP switch, the saved configuration is continued to be used. In other words, all module ID specific entries in the object directory (objects that are module ID dependent and have the „rw" attribute) signal with the old values. (i.e. Emergency ID,...)

5.3.4.2.12 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 on default value	Unsigned32	RW	1
	2	-	Unsigned32	RW	0
	3	-	Unsigned32	RW	0
	4	Set all parameters on default values once	Unsigned32	RW	1

This object allows to reset the user stored parameters to the original default values.

Sub-indexes 2 and 3 are not supported.

The load command is processed in the background and takes approx. 2-3 seconds. When the performance is finished, the SDO reply message is sent. Communication can be continued during performance using SDOs. An error message is only tripped with another attempt to send a load command, if the previous one is not yet completed. It is also not possible to trigger a load command for as long as „Save" is active.

5.3.4.2.12.1 Sub-Index 1 - Permanent entry of default parameters

Writing the signature „load" (lower case letters ASCII - MSB 0x64 0x61 0x6F 0x6C LSB) into the index 0x1011 sub-index 1 entails loading of the standard factory settings after the following Power ON and each further Power On (until the next SAVE command is given).

5.3.4.2.12.2 Sub-index 4 – On-off entry of default parameters

Writing the signature „load" (lower case letters ASCII - MSB 0x64 0x61 0x6F 0x6C LSB) into the index 0x1011 sub-index 4 entails loading of the standard factory settings once only after the following Power ON. The saved configuration is re-loaded after each further Power ON. This feature can, for instance, be used during the development phase in order to quickly obtain a behavior comparison of saved and default configurations without having to re-set and re-store all parameters each time.

Sequence:

replace configuration once by default configuration

-> load (index 0x1011, sub-index 4)

-> Reset

-> default values
(re-use of the load (index 0x1011, sub-index 4) command is not permitted in this status!)

-> Reset

-> stored configuration

5.3.4.2.13 Object 0x1014, COB-ID Emergency Object

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

The object defines the COB ID for the EMCY message.

Design:

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

If a new COB ID is to be entered, set bit 31 to 1 first, because standard DS301 does not allow to change a valid COB ID (Bit31=0).

5.3.4.2.14 Object 0x1015, Inhibit Time Emergency Object

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

Diese Object gibt die min. Zeit an, die vergehen muss, bevor ein weiteres This object indicates the time in minutes which must be allowed to elapse prior to another Emergency to be sent.

An entry of zero deactivates the delayed transmission.

Due to the fact that with delayed transmission the entries are entered in a queue, the max. number of Emergencies in quick succession is limited to the queue size (20 entries). If this number is exceeded, an Emergency is sent immediately indicating the overflow.

One time unit is 100µs.

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

5.3.4.2.15 Object 0x1016, Consumer Heartbeat Time

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1016	0	Max. monitorable Modules	Unsigned8	RO	5
	1	1. Heartbeat Time Entry	Unsigned32	RW	0
	2	2. Heartbeat Time Entry	Unsigned32	RW	0
	3	3 .Heartbeat Time Entry	Unsigned32	RW	0
	4	4. Heartbeat Time Entry	Unsigned32	RW	0
	5	5. Heartbeat Time Entry	Unsigned32	RW	0

This entry allows the monitoring of a maximum of 5 modules. The system checks whether each module defined in this object has created a *Heartbeat* within the set time. If the set time was exceeded, a *Heartbeat-Event* is triggered. The *Heartbeat-Time* is entered in milli-seconds. The monitoring is deactivated, if the time value is 0.

Design:

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

5.3.4.2.16 Object 0x1017, Producer Heartbeat Time

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1017	0	Producer Heartbeat Time	Unsigned16	RW	0

The object defines the time between two Heartbeat messages sent in milli-seconds. If the time is 0, no Heartbeat is sent. The Heartbeat transmission starts as soon as a value other than 0 is entered.

5.3.4.2.17 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	i. e. 347
	3	Revision Number	Unsigned32	RO	akt. Rev.-No.
	4	Serial Number	Unsigned32	RO	akt. Serial-No.

The object specifies the device used.

The manufacturer ID has an unambiguous number assigned to each manufacturer. WAGO was assigned ID 33.
The device description reflects the product family.
The Rev. No. contains a specific CANopen behavior, the *Major-Rev.-No.* contains the CANopen functionality. If the functionality is changed, the *Major-Rev.-No.* is increased. Various versions of the same CANopen behavior can be differentiated by the *Minor-Rev.-No.*

Design Rev. No.:

Bit31	Bit16	Bit15	Bit0
Major-Rev.-Nr.		Minor-Rev.-Nr.	

The serial number is unambiguous for this device family.

5.3.4.2.18 Object 0x1029, Error Behaviour

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1029	0	Max. supported Sub-Index	Unsigned8	RO	1
	1	communication error	Unsigned8	RW	0

This object specifies, into which state the module in the case of a communication error (e.g. loss Node Guarding) changes.

Design communication error -Entry:

communication error	Action
0	Change into status *PRE-OPERATIONAL* (only if the current status were *OPERATIONAL*)
1	No status change
2	Change into status *Stopped*

5.3.4.2.19 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

Access to the entries in the object directory is made via this object.

The default value of the second SDO is not active. Any change to the COB IDs is prohibited in the second SDO, if these are active (Bit 31 = 0).

Design COB ID:

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

5.3.4.2.20 Object 0x1400– 0x1404, Receive PDO Communication Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1400 to 0x1404	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 0x80000000
	2	Transmission type	Unsigned8	RW	255

This object is used to set the communication parameters of the RxPDOs. 5 RxPDOs are supported. The default COB IDs of the first four PDOs are pre-assigned according to the DS301 standard. All further PDOs are deactivated. If not all default PDOs are used (i.e. a smaller number of modules is connected), also the default PDOs not used are deactivated.

Design COB-ID:

Bit 31	Bit 30	Bit 29 Bit 11	Bit 10 Bit 0
0/1 valid/invalid	0/1 RTR allowed / not allowed	reserved (always 0)	COB-ID

If a new COB ID is to be entered, bit 31 must be set to 1 first, because the DS301 standard does not permit to change a valid COB ID (Bit31=0).

A mode can be defined for each PDO for the purpose of data transmission (transmission type in the Index Communication Parameter). As standard, digital and analog inputs are transmitted as ´Change of Value´(COV). The type of transmission depending of the set transmission type is explained in the following table.

Trans-mission type	**PDO transmission**						
	cyclic	acyclic	synchro-nous	asynchro-nous	RTR only	TxPDO (inputs)	RxPDO (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 x SYNC (x = 1 to 240)	Set outputs **after each SYNC** as requested by the last PDO received
241 - 251	- reserved -						
252			X		X	Data is read-in 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 set inhibit time

5.3.4.2.21 Object 0x1600– 0x1604, Receive PDO Mapping Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1600 to 0x1604	0	Number of mapped Objects	Unsigned8	RW	-
	1 to 8	1.Object to 8.Object	Unsigned32	RW	-

This object is used to define the data, which is to be transmitted by means of the PDO.

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

Design 1. to 8. Object:

Bit31 Bit16	Bit 15 Bit8	Bit7 Bit0
Index	Sub-Index	Object size

Index: Index of the object to be transmitted

Sub-Index: Sub-index of the object to be transmitted

Object size: Object size in bits
Due to the fact that max. 8 bytes can be transmitted in a PDO, the sum of the valid object lengths must not exceed 64 (8Byte*8Bit)

5.3.4.2.22 Object 0x1800– 0x1804, Transmit PDO Communication Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1800 to 0x1804	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 0x80000000
	2	Transmission type	Unsigned8	RW	255
	3	Inhibit Time	Unsigned16	RW	Idx 0x1800 0 Idx 0x1801 – 1804 100
	4	Reserved	Unsigned8	RW	0
	5	Event Timer	Unsigned16	RW	0

This object is used to set the communication parameters of the TxPDOs. 32 TxPDOs are supported. The default COB IDs of the first four PDOs are pre-assigned according to the DS301 standard. All other PDOs are de-activated. If not all default PDOs are used (i.e. a smaller number of modules is connected), also the default PDOs not used are de-activated.

Design COB-ID:

Bit 31	Bit 30	Bit 29 Bit 11	Bit 10 Bit 0
0/1 valid/invalid	0/1 RTR allowed / not allowed	reserved (always 0)	COB-ID

If a new value is to be entered, the COB ID has to be set invalid (Bit 31 = 1), because the DS301 standard does not permit to enter a new time when the COB ID (Bit31=0) is valid.

A mode can be defined for each PDO for the purpose of data transmission (transmission type in the Index Communication Parameter). As standard, digital and analog inputs are transmitted as ´Change of Value´(COV). The type of transmission depending of the set transmission type is explained in the following table.

Trans-mission type	PDO transmission						
	cyclic	acyclic	synchro-nous	asynchro-nous	RTR only	TxPDO (inputs)	RxPDO (outputs)
0		X	X			if COV is transmitted with each SYNC	Set outputs after each SYNC as requested by the last PDO received

Trans-mission type	**PDO transmission**						
1 – 240	X		X			Transmission with each x SYNC (x = 1 to 240)	Set outputs **after each SYNC** as requested by the last PDO received
241 - 251	- reserved -						
252			X		X	Data is read-in 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 set inhibit time

Inhibit Time shows the min. time between two consecutive PDOs having the same COB ID. One time unit is 100us. The transmitted value is internally rounded to the next smaller milli-second.

Example: Min. time interval between two PDOs having the same COB ID: 30ms.
Sub-index 3 = 300 = 0x12C

The Event Timer defines the time after the elapse of which a PDO is sent, even if no change of the PDO data has occurred. Enter the time in milli-seconds. The timer is re-started whenever an event occurs (change to the PDO data).

If the time is shorter than the inhibit time, a new event is generated once the inhibit time has elapsed! The event timer can only be used for the transmission types 254/255.

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

5.3.4.2.23 Object 0x1A00 – 0x1A04, Transmit PDO Mapping Parameter

Idx	S-Idx	Name	Type	Attribute	Default Value
0x1A00 to 0x1A04F	0	Number of mapped objects	Unsigned8	RW	-
	1 to 8	1.Object to 8.Object	Unsigned32	RW	-

This object is used to define the data, which is transmitted using the PDO.

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

Design 1. to 8. Object:

Bit 31 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0
Index	Sub-Index	Object size

Index: Index of the object to be transmitted

Sub-Index: Sub-index of the object to be transmitted

Object size: Object size in bits
Due to the fact that max. 8 bytes can be transmitted in a PDO, the sum of the valid object lengths must not exceed 64 (8Byte*8Bit)

5.3.4.3 Manufacturer Specific Profile Area

The non-standard device profile-specific I/O functionality of special modules and other modules as well as special functions (i.e. empty module configuration,...) are imaged in the 'Manufacturer Specific Area' profile. The objects defined there provide data word widths of a modularity from 1 to 8 bytes.

The indexes 0x2000 (digital inputs), 0x2100 (digital outputs), 0x2400 (2 byte special module inputs) and 0x2500 (2 byte special module outputs) are mirror imaged by the corresponding indexes of the device profile DS 401 (0x6000, 0x6200, 0x6401, 0x6411). This means for instance: object 0x2000 and object 0x6000 refer to the same memory places in the process image.

The following table shows all objects of the manufacturer profile supported by the bus coupler.

Idx	Name	Type	Meaning	See on page
0x2000	Digital inputs	Array Unsigned8	Data of digital input modules	104
0x2100	Digital outputs	Array Unsigned8	Data of digital output modules	104
0x2200	1 byte special modules, inputs	Array Unsigned8	Data of 1 byte special input modules	104
0x2300	1 byte special modules, outputs	Array Unsigned8	Data of 1 byte special output modules	104
0x2400	2 byte special modules, inputs	Array Unsigned16	Data of 2 byte special input modules	104
0x2500	2 byte special modules, outputs	Array Unsigned16	Data of 2 byte special output modules	104
0x2600	3 byte special modules, inputs	Record	Data of 3 byte special input modules	105
0x2700	3 byte special modules, outputs	Record	Data of 3 byte special output modules	105
0x2800	4 byte special modules, inputs	Record	Data of 4 byte special input modules	105
0x2900	4 byte special modules, outputs	Record	Data of 4 byte special output modules	105
0x3000	5 byte special modules, inputs	Record	Data of 5 byte special input modules	105
0x3100	5 byte special modules, outputs	Record	Data of 5 byte special output modules	105
0x3200	6 byte special modules, inputs	Record	Data of 6 byte special input modules	106
0x3300	6 byte special modules, outputs	Record	Data of 6 byte special output modules	106
0x3400	7 byte special modules, inputs	Record	Data of 7 byte special input modules	106
0x3500	7 byte special modules, outputs	Record	Data of 7 byte special output modules	106
0x3600	8 byte special modules, inputs	Record	Data of 8 byte special input modules	106
0x3700	8 byte special modules, outputs	Record	Data of 8 byte special output modules	107
0x4500	Empty module configuration	Record	Configuration of virtual modules	107
0x5000	Input PA	Record	Reading of the input process image	111
0x5001	Output PA	Record	Writing the output process image	111

5.3.4.3.1 Object 0x2000, Digital Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2000	0	8-Bit digital input block	Unsigned8	RO	-	Number of digital 8 bit input blocks
	1	1. input block	Unsigned8	RO	-	1. digital input block
	....	....	....	....	....	....
	32	32. input block	Unsigned8	RO	-	32. digital input block

5.3.4.3.2 Object 0x2100, Digital Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2100	0	8-Bit digital output block	Unsigned8	RO	-	Number of digital 8 bit output blocks
	1	1. output block	Unsigned8	RW	0	1. digital output block
	....	....	....	....	....	....
	32	32. output block	Unsigned8	RW	0	32. digital output block

5.3.4.3.3 Object 0x2200, 1 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2200	0	Special 1 byte input	Unsigned8	RO	-	Number of 1 Byte special channels
	1	1. special input	Unsigned8	RO	-	1. Input channel
	....	....	....	....	....	....
	32	32. special input	Unsigned8	RO	-	32. Input channel

5.3.4.3.4 Object 0x2300, 1 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2300	0	special 1 byte output	Unsigned8	RO	-	Number of 1 Byte special channels
	1	1. special output	Unsigned8	RW	0	1. Output channel
	....	....	....	....	....	....
	32	32. special output	Unsigned8	RW	0	32. Output channel

5.3.4.3.5 Object 0x2400, 2 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2400	0	special 2 byte input	Unsigned8	RO	-	Anzahl der 2 Byte Sonderkanäle
	1	1. special input	Unsigned16	RO	-	1. Input channel
	....	....	....	....	....	....
	16	16. special input	Unsigned16	RO	-	16. Input channel

5.3.4.3.6 Object 0x2500, 2 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2500	0	special 2 byte output	Unsigned8	RO	-	Number of the 2 byte special channels
	1	1. special output	Unsigned16	RW	0	1. Output channel
	....	....	....	....	....	....
	16	16. special output	Unsigned16	RW	0	16. Output channel

5.3.4.3.7 Object 0x2600, 3 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2600	0	special 3 byte input	Unsigned8	RO	-	Number of the 3 byte special channels
	1	1.-st special input	Unsigned24	RO	-	1. Input channel
	....	....	....	....	....	....
	10	10. special input	Unsigned24	RO	-	10. Input channel

5.3.4.3.8 Object 0x2700, 3 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2700	0	special 3 byte output	Unsigned8	RO	-	Number of the 3 byte special channels
	1	1. special output	Unsigned24	RW	0	1. Output channel
	....	....	....	....	....	....
	10	10. special output	Unsigned24	RW	0	10. Output channel

5.3.4.3.9 Object 0x2800, 4 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2800	0	special 4 byte input	Unsigned8	RO	-	Number of the 4 byte special channels
	1	1.special input	Unsigned32	RO	-	1. Input channel
	....	....	....	....	....	....
	8	8. special input	Unsigned32	RO	-	8. Input channel

5.3.4.3.10 Object 0x2900, 4 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x2900	0	special 4 byte output	Unsigned8	RO	-	Number of the 4 byte special channels
	1	1. special output	Unsigned32	RW	0	1. Output channel
	....	....	....	....	....	....
	8	8. special output	Unsigned32	RW	0	8. Output channel

5.3.4.3.11 Object 0x3000, 5 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3000	0	special 5 byte input	Unsigned8	RO	-	Number of the 5 byte special channels
	1	1. special input	Unsigned40	RO	-	1. Input channel
	....	....	....	....	....	....
	6	6. special input	Unsigned40	RO	-	6. Input channel

5.3.4.3.12 Object 0x3100, 5 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3100	0	special 5 byte output	Unsigned8	RO	-	Number of the 5 byte special channels
	1	1. special output	Unsigned40	RW	0	1. Output channel
	....	....	....	....	....	....
	6	6. special output	Unsigned40	RW	0	6. Output channel

5.3.4.3.13 Object 0x3200, 6 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3200	0	special 6 byte input	Unsigned8	RO	-	Number of the 6 byte special channels
	1	1. special input	Unsigned48	RO	-	1. Input channel
	....	....	....	....	....	....
	5	5. special input	Unsigned48	RO	-	5. Input channel

5.3.4.3.14 Object 0x3300, 6 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3300	0	Special 6 byte output	Unsigned8	RO	-	Number of the 6 byte special channels
	1	1. special output	Unsigned48	RW	0	1. Output channel
	....	....	....	....	....	....
	5	5. special output	Unsigned48	RW	0	5. Output channel

5.3.4.3.15 Object 0x3400, 7 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3400	0	Special 7 byte input	Unsigned8	RO	-	Number of the 7 byte special channels
	1	1. special input	Unsigned56	RO	-	1. Input channel
	....	....	....	....	....	....
	4	4. special input	Unsigned56	RO	-	4. Input channel

5.3.4.3.16 Object 0x3500, 7 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3500	0	Special 7 byte output	Unsigned8	RO	-	7 Byte Special Modules, Outputs
	1	1. special output	Unsigned56	RW	0	1. Output channel
	....	....	....	....	....	....
	4	4. special output	Unsigned56	RW	0	4. Output channel

5.3.4.3.17 Object 0x3600, 8 Byte Special Modules, Inputs

Idx	S-Idx	Name	Type	Attribute	Default Wert	Bedeutung
0x3600	0	special 8 byte input	Unsigned8	RO	-	Number of the 8 byte special channels
	1	1. special input	Unsigned64	RO	-	1. Input channel
	....	....	....	....	....	....
	4	4. special input	Unsigned64	RO	-	4. Input channel

5.3.4.3.18 Object 0x3700, 8 Byte Special Modules, Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x3700	0	Special 8 byte output	Unsigned8	RO	-	Number of the 8 byte special channels
	1	1. special output	Unsigned64	RW	0	1. Output channel
	....	....	....	....	....	....
	4	4. special output	Unsigned64	RW	0	4. Output channel

5.3.4.3.19 Object 0x4500, Spacer Module Configuration

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x4500	0	Number of plugged and virtual I/O nodules	Unsigned8	RW	0	0 nicht aktiv 1 .. 64 Anzahl der max. gesteckten Klemmen (physikalisch + virtuell)
	1	1. I/O module description	Unsigned16	RW	0	1. I/O module
	....	....	....	....	....	....
	64	64. I/O module description	Unsigned16	RW	0	64. I/O module

With the help of this object it is possible to insert virtual modules into the node. In this manner, for instance, a max. node extension can be projected and a new node subsequently designed which represents a sub-quantity in relation to the maximum configuration. The behavior of this new node with regard to its object entries in relation to connected modules is identical to that of the maximum configuration. As such, other applications (CANopen master,...) can be designed once for the maximum configuration and have access to any sub-quantity without having to change its setting.
In the same manner it is possible to provide for future extensions from the start of the design process and making later adaptation of the mapping will be unnecessary.

- Sub-Index 0
  Sub-Index 0 = 0: Mapping of virtual modules not active
  Sub-Index 0 ≠ 0 Mapping in of virtual modules active

The entry indicates the number of the connected modules in the maximum configuration.

When changing the value from 0 to >0, the configuration is created as described from sub-index 1. During the creation of a new configuration, all previously configuration settings that have not been permanently stored, will be written over and the process image reset. For this reason, always configure index 0x4500 first, followed by all other settings (mapping, sync time,...).

Setting the sub-index 0 is only possible in the *Pre-Operational* status.

If the creation of a new configuration is free from errors, an Emergency is sent with the parameters PP=LL=SS=0.

If an error occurs during the creation of a new configuration (i.e. the number of connected modules exceeds that of the modules configured), a corresponding Emergency is transmitted. The coupler starts with the default configuration in accordance with the modules connected, and changes to the STOP status.

- Emergency Message:
  Error Code 0x5000
  Error Register 0x81

- Additional Code 00 03 **PP LL SS**
  PP: indicates the physical module slot, where the error has occurred
  LL: indicates the logic slot (slot in the maximum configuration) of the

module, where the error has occurred
SS: Cause of the error

Emergency structure under a faulty creation of a configuration

Emergency-Aufbau bei fehlerhafter Konfigurationserstellung

Parameter: SS (cause of the error)		
Bit 4..7	**Bit 0..3**	**Description**
0	1	analog module expected acc. to the configuration
0	2	digital module expected acc. to the configuration
0	3	output module expected acc. to the configuration
0	4	input module expected acc. to the configuration
0 *N*	5	digital module: - wrong number of bits for the module (bits per channel * number of channels). analog module: - module with *n* channels acc. to the configuration
0	6	more modules are put, as configurated
0	7	digital modules: - invalid analog modules: - wrong number of bytes per channel

Note
Should module diagnostic messages occur by means of an Emergency message, the display of the module position always refers to the logic module position in the node. Consequently these messages are always identical, irrespective of the node configuration.

- Sub-Index 1..64
  Sub-index 1..64 contains the configuration of the node in its maximum configuration. Each index stands for a connected module (sub-index.1 1st module, sub-index.2 2nd module,...). These indexes describe the corresponding module in detail

Design of 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			Out-put	Input	A/D

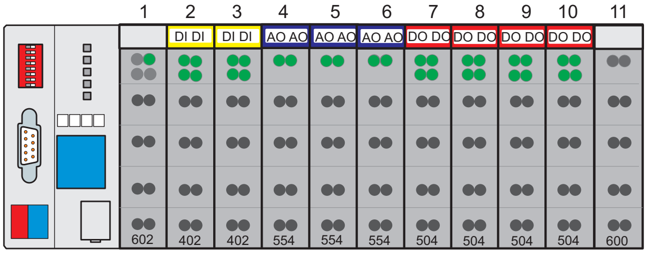
A/D:	indicates whether or not the module is of an analog or a digital type 0=analog, 1=digital
Input:	indicates whether or not the module is an input module 0=no input module, 1= input module *1)
Output:	indicates whether or not the module is an output module 0=no output module, 1= output module *1)
Channels:	indicates the number of module channels
Bits/Bytes:	specifies the number of bytes (analog module) or bits (digital module) per channel which are mapped into the process image

Reserved:	reserved
Kl:	indicates whether or not the module is connected 0=module not connected, 1=module connected

*1) The input and the output bit can be set simultaneously (i.e. digital output module with diagnostics, this module has input and output bits)

Example:

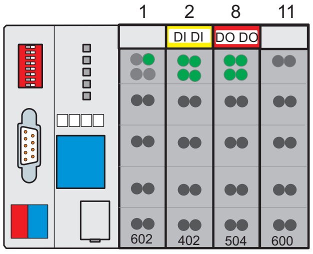
A coupler of the following configuration is projected:



Configuration for the full version:

Sub-Index.	Value	Meaning
0	0x09	Total number of connected (9) and virtual (0) modules
1	0x8063	digital, input, 4 channels, 1bit per channel, connected (mod. 1)
2	0x8063	digital, input, 4 channels, 1bit per channel, connected (mod. 2)
3	0x 8094	analog, output, 2 channels, 2byte per channel, connected (mod. 3)
4	0x 8094	analog, output, 2 channels, 2byte per channel, connected (mod. 4)
5	0x 8094	analog, output, 2 channels, 2byte per channel, connected (mod. 5)
6	0x 8055	digital, output, 2 channels, 1bit per channel, connected (mod. 6)
7	0x 8055	digital, output, 2 channels, 1bit per channel, connected (mod. 7)
8	0x 8055	digital, output, 2 channels, 1bit per channel, connected (mod. 8)
9	0x 8055	digital, output, 2 channels, 1bit per channel, connected (mod. 9)

As a second step, a coupler is protected which is a sub-quantity of the first one. The only modules used are module 2 and 8



Configuration for the version mod. 2 and mod. 8 connected:

S-Idx.	Value	Meaning
0	0x09	Total number of connected (2) and virtual (7) modules
1	0x0063	digital, input, 4 channels, 1bit per channel, not conn.(mod. 1)
2	0x8063	digital, input, 4 channels, 1bit per channel, connected (mod. 2)
3	0x 0094	analog, output, 2 channels, 2byte per channel, not conn.(mod. 3)
4	0x 0094	analog, output, 2 channels, 2byte per channel, not conn.(mod. 4)
5	0x 0094	analog, output, 2 channels, 2byte per channel, not conn.(mod. 5)

6	0x 0055	digital, output, 2 channels, 1bit per channel, not conn.(mod. 6)
7	0x 0055	digital, output, 2 channels, 1bit per channel, not conn.(mod. 7)
8	0x 8055	digital, output, 2 channels, 1bit per channel, connected (mod. 8)
9	0x 0055	digital, output, 2 channels, 1bit per channel, not conn.(mod. 9)

With regard to their entries in the object directory (connected modules), the behavior of both couplers is identical. Consequently, their behavior is also identical during PDO mapping..

Bit 4 must be set in index 0x2100, sub-index 1, for instance, to be able to set the 1st channel of the output module 8 (750-513).

This process is identical for both configurations.

If there was not an empty module configuration, in contrast to the above, bit 0 would have to be set for the 2nd configuration, index 0x2100, sub-index 1, in order to set the same output channel.

5.3.4.3.20 Object 0x5000, Read Input Process Image

Idx	S-Idx	Name	Type	Attribute	Default Wert	Bedeutung
0x5000	0	number of input byte	Unsigned16	RO	-	Number of relevant bytes in Input PA
	1	input segment	Octed_String	RO	-	Input PA segment

Permits reading of the entire input process image as a domain via SDO, allowing access to all input data „as a block“.

Note
The access via SDO being slow, we recommend transmitting time critical data only via PDO.

5.3.4.3.21 Object 0x5001, Write Output Process Image

Idx	S-Idx	Name	Type	Attribute	Default Value	Meaning
0x5001	0	number of output byte	Unsigned16	RO	-	Number of relevant bytes in Output PA
	1	output segment 1	Octed_String	RW	-	Output PA segment

Permits writing of the entire output process image as a domain via SDO „as a block“.

Note
The access via SDO being slow, we recommend transmitting time critical data only via PDO.

5.3.4.4 Standard Device Profile Area – DS 401

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

The following table shows all objects of the standard profile DS401 supported by the bus coupler.

Idx	Name	Type	Meaning	See page
0x6000	Read Input 8 Bit	Array Unsigned8	Data of digital input I/O modules	113
0x6005	Global Interrupt Enable Digital 8-Bit	Unsigned8	Global release of the transmission of 8 bit digital input data	113
0x6006	Interrupt Mask Any Change 8-Bit	Array Unsigned8	Release of the transmission with each change of 8 bit digital input data	113
0x6007	Interrupt Mask Low-to-High 8-Bit	Array Unsigned8	Release of the transmission if a positive flank of 8 bit digital input data occurs	113
0x6008	Interrupt Mask High-to-Low 8-Bit	Array Unsigned8	Release of the transmission of a negative flank of 8 bit digital input data	114
0x6200	Write Output 8-Bit	Array Unsigned8	Data of digital output I/O modules	114
0x6206	Error Mode Output 8-Bit	Array Unsigned8	Release of pre-defined error values of the 8 bit digital output data	115
0x6207	Error Value Output 8-Bit	Array Unsigned8	Pre-defined error values of the 8 bit digital output data	115
0x6401	Read Analogue Input 16-Bit	Array Unsigned16	Data of analog input I/O modules (16 bit)	115
0x6411	Write Analogue Output 16-Bit	Array Unsigned16	Data of analog output I/O modules (16 bit)	116
0x6421	Analogue Input Trigger Selection	Array Unsigned8	Determination of trigger condition for 16 bit analog input data	116
0x6423	Analogue Input Global Interrupt Enable	Boolean	Global release of the transmission of 16 bit analog input data	117
0x6424	Analogue Input Interrupt Upper Limit Integer	Array Unsigned16	Transmission of 16 bit input data if threshold value exceeded	117
0x6425	Analogue Input Interrupt Lower Limit Integer	Array Unsigned16	Transmission of 16 bit input data if threshold value exceeded	117
0x6426	Analogue Input Interrupt Delta Unsigned	Array Unsigned16	Transmission if the 16 bit input data have changed at least by the delta value	117
0x6427	Analogue Input Interrupt Negative Delta Unsigned	Array Unsigned16	Transmission if the 16 bit input data have reduced at least by the delta value	118
0x6428	Analogue Input Interrupt Positive Delta Unsigned	Array Unsigned16	Transmission if the 16 bit input data have increased at least by the delta value	118
0x6443	Analogue Output Error Mode	Array Unsigned8	Release of pre-defined error values of the 16 bit output data	118
0x6444	Analogue Output Error Value Integer	Array Unsigned16	Value in the event of an error of the 16 bit output data	119
0x67FE	Error Behaviour	Array Unsigned8	Status change in the event of an error	119

5.3.4.4.1 Object 0x6000, Digital Inputs

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6000	0	Number of digital input blocks	Unsigned8	RO	-
	1	1. input block	Unsigned8	RO	-
	2	2. input block	Unsigned8	RO	-
	....	....	....	....	....
	32	32. 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 from the left to the right, counted from starting with the bus coupler. Sub-index 2 the next etc.

5.3.4.4.2 Object 0x6005, Global Interrupt Enable Digital 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6005	0	Global Interrupt Enable Digital 8-Bit	Unsigned8	RW	1

With this object, the transmission of the digital input data is controlled using PDO. If the value is 1, the transmission is generally released, only depending on the objects 0x6006...0x6008 and the type of transmission of the PDO. If the value is 0, the digital input data is not transmitted, independent of the objects 0x6006...0x6008.

5.3.4.4.3 Object 0x6006, Interrupt Mask Any Change 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6006	0	Number of digital input blocks	Unsigned8	RO	-
	1	Maske 1. input block	Unsigned8	RW	255
	2	Maske 2. input block	Unsigned8	RW	255
	....	....	....	....	....
	32	Maske 32. input block	Unsigned8	RW	255

This object is used to define the digital input channel, which will send its data in the event of a change. Prerequisite being that transmission is generally released (Object 0x6005 = 1).

0 = Transmission blocked in the event of a change (per channel)

1 = Transmission released in the event of a change (per channel)

Example: Sub-Index 0 = 1, Sub-Index 1 = 65 = 0x41
only channel 1 and 7 will transmit their data in the event of a change

5.3.4.4.4 Object 0x6007, Interrupt Mask Low-to-High 8-Bit

Idx	S-Idx	Name	Type	Attribut	Default Value
0x6007	0	Number of digital input blocks	Unsigned8	RO	-
	1	Mask 1. input block	Unsigned8	RW	0
	2	Mask 2. input block	Unsigned8	RW	0
	....	....	....	....	....
	32	Mask 32. input block	Unsigned8	RW	0

This object is used to define the digital input channel, which will send its data in the event of a positive flank (change from 0 to 1). Prerequisite being that the transmission is generally released (Object 0x6005 = 1). This object has an OR link to object 0x6006.

0 = Transmission blocked with a positive flank (per channel)

1 = Transmission release with a positive flank (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 in the event of a change Channel 6 is only transmitted with a positive flank

5.3.4.4.5 Object 0x6008, 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 1. input block	Unsigned8	RW	0
	2	Mask 2. input block	Unsigned8	RW	0
	....	....	....	....	....
	32	Mask 32. input block	Unsigned8	RW	0

This object is used to define the digital input channel, which transmits its data in the event of a negative flank (change from 1 to 0). Prerequisite being that the transmission is generally released (Object 0x6005 = 1). This object has an OR link to object 0x6006.

0 = Transmission blocked with a negative flank (per channel)

1 = Transmission released with a negative flank (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 in the event of a change Channel 6 is only transmitted with a negative flank

5.3.4.4.6 Object 0x6200, Digital Outputs

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6200	0	Number of digital output blocks	Unsigned8	RO	-
	1	1. output block	Unsigned8	RW	0
	2	2. output block	Unsigned8	RW	0
	....	....	....	....	....
	32	32. 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 from left to right, counting starting from the bus coupler. Sub-index 2 the next etc.

5.3.4.4.7 Object 0x6206, Error Mode Output 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6206	0	Number of digital output blocks	Unsigned8	RO	-
	1	Maske 1. output block	Unsigned8	RW	255
	2	Maske 2. output block	Unsigned8	RW	255
	....	....	....	....	....
	32	Maske 32. output block	Unsigned8	RW	255

This object defines whether the outputs change to a pre-defined error status in the event of an error (i.e. bus coupler changes to the *Stopped* status, Node-guarding has failed,...) (see object 0x6207). If the error is remedied, the outputs remain in their momentary status, i.e. the set error status of the output channels remains unchanged.

0 = Outputs remain unchanged (per channel)

1 = Outputs change to a pre-defined error status (per channel)

5.3.4.4.8 Object 0x6207, Error Value Output 8-Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6207	0	Number of digital output blocks	Unsigned8	RO	-
	1	Mask 1. output block	Unsigned8	RW	0
	2	Mask 2. output block	Unsigned8	RW	0
	....	....	....	....	....
	32	Mask 32. output block	Unsigned8	RW	0

This object is used to define the values, which the outputs should assume in the event of an error. Prerequisite being that the corresponding bit in object 0x6206 is set.

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 in the event of an error

5.3.4.4.9 Object 0x6401, Analog Inputs 16 Bit

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

This object contains the process data of the analog input modules. Sub-index 1 contains the first analog input channel from left to right, counting starting with the bus coupler. Sub-index 2 the second, etc.

5.3.4.4.10 Object 0x6411, Analog Outputs 16 Bit

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6411	0	Number analog input channels (16Bit)	Unsigned8	RO	-
	1	1. channel	Unsigned16	RW	0
	....	....	....	....	....
	16	16. channel	Unsigned16	RW	0

This object contains the process data of the analog output modules. Sub-index 1 contains the first analog output channel from left to right, counting starting with the bus coupler. Sub-index 2 the second, etc.

5.3.4.4.11 Object 0x6421, Analogue Input Interrupt Trigger Selection

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

This object is used to define the condition of the transmission. Prerequisite for the transmission being that a 1 is entered in object 0x6423, and consequently the general transmission released.

Design Sub-Index 1..16:

Bit	Transmission conditions	Configuration Sub-Index
0	Threshold value exceeded	0x6424
1	Threshold value fallen short	0x6425
2	Change of the input value exceeding the delta value for the last transmission	0x6426
3	Reduction of the input value by more than the delta value for the last transmission	0x6427
4	Increase of the input value by more than the delta value for the last transmission	0x6428
5 to 7	Reserved	-

5.3.4.4.12 Object 0x6423, Analogue Input Global Interrupt Enable

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6423	0	Global Interrupt Enable Analogue 16Bit	Unsigned8	RW	0

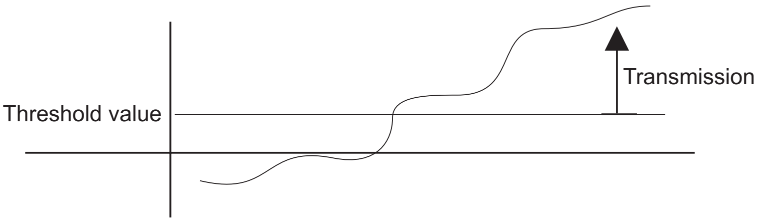
This object is used to control the transmission of the analog input data using PDO. If the value is 1, the transmission is released, and it only depends on object 0x6421 and the transmission type of the PDO. If the value is 0, no transmission of the analog input data is made, independent of object 0x6421.

Attention
In the default setting, the transmission of analog input data is deactivated.

5.3.4.4.13 Object 0x6424, Analogue Input Interrupt Upper Limit Integer

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6424	0	Number analog input channels (16Bit)	Unsigned8	RO	-
	1	Upper Limit 1. channel	Unsigned16	RW	0
	....	....	....	....	....
	16	Upper Limit 16. channel	Unsigned16	RW	0

This object allows a threshold value monitoring if it is configured in object 0x6423. If the input value is ≥ the defined threshold value, no transmission will take place until a further trigger condition is set (i.e. object 0x6426).

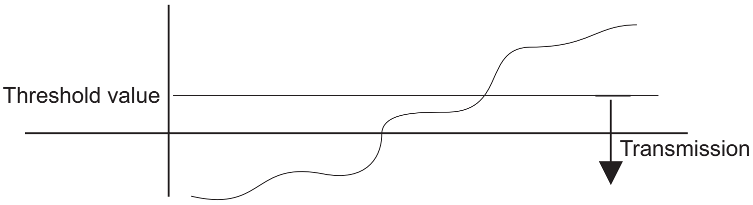


g012435d

5.3.4.4.14 Object 0x6425, Analogue Input Interrupt Lower Limit Integer

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6425	0	Anzahl Analoger Eingangskanäle (16Bit)	Unsigned8	RO	-
	1	Schwellwert 1. Kanal	Unsigned16	RW	0
	....	....	....	....	....
	16	Schwellwert 16. Kanal	Unsigned16	RW	0

This object allows a threshold value monitoring if it is configured in object 0x6423. If the input value is ≤ the determined threshold value, transmission will take place until a further trigger condition is set (i.e. object 0x6426).



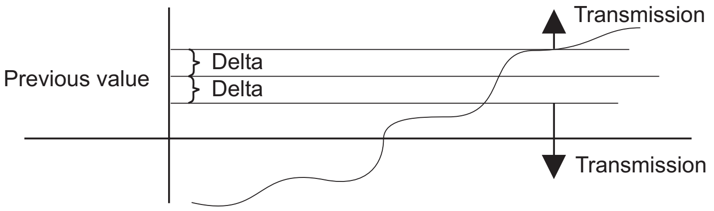
g012436d

5.3.4.4.15 Object 0x6426, Analogue Input Interrupt Delta Unsigned

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6426	0	Number analog input channels (16Bit)	Unsigned8	RO	-
	1	Deltawert 1. channel	Unsigned16	RW	0
	....	....	....	....	....
	16	Deltawert 16. channel	Unsigned16	RW	0

The new value to be transmitted must, by definition of this object, be larger by at least the delta value or smaller than the value sent before.

This object, for instance, can be linked with the object 0x6424, so that the transmission will only be completed when the set threshold value and also the delta function are fulfilled.

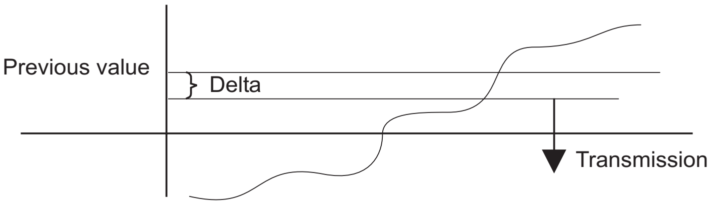


g012437e

5.3.4.4.16 Object 0x6427, Analogue Input Interrupt Negative Delta Unsigned

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6427	0	Number analog input channels (16Bit)	Unsigned8	RO	-
	1	Deltawert 1. channel	Unsigned16	RW	0
	....	....	....	....	....
	16	Deltawert 16. channel	Unsigned16	RW	0

By definition of this object, the new value to be transmitted, must be smaller than the previously sent value, at least by the delta value.

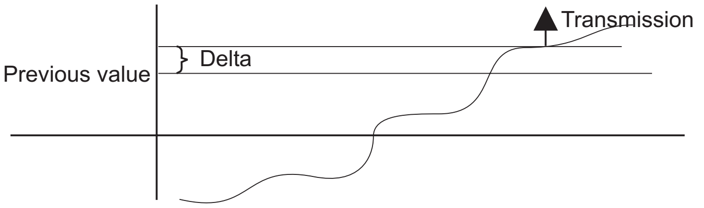


g012438e

5.3.4.4.17 Object 0x6428, Analogue Input Interrupt Positive Delta Unsigned

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6428	0	Number analog input channels (16Bit)	Unsigned8	RO	-
	1	Deltawert 1. channel	Unsigned16	RW	0
	....	....	....	....	....
	16	Deltawert 16. channel	Unsigned16	RW	0

By definition of this object, the new value to be transmitted, must be larger than the previously sent value, at least by the delta value.



g012439e

5.3.4.4.18 Object 0x6443, Analogue Output Error Mode

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6443	0	Number analog output channels (16Bit)	Unsigned8	RO	-
	1	Error Mode 1. channel	Unsigned8	RW	1
	....	....	....	....	....
	16	Error Mode 16. channel	Unsigned8	RW	1

This object is used to define whether the outputs change to a pre-defined error status (see object 0x6444) in the event of an error (i.e. bus coupler changes to the *Stopped* status, Nodeguarding has failed,...). Once the error is remedied, the outputs retain their momentary status, i, e. the set error status of the output channels remains unchanged.

All analog outputs that are not covered by the object 0x6444 (i.e. analog 6 byte modules) are always set to 0 in the event of an error.

0 = The output remains unchanged

1 = The output changes to a pre-defined error status

5.3.4.4.19 Object 0x6444, Analogue Output Error Value Integer

Idx	S-Idx	Name	Type	Attribute	Default Value
0x6444	0	Number analog output channels (16Bit)	Unsigned8	RO	-
	1	Error Mode 1. channel	Unsigned16	RW	0
	....	....	....	....	....
	16	Error Mode 16. channel	Unsigned16	RW	0

This object is used to define values that they are to assume in the event of an error. Prerequisite being that the corresponding bit is set in object 0x6443.

5.3.4.4.20 Object 0x67FE, Error Behaviour

Idx	S-Idx	Name	Type	Attribute	Default Value
0x67FE	0	Max. unterstützter Sub-Index	Unsigned8	RO	1
	1	Kommunikationsfehler	Unsigned8	RW	0

This object is an identical copy of the Error Behaviour Object 0x1029. Both objects have the same functionality. It is used to save the compatibility to control systems, which only support the Error Behaviour Object 0x67FE.

5.3.4.5 Reserved area

Note
This object directory area will remain unassigned.

5.3.5 PDO Transmission

Data transmission with PDOs is only possible in the *Operational* status.

When changing to the *Operational* status, all TxPDOs are transmitted once with the transmission type 254 and 255.

→

Note
Special transmission type 254 and 255
(Index 0x1800 ... 0x181F, sub-index 2):
The analog changes are not transmitted because of the default value (=FALSE) according to the device profile DS401-Object 0x6423 (analog Input Global Interrupt Enable). In this manner, a CAN bus overflow with CAN messages is prevented. To prevent an overflow when setting the object 0x6423 = TRUE, a correspondingly long *Inhibit Time* can be selected. Moreover, there is the possibility 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).

5.3.5.1 Mapping

By PDO mapping you can define the data to be transmitted by means of PDOs.

If no stored customer specific configuration is used and if no other settings are performed, the object directory is assigned with a default configuration according to the device profile DS 401 (refer to chapter 5.3.4.1 "Initialization").

If the coupler is in the PRE-OPERATIONAL status, its mapping can be modified via SDOs instead, in an application specific manner.

i

Further information
For an example of how to create an application specific mapping configuration, refer to chapter 3.1.6 "Starting up CANopen Fieldbus Nodes".

5.3.6 SYNC Monitoring

If the value of the communication cycle period is unequal to 0, monitoring is made with the first arrival of a SYNC message if the bus coupler is in the *Operational* status.

Failure of SYNC message:

If no SYNC message is received within the monitoring time (communication cycle period), this is signalled by a blink code. No status change occurs. In addition, an emergency message (Error Code: 0x8100, Error Register: 0x81, Additional Code: 00 04 00 00 00) is sent. The failure of the SYNC message will be displayed even if the MASTER provokes a status change.

It is only after the repeated receipt of the SYNC message in the *OPERATIONAL* status that the LEDs regain their normal operating status, and another emergency message (Error Code: 0x0000, Error Register: 0x81, Additional Code: 00 04 00 000 0) is sent to show that the SYNC monitoring functions again.

5.3.7 Node Guarding

The Node Guarding starts for the bus coupler when the first remote transmit request message (RTR) is received on the COB ID for the Node Guarding (0x700+Module-ID). If the bus coupler receives no corresponding message, the Node Guarding is not monitored by the bus coupler.

In the default setting, the Node Guarding is deactivated, because a 0 is entered in the corresponding indexes (0x100C = Guard-Time, 0x100D = Life Time Factor).

The NMT master polls the bus coupler at regular intervals. This interval is termed Guard-Time (Index 0x100C). The internal status of the bus coupler is in the reply message.

On the arrival of an RTR request without the Guard Time being set, the Node Guarding is not monitored, nevertheless the bus coupler replies with its internal status.

The status is coded as follows:

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

Life-Time is the product of Guard-Time (Index 0x100C) and Life Time Factor (Index 0x100D).

Failure of Node Guarding:

If no Node Guarding message is received with the life-time, this is shown by a blink code. In addition, an emergency message (Error Code: 0x8130, Error Register: 0x11, Additional Code: 0x00 04 00 000 0) is sent, the outputs are activated according to the objects 0x6206, 0x6207, 0x6443 and 0x6444, and the bus coupler changes to the pre-defined status according to object 0x67FE.

As soon as the Node Guarding protocol is recorded, another emergency message (Error Code: 0x0000, Error Register: 0x11, Additional Code: 00 04 00 000 0) is sent to show that the Node Guarding is reactivated, whereby the outputs and the bus coupler status remain unchanged.

It is possible to only use the Node Guarding protocol or the Heartbeat protocol. If the Heartbeat-Producer-Time is configured, the Heartbeat protocol is always used.

5.3.8 Heartbeat Monitoring

This protocol allows for monitoring the modules without having to use RTR frames.

The Heartbeat generator cyclically generates a message (time interval defined in object 0x1017), in which it transmits the module status. Transmission begins immediately after configuring the object 0x1017. The message can be evaluated by one or several Heartbeat consumers (object 0x1016). A maximum of 5 modules can be monitored. Monitoring starts with the first arrival of a Heartbeat message (separate for every module to be monitored).

Failure of the Heartbeat:

If no corresponding Heartbeat message is received within the configured time (object 0x1016), this is signalled by a blink code. In addition, an emergency message (Error Code: 0x8130, Error Register: 0x11, Additional Code: 0x00 05 KK 00 00, KK node number which has triggered EMCY) is sent. The outputs are activated according to objects 0x6206, 0x6207, 0x6443 and 0x6444 and the bus coupler changes to the status pre-defined according to 0x67FE.

As soon as the Heartbeat protocol is recorded, another emergency message (Error Code: 0x0000, Error Register: 0x11, Additional Code: 0x00 05 KK 00 00) is sent to display that Heartbeat is active again, whereby the outputs and the bus coupler status remain unchanged. If several modules are monitored, the blink code signalling the failure of the Heartbeat only stops after the previous Heartbeat has been resumed.

The only protocols to be used are the Node Guarding or the Heartbeat protocol. The Heartbeat protocol is used whenever the Heartbeat producer time is configured.

5.3.9 Error Message (Emergency)

Emergency messages are always sent in the event of a critical error situation having occurred/overcome in the device, or if important information has to be communicated to other devices.

Structure and meaning of the entries in the emergency object are explained in the table “EMCY-CODE”, they are coded in the bus message in a Lowbyte / Highbyte order.

An emergency object is also sent, after an error is remedied (Error Code = 0x0000, the Error Register and the Additional Code behave as described in the table “EMCY-CODE”).

Following Power On an emergency object is sent if the loaded settings are the default settings. This occurs for two reasons:

- No settings have yet been saved (Index 0x1010).
- The saved setting were discarded by the bus coupler, because modules were connected or disconnected.

EMCY-CODE

Byte:	**0 1**	**2**	**3 7**	
Name	**Error Code**	**Error Register**	**Additional Code**	**Meaning**
	0x0000*	0x00	00 00 00 00 00	The "predefined error field" Index 0x1003 SubIdx. 0 set to zero or all errors are cleared
	0x5000*	0x81	00 01 00 00 00	Changed hardware configuration after power on or reset Node / communication The fieldbus coupler is be initialized, because no stored configuration is available or the one available does not coincide with the current configuration
	0x5000*	0x81	00 02 00 00 00	Flash errors An error has occurred when saving the configuration in Flash.
	0x5000*	0x81	00 03 PP LL SS	The programmed configuration does not coincide with the actual one PP: physical module slot where the error has occurred LL: logic module slot where the error has occurred SS: Cause of the error
	0x5000*	0x81	00 09 00 00 00	Queue overflow for emergency messages (can only occur when the inhibit time for emergency is used)
	0x8100*	0x81	00 04 00 00 00	The time span between two SyncObjects is longer than the communication_Cycle_Period
	0x8110*	0x11	00 01 00 00 00	internal receive buffer overflow, status change as defined in object 0x67FE. The outputs are switched as defined in the Error-Mode/Value Objects
	0x8110*	0x11	00 02 00 00 00	internal transmit buffer overflow, status change as defined in object 0x67FE. The outputs are switched as defined in the Error-Mode/Value objects
	0x8120*	0x11	00 03 00 00 00	CAN Controller in Error Passive Mode
	0x8130*	0x11	00 04 00 00 00	The time between two node guarding telegrams is greater than Guard_Time * Life_Time_Faktor.
	0x8130*	0x11	00 05 KK 00 00	The time between two Heartbeat telegrams is greater than configured KK: Node that has tripped the time overflow

Byte:	**0 1**	**2**	**3 7**	
Name	**Error Code**	**Error Register**	**Additional Code**	**Meaning**
	0x8210*	0x81	00 05 SS II NN	PDO was sent with a number of bytes smaller than that configured in the communication profile. The PDO data is discarded, i.e. the outputs remain unchanged SS:Set point value - configured value (i.e. in Index 0x1600 Sub-index 0) II:Actual value - number of bytes sent NN:Number of PDO (1..32)
	0x8220*	0x81	00 08 SS II NN	PDO was sent with a number of bytes larger than that configured in the communication profile. Only the first n data is used (n = total length configured in the object directory) SS:Set point value - configured value (total length of all valid and configured objects in bytes) II:Actual value - number of bytes sent NN:Number of PDO (1..32)
	0xFF00*	0x81	00 06 PP 00 00	Module bus error, change to the STOP status - PP: Module position
	0xFF00*	0x81	DD 07 PP SK NN	Diagnosis message - DD: Diagnosis byte - PP: Module position - SK:Error status and channel number - NN :Number of current module error

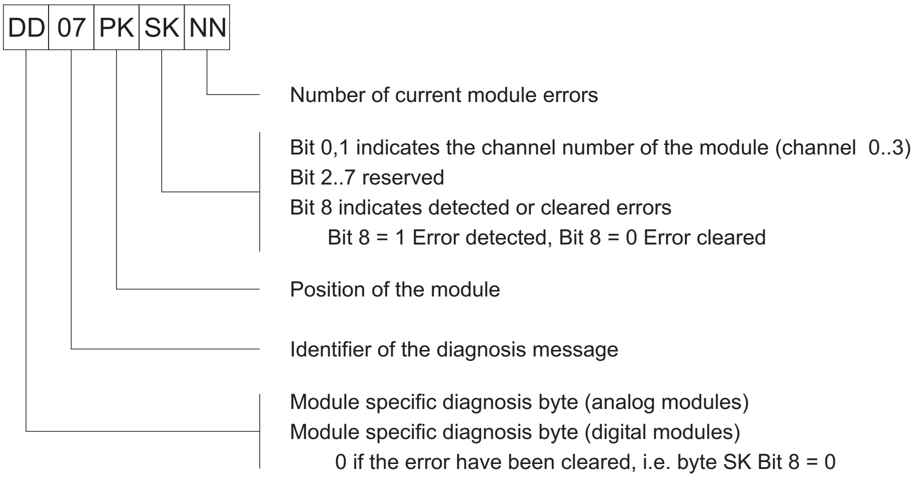
* Byte 0 = Lowbyte und Byte 1 = Highbyte

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

5.3.9.1 Diagnostic message of I/O modules

In the event of an error occurring in a module, which supports diagnostics, the diagnostic status is transmitted by means of the emergency message.

Design of the Additional Code of the Diagnosis message:



Number of current module errors

Bit 0,1 indicates the channel number of the module (channel 0..3)
Bit 2..7 reserved
Bit 8 indicates detected or cleared errors
Bit 8 = 1 Error detected, Bit 8 = 0 Error cleared

Position of the module

Identifier of the diagnosis message

Module specific diagnosis byte (analog modules)
Module specific diagnosis byte (digital modules)
0 if the error have been cleared, i.e. byte SK Bit 8 = 0

Fig. 5-19: Design of Additional Code

g012423x

Example:

- 2 channel analog input module 750-465, connected at position14, current on channel 0 has more than 20mA.

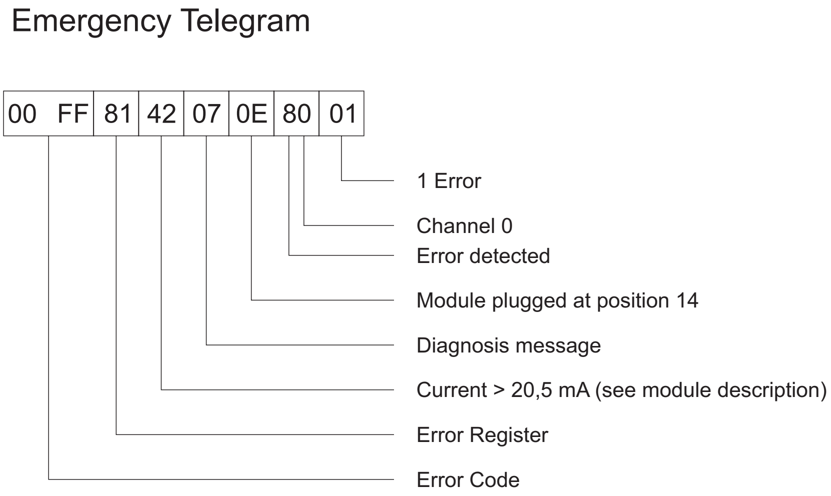


Fig. 5-20: Design Emergency Message 1 g012424x

- In addition to the first error, another error occurs on a 2 channel digital output module 750-506. A wire break on channel 1, the module is connected at position 17.

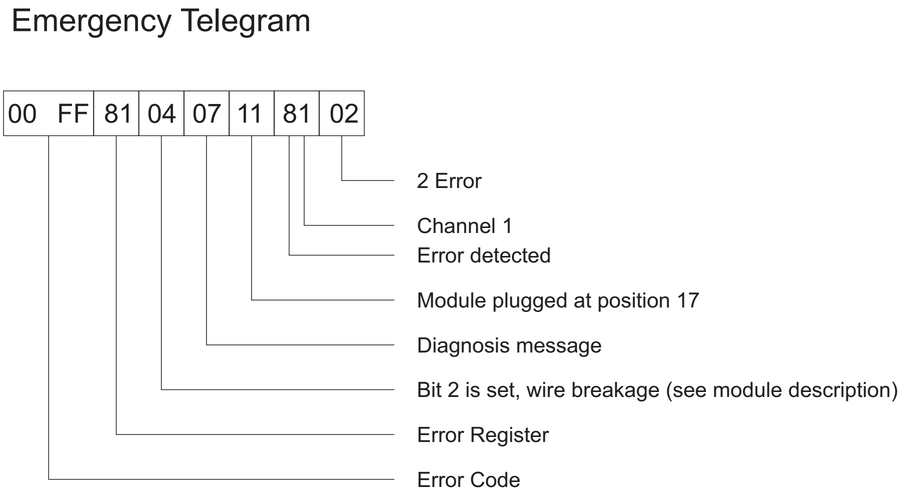


Fig. 5-21: Design Emergency Message2 g012425x

- the occurred error (wire break at digital module 750-506) is overcome.

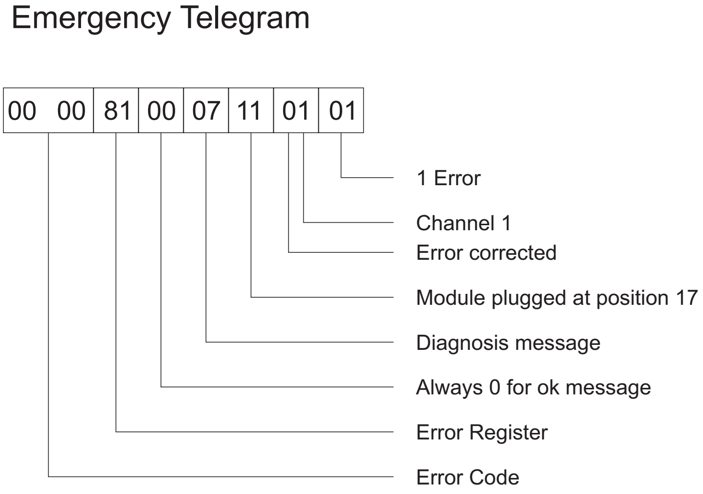


Fig. 5-22: Design Emergency Message 3 g012426x

Note
For the digital modules with diagnostics, the diagnostic data is additionally mapped into the process image. Module 750-506 used as an example hereabove, shows 2 bit in the output process (output values of the 2 channels) and 4 bit in the input process image (2 bit per channel - wire break, short circuit).

6 Literature list

Further information
CAN in Automation (CiA) provides further documentation for its members in INTERNET.
can-cia.de

7 Index