WAGO-I/O-SYSTEM 758

Modular I/O System

WAGO-I/O-IPC
758-870/xxx-xxx

Manual

Technical description,
installation and
configuration

Version 1.1.0

INNOVATIVE CONNECTIONS

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

CONTENT

1 Important Notes

This section provides only a summary of the most important safety requirements and notes which will be mentioned in the individual sections. To protect your health and prevent damage to the devices, it is essential to read and carefully follow the safety guidelines.

1.1 Legal Principles

1.1.1 Copyright

This manual including all figures and illustrations contained therein is subject to copyright. Any use of this manual which infringes the copyright provisions stipulated herein, is not permitted. Reproduction, translation and electronic and phototechnical archiving and amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden. Non-observance will entail the right of claims for damages.

WAGO Kontakttechnik GmbH & Co. KG reserves the right of changes serving technical progress.
All rights developing from the issue of a patent or the legal protection of utility patents are reserved to WAGO Kontakttechnik GmbH & Co. KG. Third-party products are always indicated without any notes concerning patent rights. Thus, the existence of such rights must not be excluded.

1.1.2 Personnel Qualification

The use of the product described in this manual requires special qualifications, as shown in the following table:

Activity	Electrical specialist	Instructed personnel*)	Specialists**) having qualifications in PLC programming
Assembly	X	X	
Commissioning	X		X
Programming			X
Maintenance	X	X	
Troubleshooting	X		
Disassembly	X	X	

*) Instructed persons have been trained by qualified personnel or electrical specialists.

**) A specialist is someone who, through technical training, knowledge and experience, demonstrates the ability to meet the relevant specifications and identify potential dangers in the mentioned field of activity.

All personnel must be familiar with the applicable standards.
WAGO Kontakttechnik GmbH & Co. KG declines any 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 Conforming Use of Series 750

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

The device is designed for IP20 protection class. It is protected against finger touch and solid impurities up to 12.5mm diameter, but not against water penetration. Unless otherwise specified, the device must not be operated in wet and dusty environments.

1.1.4 Technical Condition of the Devices

For each individual application, the components are supplied from the factory with a dedicated hardware and software configuration. Changes in hardware, software and firmware are only admitted within the framework of the possibilities documented in the manuals. All changes to the hardware or software and the non-conforming use of the components entail the exclusion of liability on the part of WAGO Kontakttechnik GmbH & Co. KG.

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

1.2 Standards and Regulations for Operating the 750 Series

Please observe the standards and regulations that are relevant to your installation:

- The data and power lines must be connected and installed in compliance with the standards to avoid failures on your installation and eliminate any danger to personnel.
- For installation, startup, maintenance and repair, please observe the accident prevention regulations of your machine (e.g. BGV A 3, "Electrical Installations and Equipment").
- Emergency stop functions and equipment must not be made ineffective. See relevant standards (e.g. DIN EN 418).
- Your installation must be equipped in accordance to the EMC guidelines so that electromagnetic interferences can be eliminated.
- Operating 750 Series components in home applications without further measures is only permitted if they meet the emission limits (emissions of interference) according to EN 61000-6-3. You will find the relevant information in the section on "WAGO-I/O-SYSTEM 750" → "System Description" → "Technical Data".
- Please observe the safety measures against electrostatic discharge according to DIN EN 61340-5-1/-3. When handling the modules, ensure that the environment (persons, workplace and packing) is well grounded.
- The relevant valid and applicable standards and guidelines concerning the installation of switch cabinets are to be observed.

1.3 Symbols

Danger
Always observe this information to protect persons from injury.

Warning
Always observe this information to prevent damage to the device.

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

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

Note
Make important notes that are to be complied with so that a trouble-free and efficient device operation can be guaranteed.

Additional Information
References to additional literature, manuals, data sheets and internet pages.

1.4 Safety Information

When connecting the device to your installation and during operation, the following safety notes must be observed:

Danger
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 is only permitted via a key or tool to authorized qualified personnel.

Danger
All power sources to the device must always be switched off before carrying out any installation, repair or maintenance work.

Warning
Replace defective or damaged device/module (e.g. in the event of deformed contacts), as the functionality of field bus station in question can no longer be ensured on a long-term basis.

Warning
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 the components must be installed in an enclosure that is resistant against the above mentioned materials. Clean tools and materials are generally required to operate the device/module.

Warning
Soiled contacts must be cleaned using oil-free compressed air or with ethyl alcohol and leather cloths.

Warning
Do not use contact sprays, which could possibly impair the functioning of the contact area.

Warning
Avoid reverse polarity of data and power lines, as this may damage the devices.

ESD (Electrostatic Discharge)
The devices are equipped with electronic components that may be destroyed by electrostatic discharge when touched.

1.5 Font Conventions

italic	Names of paths and files are marked in italic. e.g.: *C:\Programs\WAGO-IO-CHECK*
italic	Menu items are marked in bold italic. e.g.: ***Save***
\	A backslash between two names characterizes the selection of a menu point from a menu. e.g.: ***File \ New***
END	Press buttons are marked as bold with small capitals e.g.: **ENTER**
< >	Keys are marked bold within angle brackets e.g.: **<F5>**
`Courier`	The print font for program codes is Courier. e.g.: `END_VAR`

1.6 Number Notation

Number code	Example	Note
Decimal	100	Normal notation
Hexadecimal	0x64	C notation
Binary	'100' '0110.0100'	Within inverted commas, Nibble separated with dots

1.7 Scope

This manual describes the I/O IPC **758-870/xxx-xxx** with all its variations 758-870/xxx-xxx, including the WAGO-I/O-SYSTEM 750.

1.8 Abbreviations

AO	Analog Output Module
AI	Analog Input Module
DI	Digital Input
DO	Digital Output
I/O	input/output
ID	Identifier
PFC	Programmable Fieldbus Controller

2 The WAGO-I/O-SYSTEM 750

2.1 System Description

The WAGO-I/O-SYSTEM 750 is a modular, field bus independent I/O system. It is comprised of a field bus coupler/controller (1) and connected field bus modules (2) for any type of signal. Together, these make up the field bus node. The end module (3) completes the node.

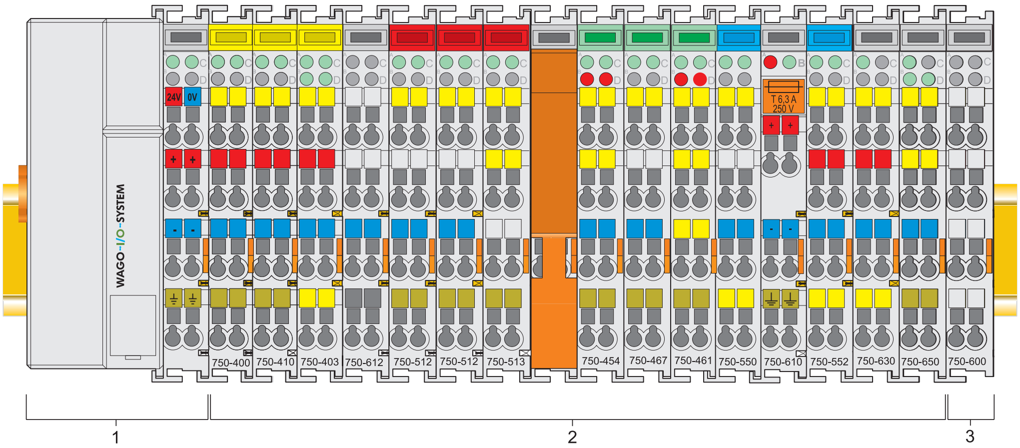


Fig. 2-1: Field bus node g0xxx00x

Couplers/controllers for field bus systems such as PROFIBUS, INTERBUS, ETHERNET TCP/IP, CAN (CANopen, DeviceNet, CAL), MODBUS, LON and others are available.

The coupler/controller contains the field bus interface, electronics and a power supply terminal. The field bus interface forms the physical interface to the relevant field bus. The electronics process the data of the bus modules and make it available for the field bus communication. The 24 V system supply and the 24 V field supply are fed in via the integrated power supply terminal. The field bus coupler communicates via the relevant field bus. The programmable field bus controller (PFC) enables the implementation of additional PLC functions. Programming is done with the **WAGO-I/O-*PRO*** in accordance with IEC 61131-3.

Bus modules for diverse digital and analog I/O functions as well as special functions can be connected to the coupler/controller. The communication between the coupler/controller 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 status indication, insertable mini WSB markers and pullout group marker carriers. The 3-wire technology supplemented by a ground wire connection allows for direct sensor/actuator wiring.

2.2 Technical Data

Mechanic	
Material	Polycarbonate, Polyamide 6.6
Dimensions W x H* x L * from upper edge of DIN 35 rail - Coupler/Controller (Standard) - Coupler/Controller (ECO) - Coupler/Controller (FireWire) - I/O module, single - I/O module, double - I/O module, fourfold	 - 51 mm x 65 mm x 100 mm - 50 mm x 65 mm x 100 mm - 62 mm x 65 mm x 100 mm - 12 mm x 64 mm x 100 mm - 24 mm x 64 mm x 100 mm - 48 mm x 64 mm x 100 mm
Installation	on DIN 35 with interlock
Modular by	double featherkey-dovetail
Mounting position	any position
Marking	standard marking label type group marking label 8 x 47 mm
Connection	
Connection type	CAGE CLAMP®
Wire range	0.08 mm² ... 2.5 mm², AWG 28-14
Stripped length	8 … 9 mm, 9 … 10 mm for components with pluggable wiring (753-xxx)
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 1.5 µm, self-cleaning
Climatic environmental conditions	
Operating temperature	0 °C ... 55 °C, -20 °C … +60 °C for components with extended temperature range (750-xxx/025-xxx)
Storage temperature	-20 °C ... +85 °C
Relative humidity	5 % … 95 % without condensation
Resistance to harmful substances	acc. to IEC 60068-2-42 and IEC 60068-2-43
Maximum pollutant concentration at relative humidity < 75%	$SO_2 \leq 25$ ppm $H_2S \leq 10$ ppm
Special conditions	Ensure that additional measures for components are taken, which are used in an environment involving: – dust, caustic vapors or gases – ionization radiation

Safe electrical isolation	
Air and creepage distance	acc. to IEC 60664-1
Degree of pollution acc. To IEC 61131-2	2

Degree of protection	
Degree of protection	IP 20

Electromagnetic compatibility

Immunity to interference for industrial areas acc. to EN 61000-6-2 (2001)

Test specification	**Test values**		**Strength class**	**Evaluation criteria**
EN 61000-4-2 ESD	4 kV/8 kV (contact/air)		2/3	B
EN 61000-4-3 electromagnetic fields	10 V/m 80 MHz ... 1 GHz		3	A
EN 61000-4-4 burst	1 kV/2 kV (data/supply)		2/3	B
EN 61000-4-5 surge	Data:	-/- (line/line)		B
		1 kV (line/earth)	2	
	DC sup- ply:	0.5 kV (line/line)	1	B
		0.5 kV (line/earth)	1	
	AC sup- ply:	1 kV (line/line)	2	B
		2 kV (line/earth)	3	
EN 61000-4-6 RF disturbances	10 V/m 80 % AM (0.15 ... 80 MHz)		3	A

Emission of interference for industrial areas acc. to EN 61000-6-4 (2001)

Test specification	**Limit values/[QP]*)**	**Frequency range**	**Distance**
EN 55011 (AC supply, conducted)	79 dB (µV)	150 kHz ... 500 kHz	
	73 dB (µV)	500 kHz ... 30 MHz	
EN 55011 (radiated)	40 dB (µV/m)	30 MHz ... 230 MHz	10 m
	47 dB (µV/m)	230 MHz ... 1 GHz	10 m

Emission of interference for residential areas acc. to EN 61000-6-3 (2001)

Test specification	**Limit values/[QP]*)**	**Frequency range**	**Distance**
EN 55022 (AC supply, conducted)	66 ... 56 dB (µV)	150 kHz ... 500 kHz	
	56 dB (µV)	500 kHz ... 5 MHz	
	60 dB (µV)	5 MHz ... 30 MHz	
EN 55022 (DC supply/data, conducted)	40 ... 30 dB (µA)	150 kHz ... 500 kHz	
	30 dB (µA)	500 kHz ... 30 MHz	
EN 55022 (radiated)	30 dB (µV/m)	30 MHz ... 230 MHz	10 m
	37 dB (µV/m)	230 MHz ... 1 GHz	10 m

Mechanical strength acc. to IEC 61131-2		
Test specification	**Frequency range**	**Limit value**
IEC 60068-2-6 vibration	5 Hz ≤ f < 9 Hz	1.75 mm amplitude (permanent) 3.5 mm amplitude (short term)
	9 Hz ≤ f < 150 Hz	0.5 g (permanent) 1 g (short term)
	Note on vibration test: a) Frequency change: max. 1 octave/minute b) Vibration direction: 3 axes	
IEC 60068-2-27 shock		15 g
	Note on shock test: a) Type of shock: half sine b) Shock duration: 11 ms c) Shock direction: 3x in positive and 3x in negative direction for each of the three mutually perpendicular axes of the test specimen	
IEC 60068-2-32 free fall		1 m (module in original packing)

*) QP: Quasi Peak

Note:
If the technical data of components differ from the values described here, the technical data shown in the manuals of the respective components shall be valid.

For Products of the WAGO-I/O-SYSTEM 750 with ship specific approvals supplementary guidelines are valid:

<table>
<tr><th colspan="5">Electromagnetic compatibility</th></tr>
<tr><th colspan="5">Immunity to interference acc. to Germanischer Lloyd (2003)</th></tr>
<tr><th>Test specification</th><th colspan="2">Test values</th><th>Strength class</th><th>Evaluation criteria</th></tr>
<tr><td>IEC 61000-4-2 ESD</td><td colspan="2">6 kV/8 kV (contact/air)</td><td>3/3</td><td>B</td></tr>
<tr><td>IEC 61000-4-3 electromagnetic fields</td><td colspan="2">10 V/m 80 MHz ... 2 GHz</td><td>3</td><td>A</td></tr>
<tr><td>IEC 61000-4-4 burst</td><td colspan="2">1 kV/2 kV (data/supply)</td><td>2/3</td><td>A</td></tr>
<tr><td rowspan="2">IEC 61000-4-5 surge</td><td rowspan="2">AC/DC Supply:</td><td>0.5 kV (line/line)</td><td>1</td><td rowspan="2">A</td></tr>
<tr><td>1 kV (line/earth)</td><td>2</td></tr>
<tr><td>IEC 61000-4-6 RF disturbances</td><td colspan="2">10 V/m 80 % AM (0.15 ... 80 MHz)</td><td>3</td><td>A</td></tr>
<tr><td>Type test AF disturbances (harmonic waves)</td><td colspan="2">3 V, 2 W</td><td>-</td><td>A</td></tr>
<tr><td>Type test high voltage</td><td colspan="2">755 V DC
1500 V AC</td><td>-</td><td>-</td></tr>
<tr><th colspan="5">Emission of interference acc. to Germanischer Lloyd (2003)</th></tr>
<tr><th>Test specification</th><th colspan="2">Limit values</th><th>Frequency range</th><th>Distance</th></tr>
<tr><td rowspan="3">Type test
(EMC1, conducted)
allows for ship bridge control applications</td><td colspan="2">96 ... 50 dB (µV)</td><td>10 kHz ... 150 kHz</td><td></td></tr>
<tr><td colspan="2">60 ... 50 dB (µV)</td><td>150 kHz ... 350 kHz</td><td></td></tr>
<tr><td colspan="2">50 dB (µV)</td><td>350 kHz ... 30 MHz</td><td></td></tr>
<tr><td rowspan="4">Type test
(EMC1, radiated)
allows for ship bridge control applications
except:</td><td colspan="2">80 ... 52 dB (µV/m)</td><td>150 kHz ... 300 kHz</td><td>3 m</td></tr>
<tr><td colspan="2">52 ... 34 dB (µV/m)</td><td>300 kHz ... 30 MHz</td><td>3 m</td></tr>
<tr><td colspan="2">54 dB (µV/m)</td><td>30 MHz ... 2 GHz</td><td>3 m</td></tr>
<tr><td colspan="2">24 dB (µV/m)</td><td>156 MHz ... 165 MHz</td><td>3 m</td></tr>
<tr><th colspan="5">Mechanical strength acc. to Germanischer Lloyd (2003)</th></tr>
<tr><th>Test specification</th><th colspan="2">Frequency range</th><th colspan="2">Limit value</th></tr>
<tr><td rowspan="3">IEC 60068-2-6 vibration
(category A – D)</td><td colspan="2">2 Hz ≤ f < 25 Hz</td><td colspan="2">± 1.6 mm amplitude (permanent)</td></tr>
<tr><td colspan="2">25 Hz ≤ f < 100 Hz</td><td colspan="2">4 g (permanent)</td></tr>
<tr><td colspan="4">Note on vibration test:
a) Frequency change: max. 1 octave/minute
b) Vibration direction: 3 axes</td></tr>
</table>

Range of application	Required specification emission of interference	Required specification immunity to interference
Industrial areas	EN 61000-6-4 (2001)	EN 61000-6-2 (2001)
Residential areas	EN 61000-6-3 (2001)*)	EN 61000-6-1 (2001)

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

ETHERNET	750-342/-841/-842/-860
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 field bus 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 field bus couplers/controllers under certain boundary conditions. Please contact WAGO Kontakttechnik GmbH & Co. KG.

Maximum power dissipation of the components	
Bus modules	0.8 W / bus terminal (total power dissipation, system/field)
Field bus coupler/controller	2.0 W / coupler/controller

Warning
The power dissipation of all installed components must not exceed the maximum 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

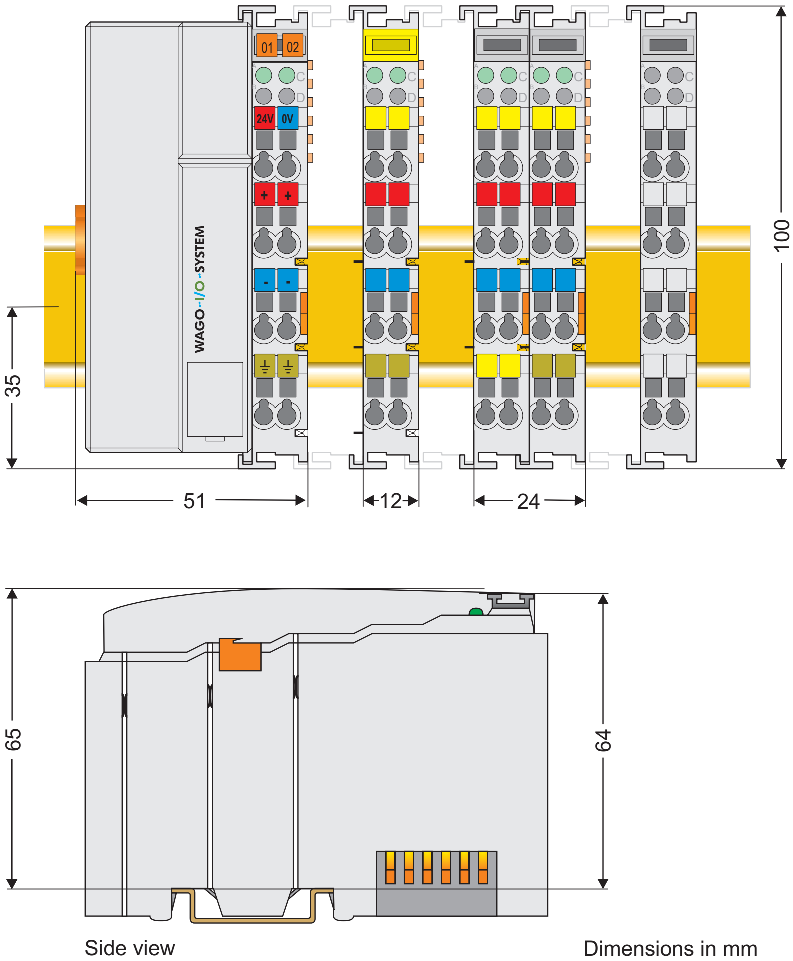


Fig. 2-2: Dimensions g01xx05e

Note:
The illustration shows a standard coupler. For detailed dimensions, please refer to the technical data of the respective coupler/controller.

2.3 Manufacturing Number

The manufacturing number indicates the delivery status directly after production.
This number is part of the lateral marking on the component.
In addition, starting from calendar week 43/2000 the manufacturing number is also printed on the cover of the configuration and programming interface of the field bus coupler or controller.

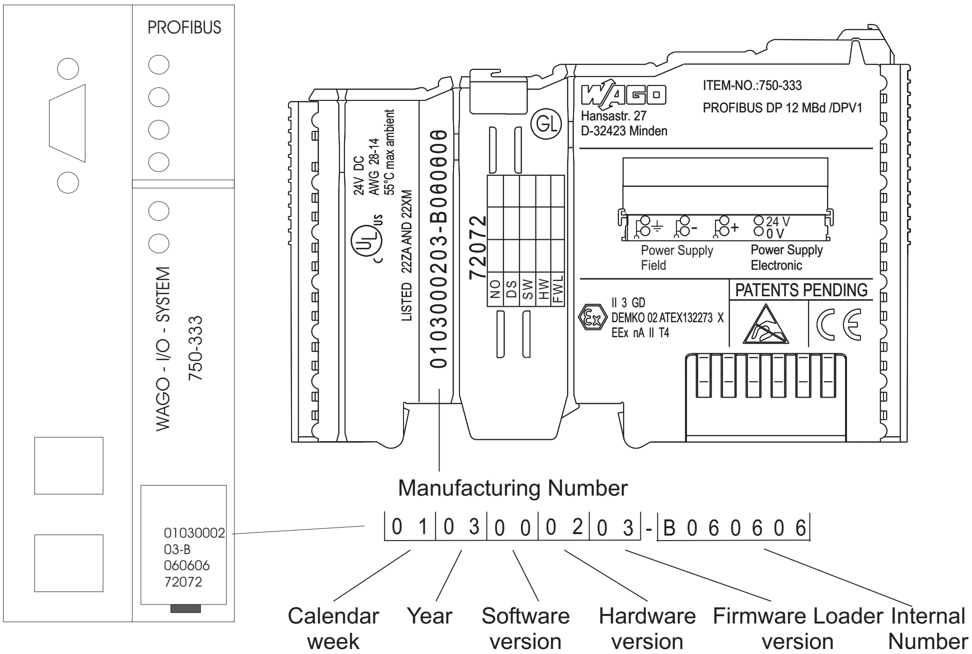


Fig. 2-3: Example: Manufacturing Number of a PROFIBUS field bus coupler 750-333

g01xx15e

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

2.4 Component Update

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

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

Update Matrix

Current Version data for:		1. Update	2. Update	3. Update	
Production Order Number	**NO**				← only starting from calendar week 13/2004
Datestamp	**DS**				
Software index	**SW**				
Hardware index	**HW**				
Firmware loader index	**FWL**				← only for coupler/ controller

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

Additionally with the update of a field bus coupler or controller also the cover of the configuration and programming interface of the coupler or controller is printed on with the current manufacturing and production order number.

The original manufacturing data on the housing of the component remain thereby.

2.5 Storage, Assembly and Transport

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

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

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.

2.6 Mechanical Setup

2.6.1 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 DIN 35 rail, 6 mm wide
WAGO item 249-117	End stop for DIN 35 rail, 10 mm wide

2.6.2 Total Expansion

The length of the module assembly (including one end module of 12mm width) that can be connected to the coupler/controller is 780 mm. When assembled, the I/O modules have a maximum length of 768 mm.

Examples:

- 64 I/O modules of 12 mm width can be connected to one coupler/controller.
- 32 I/O modules of 24 mm width can be connected to one coupler/controller.

Exception:

The number of connected I/O modules also depends on which type of coupler/controller is used. For example, the maximum number of I/O modules that can be connected to a PROFIBUS coupler/controller is 63 without end module. The maximum total expansion of a node is calculated as follows:

Warning
The maximum total length of a node without coupler/controller must not exceed 780 mm. Furthermore, restrictions made on certain types of couplers/controllers must be observed (e.g. for PROFIBUS).

2.6.3 Assembly onto Carrier Rail

2.6.3.1 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 Kontakttechnik GmbH & Co. KG supplies standardized carrier rails that are optimal for use with the I/O system. If other carrier rails are used, then a technical inspection and approval of the rail by WAGO Kontakttechnik GmbH & Co. KG should take place.

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

- The material must be non-corrosive.
- Most components have a contact to the carrier rail to ground electromagnetic 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 mounting points (every 20 cm) should be used in order to prevent bending and twisting (torsion).
- The geometry of the carrier rail must not be altered in order to secure the safe hold of the components. In particular, when shortening or mounting the carrier rail, it must not be crushed or bent.
- The base of the I/O components extends into the profile of the carrier rail. For carrier rails with a height of 7.5 mm, mounting points are to be riveted under the node in the carrier rail (slotted head captive screws or blind rivets).

2.6.3.2 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

2.6.4 Spacing

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

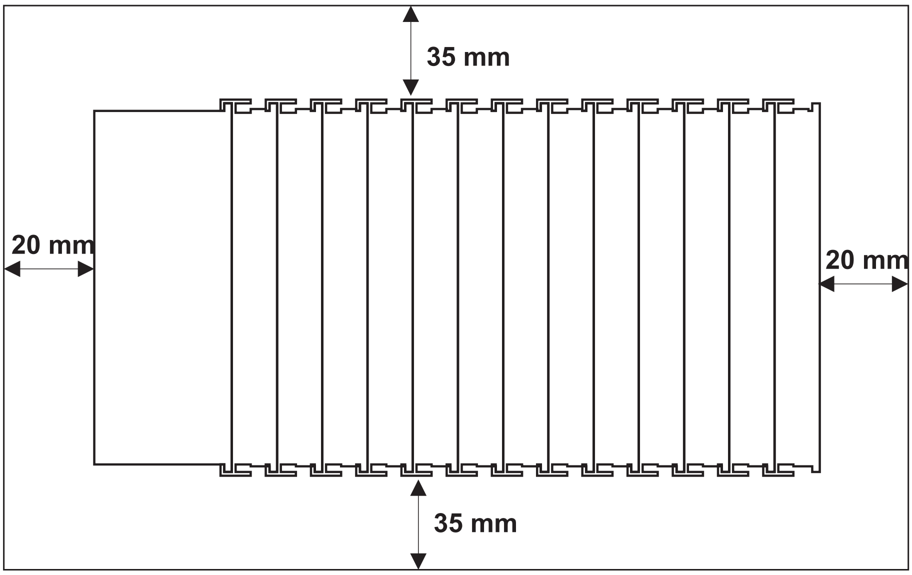


Fig. 2-4: Spacing g01xx13x

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

2.6.5 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 coupler/controller from jamming, it should be fixed onto the carrier rail with the locking disc To do so, push on the upper groove of the locking disc using a screwdriver.

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

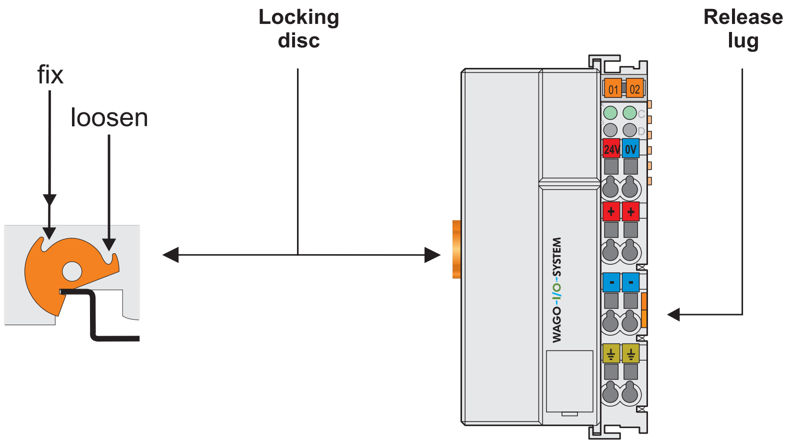


Fig. 2-5: Coupler/Controller and unlocking lug g01xx12e

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

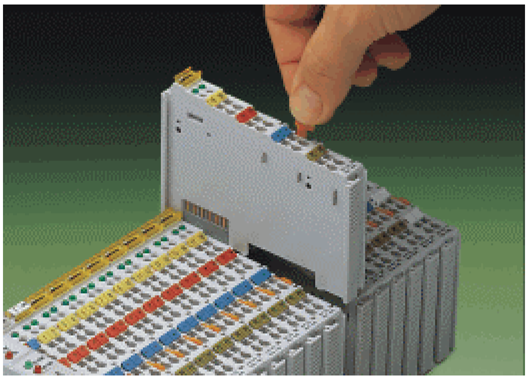


Fig. 2-6: removing bus terminal p0xxx01x

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 2.6.3.

2.6.6 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 coupler/controller, 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 coupler/controller, and 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 field bus node with an end module (750-600).

2.6.7 Internal Bus/Data Contacts

Communication between the coupler/controller 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.

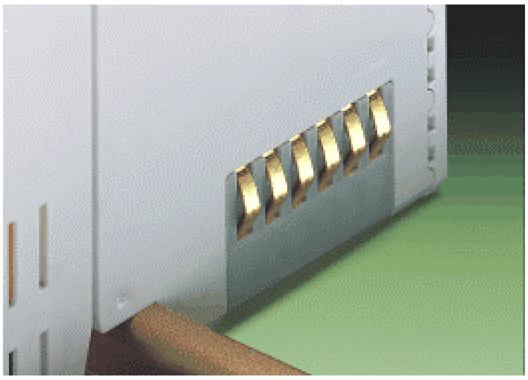


Fig. 2-7: Data contacts p0xxx07x

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

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. data contacts.

2.6.8 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

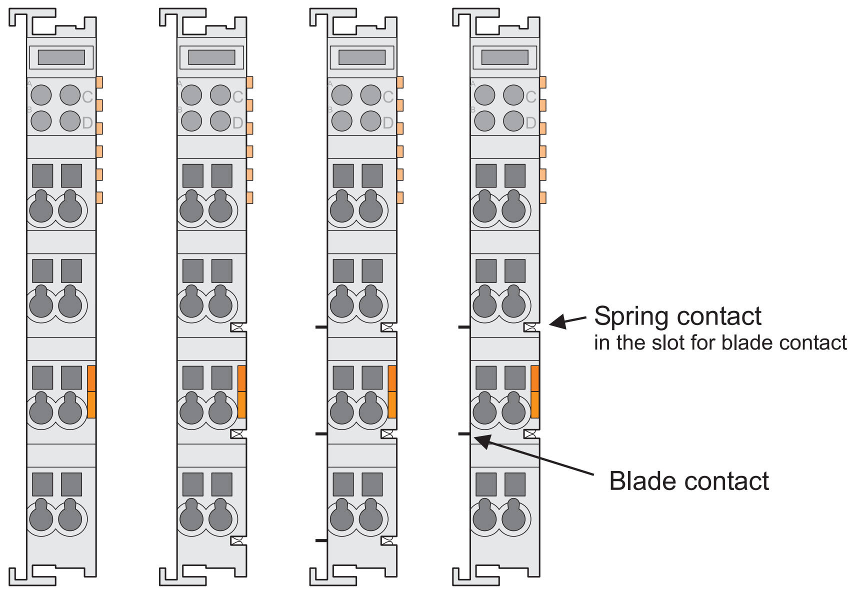


Fig. 2-8: Example for the arrangement of power contacts g0xxx05e

Recommendation
With the WAGO ProServe® Software smartDESIGNER, the structure of a field bus node can be configured. The configuration can be tested via the integrated accuracy check.

2.6.9 Wire Connection

All components have CAGE CLAMP® connections.

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

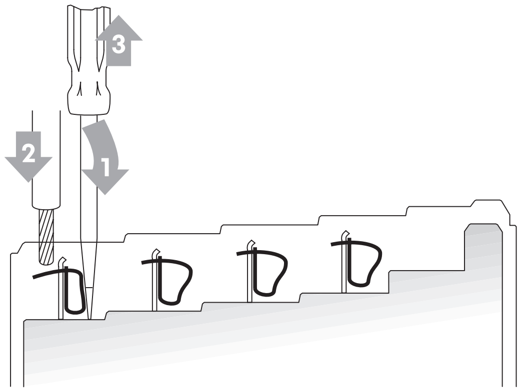


Fig. 2-9: CAGE CLAMP® Connection g0xxx08x

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 made at one connection point, then they should be made away from the connection point using WAGO Terminal Blocks. The terminal blocks may be jumpered together and a single wire brought back to the I/O module connection point.

Attention
If it is unavoidable to jointly connect 2 conductors, then a ferrule must be used to join the wires together.
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

2.7 Power Supply

2.7.1 Isolation

Within the field bus node, there are three electrically isolated potentials.

- Operational voltage for the field bus 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 digital and analog input modules have each channel electrically isolated, please see catalog.

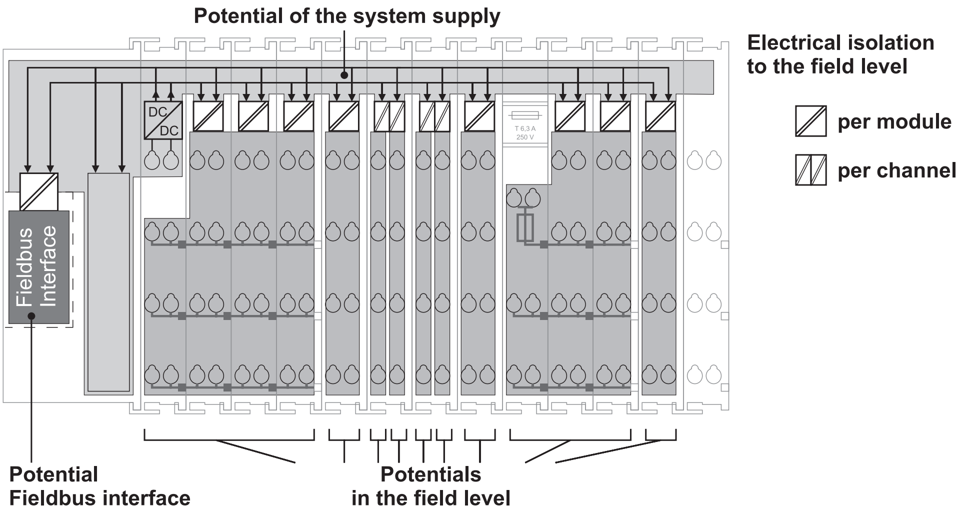


Fig. 2-10: Isolation g0xxx01e

Attention
The ground wire connection must be present in each group. In order that all protective conductor functions are maintained under all circumstances, it is recommended that a ground wire be connected at the beginning and end of a potential group. (ring format, please see chapter 2.8.3). 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 eliminated for the potential group.

2.7.2 System Supply

2.7.2.1 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.

Attention
The use of an incorrect supply voltage or frequency can cause severe damage to the component.

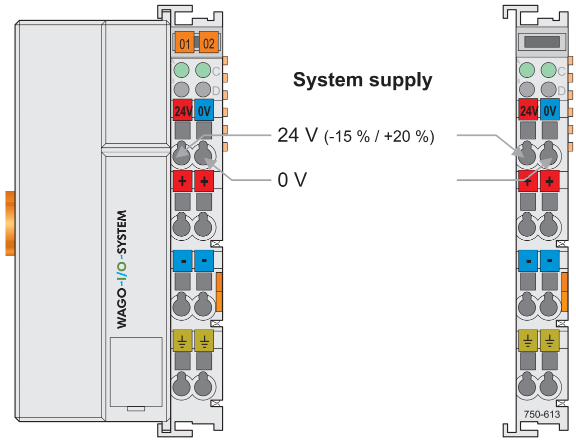


Fig. 2-11: System Supply g0xxx02e

The direct current supplies all internal system components, e.g. coupler/controller electronics, field bus 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.

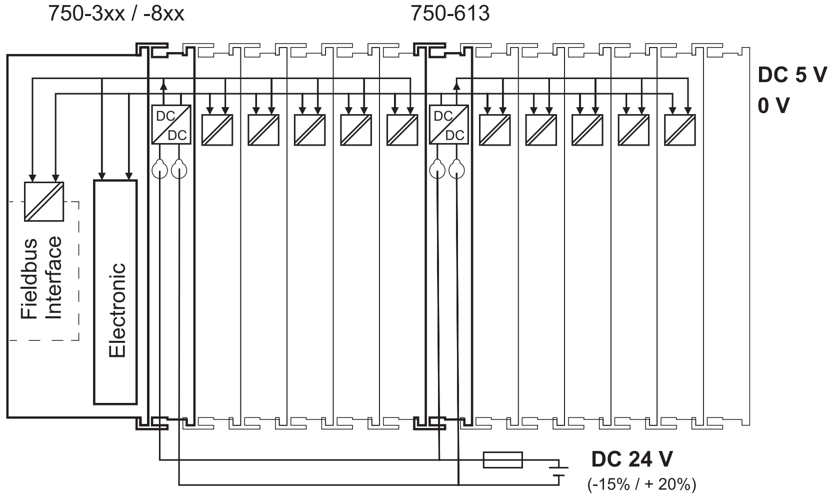


Fig. 2-12: System Voltage g0xxx06e

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

2.7.2.2 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 coupler/controller 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 coupler/controller
Residual current for bus terminals*)	Available current for the bus modules. Provided by the bus power supply unit. See coupler/controller and internal system supply module (750-613)

*) cf. catalogue W4 Volume 3, manuals or internet

Example	**Coupler 750-301:**	
	internal current consumption:350 mA at 5V	
	residual current for	
	bus modules:	1650 mA at 5V
	sum $I_{(5V)\ total}$:	2000 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 Coupler 750-333 consists of 20 relay modules (750-517) and 10 digital input modules (750-405).

Current consumption:
20* 90 mA = 1800 mA
10* 2 mA = 20 mA
Sum 1820 mA

The coupler can provide 1650 mA for the bus modules. Consequently, an internal system supply module (750-613), e.g. in the middle of the node, should be added.

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

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

Coupler/Controller

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

750-613

$I_{(5\ V)\ total}$ = *Sum of all the internal current consumption of the connected bus modules*

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

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

Note
If the electrical consumption of the power supply point for the 24 V-system supply exceeds 500 mA, 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.

2.7.3 Field Supply

2.7.3.1 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.

The coupler/controller provides field side power (DC 24V). In this case it is a passive power supply without protection equipment.
Power supply modules are available for other potentials, e. g. 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.

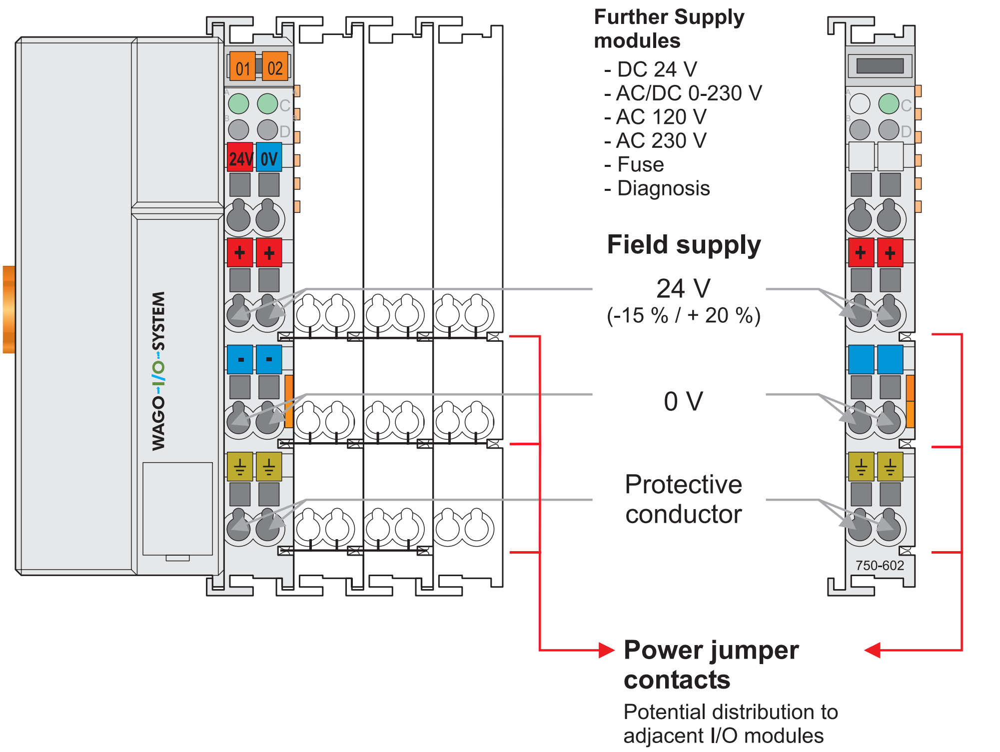


Fig. 2-13: Field Supply (Sensor/Actuator) g0xxx03e

The supply voltage for the field side is automatically passed to the next module 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 inserting an additional power supply module, the field supply via the power contacts is disrupted. From there a new power supply occurs which may also contain a new voltage potential.

Attention
Some bus modules have no or very few power contacts (depending on the I/O function). Due to this, the passing through of the relevant potential is disrupted. If 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.

2.7.3.2 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

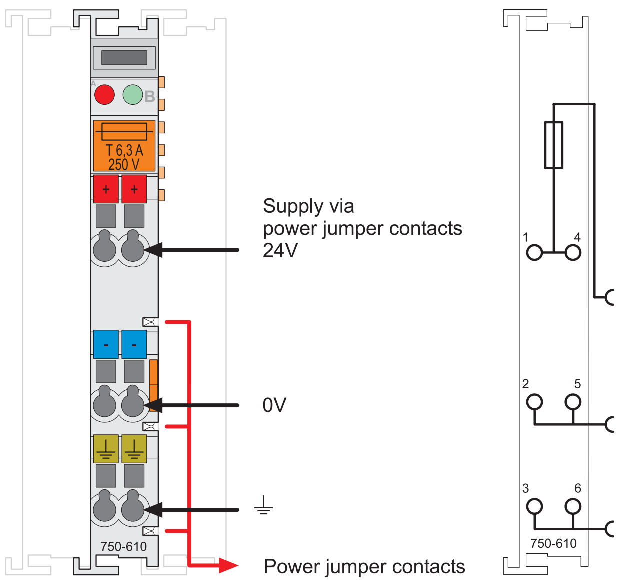


Fig. 2-14: Supply module with fuse carrier (Example 750-610) g0xxx09x

Warning
In the case of power supply modules with fuse holders, only fuses with a maximum 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.

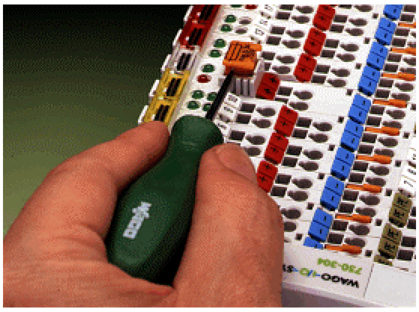


Fig. 2-15: Removing the fuse carrier p0xxx05x

Lifting the cover to the side opens the fuse carrier.

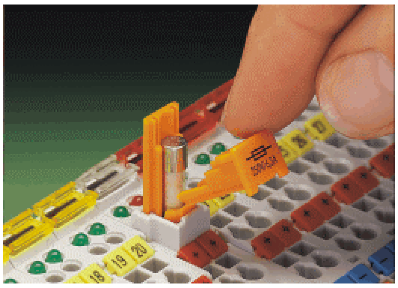


Fig. 2-16: Opening the fuse carrier p0xxx03x

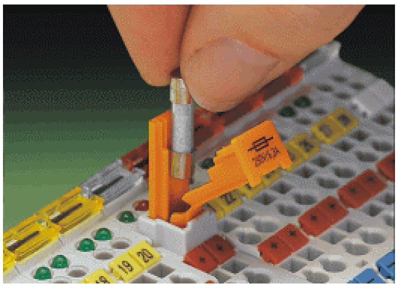


Fig. 2-17: Change fuse p0xxx04x

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.

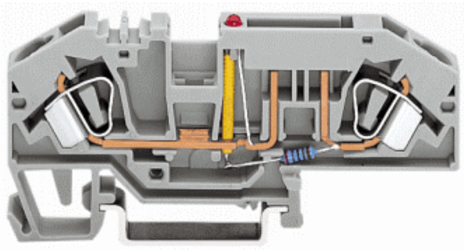


Fig. 2-18: Fuse modules for automotive fuses, series 282 pf66800x

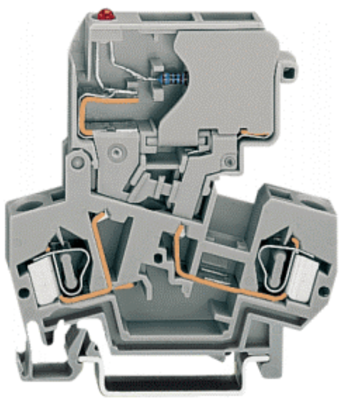


Fig. 2-19: Fuse modules with pivotable fuse carrier, series 281 pe61100x

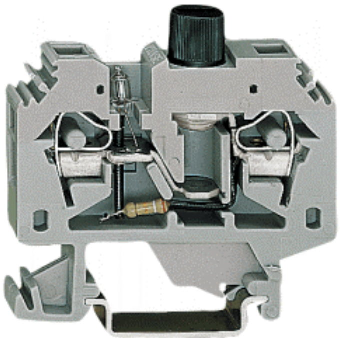


Fig. 2-20: Fuse modules, series 282 pf12400x

2.7.4 Supplementary Power Supply Regulations

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

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

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

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

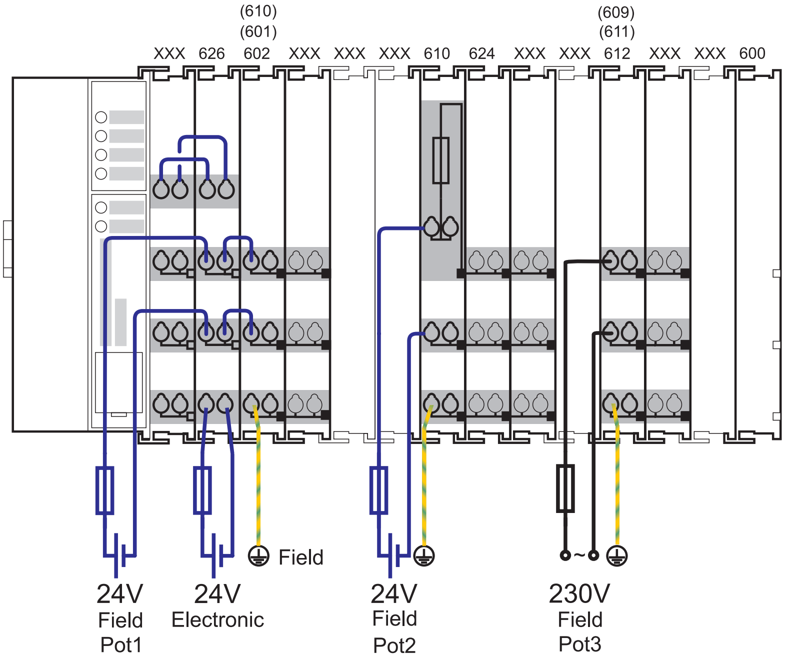


Fig. 2-21: Power supply concept g01xx11e

Note
Another potential power terminal 750-601/602/610 must only be used behind the filter terminal 750-626 if the protective earth conductor is needed on the lower power contact or if a fuse protection is required.

2.7.5 Supply Example

Attention
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.

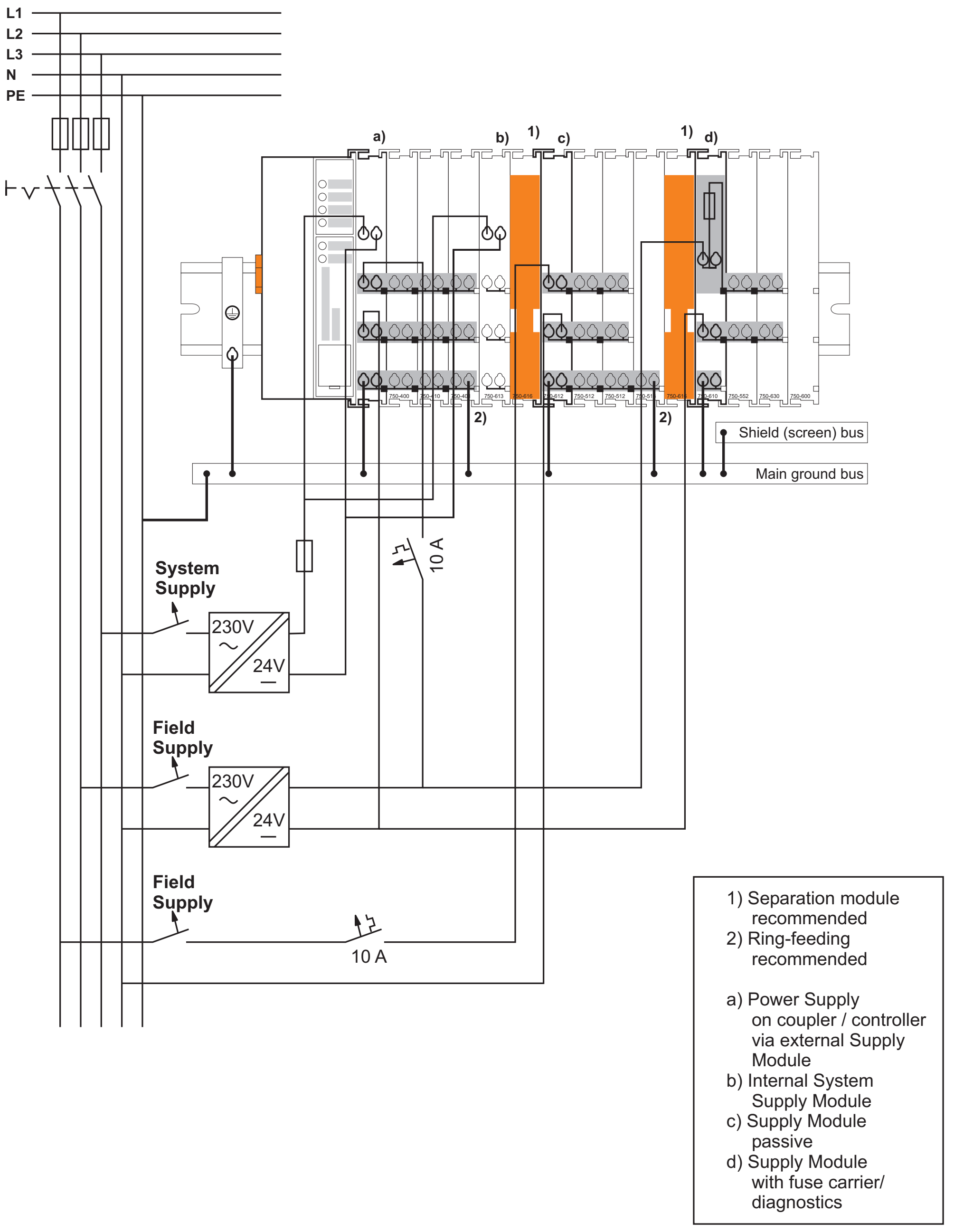


Fig. 2-22: Supply example

g0xxx04e

2.7.6 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 approx 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 from the power supplies in order to ensure bus operation in the event of short circuits on the actuator side.

WAGO products Item 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
 288-809 288-810 288-812 288-813	Rail-mounted modules with universal mounting carrier AC 115 V / DC 24 V; 0,5 A AC 230 V / DC 24 V; 0,5 A AC 230 V / DC 24 V; 2 A AC 115 V / DC 24 V; 2 A

2.8 Grounding

2.8.1 Grounding the DIN Rail

2.8.1.1 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.

2.8.1.2 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.

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

2.8.2 Grounding Function

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

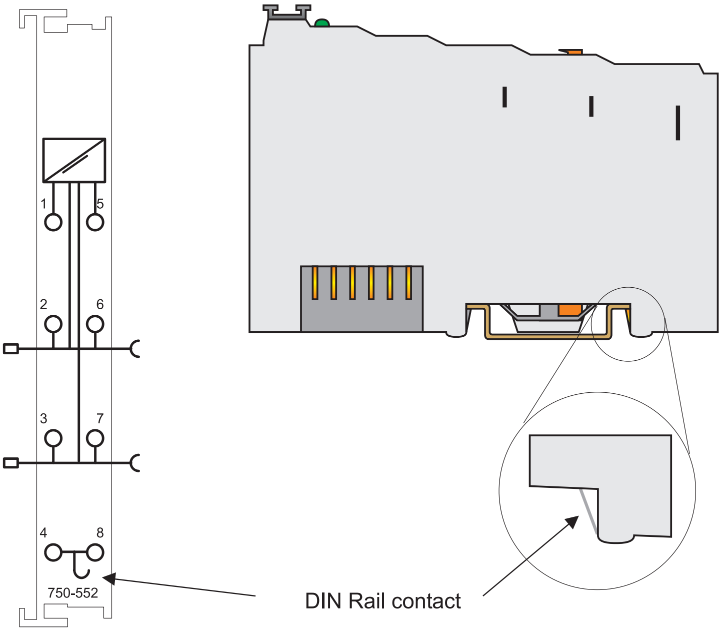


Fig. 2-23: Carrier rail contact g0xxx10e

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

The carrier rail must be grounded.

For information on carrier rail properties, please see chapter 2.6.3.2.

2.8.3 Grounding Protection

For the field side, the ground wire is connected to the lowest connection terminals of the power supply module. The ground connection is then connected to the next module via the Power Jumper Contact (PJC). If the bus module has the lower power jumper contact, then the ground wire connection of the field devices can be directly connected to the lower connection terminals of the bus module.

Attention
Should the ground conductor connection of the power jumper contacts within the node become disrupted, e. g. due to a 4-channel bus terminal, the ground connection will need to be re-established.

The ring feeding of the grounding potential will increase the system safety. When one bus module is removed from the group, the grounding connection will remain intact.

The ring feeding method has the grounding conductor connected to the beginning and end of each potential group.

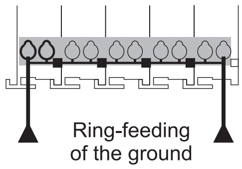


Fig. 2-24: Ring-feeding g0xxx07e

Attention
The regulations relating to the place of assembly as well as the national regulations for maintenance and inspection of the grounding protection must be observed.

2.9 Shielding (Screening)

2.9.1 General

The shielding of the data and signal conductors reduces electromagnetic interferences thereby increasing the signal quality. Measurement errors, data transmission errors and even disturbances caused by overvoltage can 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 separated from all high-voltage cables.

The cable shield should be 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.

2.9.2 Bus Conductors

The shielding of the bus conductor is described in the relevant assembly guidelines and standards of the bus system.

2.9.3 Signal Conductors

Bus modules for most analog signals along with many of the interface bus modules include a connection for the shield.

Note
For a better shield performance, the shield should have previously been placed over a large area. The WAGO shield connection system is suggested for such an application.
This suggestion is especially applicable if the equipment can have even current or high impulse formed currents running through (for example initiated by atmospheric discharge).

2.9.4 WAGO Shield (Screen) Connecting System

The WAGO Shield Connecting system includes a shield clamping saddle, a collection of rails and a variety of mounting feet. Together these allow many different possibilities. See catalog W4 volume 3 chapter 10.

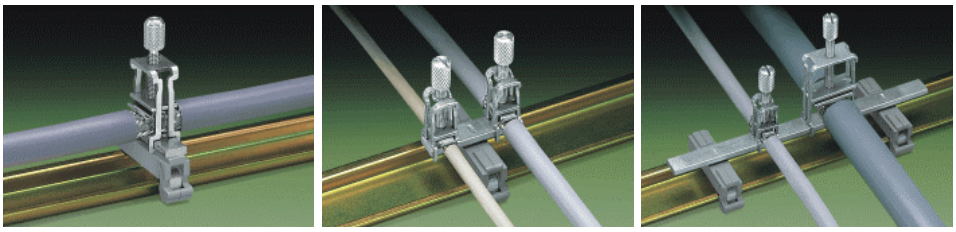


Fig. 2-25: WAGO Shield (Screen) Connecting System p0xxx08x, p0xxx09x, and p0xxx10x

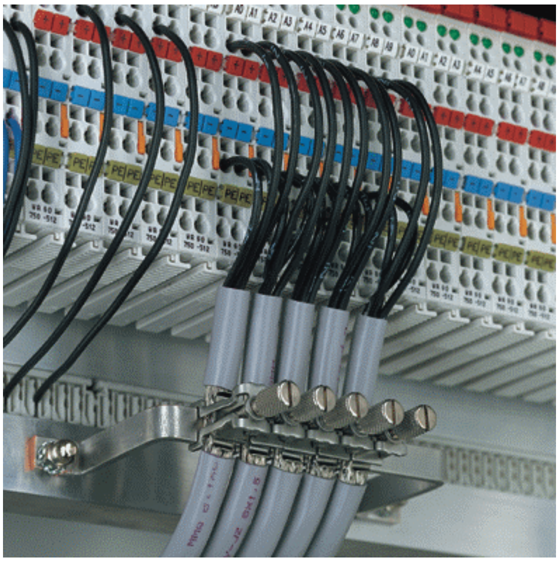


Fig. 2-26: Application of the WAGO Shield (Screen) Connecting System p0xxx11x

2.10 Assembly Guidelines/Standards

DIN 60204,	Electrical equipping of machines
DIN EN 50178	Equipping of high-voltage systems with electronic components (replacement for VDE 0160)
EN 60439	Low voltage – switch box combinations

3 Overview of all available I/O-IPC Versions of the WAGO-I/O-IPC 758-870/xxx-xxx

The descriptions for the fieldbuses PROFIBUS and CANopen for the I/O-IPC starts in section 3.2.

32-MB Versions

I/O-IPC	758-870	758-870/ 000-001	758-870/ 000-002	758-870/ 000-004
Fieldbus connection	-	PROFIBUS	CANopen	PROFIBUS
Function	-	Master	-	Slave
Memory size	32 MB	32 MB	32 MB	32 MB
Comments	Programmable in accordance with IEC 61131-3, without CoDeSys Web visualization.			

128-MB Versions

I/O-IPC	758-870 000-010	758-870/ 000-011	758-870/ 000-012	758-870/ 000-014
Fieldbus Connection	-	PROFIBUS	CANopen	PROFIBUS
Function	-	Master	-	Slave
Memory size	128 MB	128 MB	128 MB	128 MB
Comments	Programmable in accordance with IEC 61131-3, with CoDeSys Web visualization.			

Currently, the PROFIBUS master only supports DPV0 services.

3.1 WAGO-I/O-IPC 758-870 and 758-870/000-010

3.1.1 Description

The I/O-IPC is an industrial PC that combines the functionalities of a PC and a programmable logic control (PLC). I/O modules and the PROFIBUS or CANopen fieldbus can be connected. The fieldbus connections depend on the version of the IPC. All versions have two 10/100BASE-TX Ethernet interfaces.

If using the I/O-IPC as a PLC, it is possible to control all or some of the I/O modules locally by using WAGO-I/O-*PRO* CAA (CoDeSys). WAGO-I/O-*PRO* CAA is an IEC 61131-3 compliant programming tool for programming and configuring the I/O-IPC.

The protocols HTTP, BootP, DHCP, SNTP, FTP and SMTP can be used for system management and system diagnostics.

In addition to Ethernet (TCP/UDP) the version-dependent fieldbus protocols PROFIBUS and CANopen are also supported.

Additional library functions extend the versatility of programming functions. The IEC 61131-3 library "SysLibRTC.lib," for example, allows the integration of a battery-backed real-time clock with date (resolution 1 second), alarm function and timer. If the power supply is interrupted, the clock will be powered by a battery.

The I/O-IPC is based on a Pentium compatible MMX CPU and is a multitasking tool; i.e., several programs can run simultaneously.

The I/O-IPC has an internal server for web-based applications. As standard, the HTML pages in the I/O-IPC contain information about the configuration and the status of the I/O-IPC. A web browser will open and display the HTML pages. Furthermore, it provides a file system that saves individual HTML pages in the I/O-IPC via FTP download.

3.1.2 Hardware Description

3.1.2.1 View

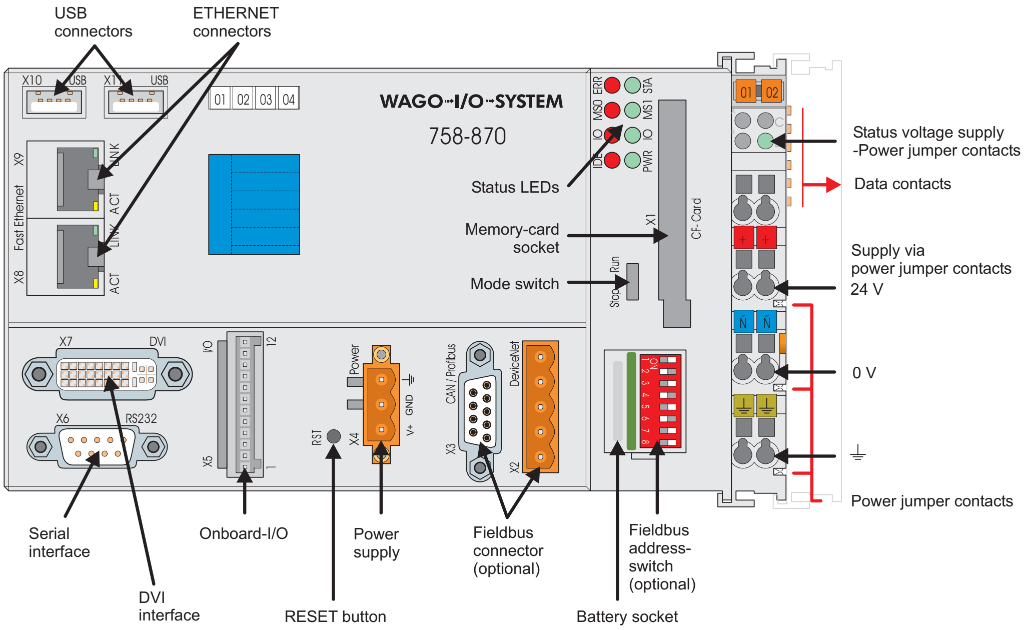


Fig. 3-1: I/O-IPC

The I/O-IPC consists of the following components:

- computer unit with interfaces for LAN, USB devices, I/O modules, screen as well as slot for Compact Flash cards, type I and II.
- power supply unit for the system supply and power jumper contacts for the field supply of the connected I/O modules.
- four integrated digital inputs and four outputs, independent of the I/O modules connected to the internal bus.
- one internal bus interface for I/O modules from the WAGO-I/O-SYSTEM 750.
- optional fieldbus interface for PROFIBUS or CANopen.
- indicators (LEDs) for operating status, diagnostics and bus communication.
- mode switch (reset, run/stop, DIP switch for fieldbus address).

3.1.2.2 Power Supply

Attention
The supply cable for the 24 V system supply must be provided with a ferrite core; e.g., Würth 74271112! The length of the supply line must not exceed 10 m!

Use the X4 connection for the I/O-IPC power supply. The power supply unit provides the required voltage to power to the electronics and the I/O modules.

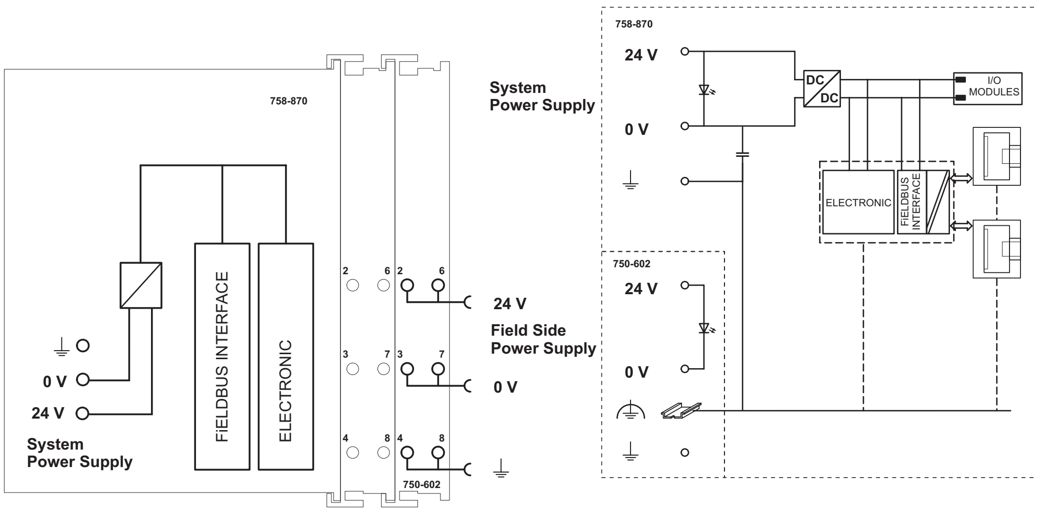


Fig. 3-2: Power supply

Attention:
Do not use the I/O-IPC module to supply power to the power jumper contacts (field supply).
Only use the 750-602 supply module that is provided with the I/O-IPC!

The internal electronics of the I/O-IPC and the I/O modules are electrically isolated from the field connections and the field devices by means of DC/DC converters and optocouplers.

3.1.2.3 Battery Compartment

The battery compartment holds a 3.3 V lithium battery (type CR2032) that supplies the real-time clock and the SRAM with power in case of a power failure.

If changing the lithium battery during a power failure, ensure the replacement battery is on hand. The gold cap capacitor can only supply power for a limited time. The data of the volatile memory (SRAM) is therefore preserved during a battery exchange.

At room temperature, the battery has a life span of approximately one year.

It is not possible to check the capacity of the battery via CoDeSys function SysRtcCheckBattery.

3.1.3 Description of the Integrated Interfaces

3.1.3.1 Slot for CF card (X1)

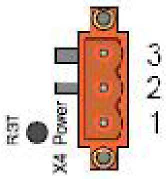
This is a standard IDE interface that supports CF cards (type I and II), also among them are IBM microdrives. Use the CF card with item no. 758-879; it is designed for a high-temperature range and can guarantee data integrity.

3.1.3.2 Fieldbus Connections (X3)

Depending on the I/O-IPC version, there is a fieldbus connection for either PROFIBUS or CANopen.
An interface illustrations are provided in the corresponding section of the particular fieldbus (starting with section 3.2).

3.1.3.3 Power Supply Connection (X4)

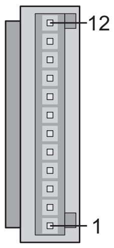
Use the X4 connection for the I/O-IPC power supply. The power supply unit provides the required voltage to the electronics and the I/O modules.



SIGNAL:	PIN:
V_EXT	1
GNDEXT	2
Shield	3

3.1.3.4 OnBoard IOs (X5)

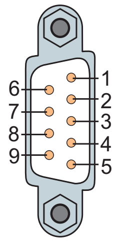
This interface provides two integrated digital inputs and two outputs. These outputs connect the sensors and actuators that need to be supplied with power independent of the internal bus.
The following table and illustration provide information on the pin assignment of the interface:



SIGNAL:	PIN:
DIN0	1
~DIN0	2
DIN1	3
~DIN1	4
DOUT0	5
~DOUT0	6
DOUT1	7
~DOUT1	8
WDOG	9
REL_NC	10
REL_NO	11
SHIELD	12

3.1.3.5 Serial Interface (X6)

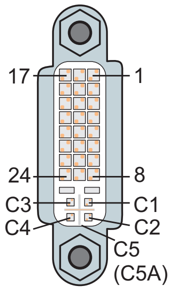
The following table and illustration provide information on the pin assignment of the interface:



SIGNAL:	PIN:
DCD1	1
RXD1	2
TXD1	3
DTR1	4
GND	5
DSR1	6
RTS1	7
CTS1	8
RI1	9

3.1.3.6 CRT/DVI Interface (X7)

The following table and illustration provide information on the pin assignment of the analog/digital interface (DVI-I):



SIGNAL:	PIN:	SIGNAL:	PIN:	SIGNAL:	PIN:
TXD2-	1	TXD1-	9	TXD0-	17
TXD2+	2	TXD1+	10	TXD0+	18
GND	3	GND	11	GND	19
NC	4	NC	12	NC	20
NC	5	NC	13	NC	21
DDCCLK	6	VCC_DVI	14	GND	22
DDCDATA	7	GND	15	TXCP	23
CRT_VSY	8	NC	16	TXCN	24
CRT_R	C1	CRT_G	C2	CRT_B	C3
CRT_HSY	C4	GND	C5	GND	C5A

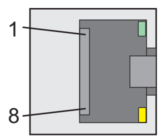
3.1.3.7 Ethernet Interface (X8, X9)

Two 10/100BASE-TX Ethernet interfaces are available to connect to the network. Two LEDs indicate the operating status: the LINK LED becomes green when a connection is established, the ACT LED blinks yellow during data communication.

An IP address can only be requested from the BootP server via X9 Ethernet interface. For this reason, use X9 first. After an IP address has been assigned to Erthernet interface X9, it is possible to configure X8 with the web server.

Note:
A bandwidth of 100 Mbit for the data transmission via Ethernet interface X9 can only be guaranteed if the cable length does not exceed 30 m. If the cable is longer, data transmission will be limited, or may not occur at all.
It is possible to use a 100 m-long cable with a bandwidth of 10 Mbit.

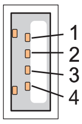
The following table and illustration provide information on the pin assignment of the interface.



SIGNAL:	PIN:
TX+	1
TX-	2
RX+	3
RXC/CMT	4
RXC/CMT	5
RX-	6
RXC/CMT	7
RXC/CMT	8

3.1.3.8 USB Interface (X10, X11)

The following table and illustration provide information on the pin assignment of the interfaces. The interface connection adheres to the USB specification 1.1.



SIGNAL:	PIN:
USB_VCC1	1
USB_N1	2
USB_P1	3
USB_GND	4

3.1.4 Description of the Indicators

The operational status of the I/O-IPC or of the node is indicated via nine LEDs.

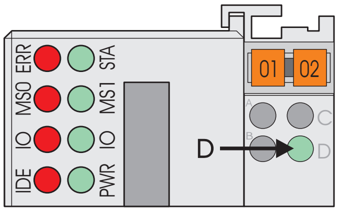


Fig. 3-3: Indicators g287002x

LED:	Color:	Indicates:
ERR	Red	Fieldbus status indication (depends on the fieldbus)
STA	Green	
MS0	Red	Module status indication (depends on the application)
MS1	Green	
IO	Red	Internal bus status indication. Blinks after start-up. Internal bus is defective. Also see section 3.1.15.6 blink code.
IO	Green	Blinks after start-up. Internal bus is operational.
IDE	Red	Activity status (IDE) of the CF card and power supply (PWR). Initially controlled by the hardware; after start-up, controllable by application software.
PWR	Green	
D	Green	Indication of the power supply status for the power jumper contacts. When using the supply module 750-602, the LED is always off.

For more details on all functions see section 3.1.15.

3.1.5 Description of the Operating Elements

3.1.5.1 Operating Mode Switch

The operating mode switch has the following functions:

Position:	Function:
Run	If the switch is in this position, start the PLC program with CoDeSys. A saved program will also start automatically when the power supply is turned off and on again.
Stop	If the switch is in this position, the PLC program is stopped.

The following table illustrates the program flow when using the “Run/Stop” operating mode switch in connection with the (Run/Stop) option in CoDeSys:

Position of the “Run/Stop” switch at the I/O-IPC	Option “Start/Stop” in CoDeSys	Program flow
Run	Run	Run
Stop	Run	Stop
Run	Stop	Stop
Stop	Stop	Stop

APIs are available for an IPC without WAGO-I/O-*PRO* CAA in order to read the status of the operating mode switch.

3.1.5.2 Reset Button

Press the **“Reset”** button (RST) to restart the I/O-IPC. All values, except for those stored in the SRAM, are reset to default values. For safety reasons, the **“Reset”** button can only be reset with a pointed object; e.g., a ballpoint pen. This design feature ensures that the **“Reset”** button is not pressed unintentionally.

3.1.5.3 Hardware Address (MAC ID)

Every I/O-IPC has two 10/100BASE-TX Ethernet interfaces. Every interface is provided with a unique and internationally unambiguous physical Ethernet address, also referred to as the MAC-ID “Media Access Control Identity.” The ID is located on a label on both the bottom and side surfaces of the I/O-IPC. The address has a fixed length of 6 bytes (48 bits) and contains the address type, the manufacturer’s ID and the serial number.

3.1.6 Description of the PLC Program Processing

After either a power-on or after a reset, the operating system of the I/O-IPC starts. After the initialization of the I/O-IPC and of the internal bus, the PLC program (if available) is loaded. The PLC program is WAGO-I/O-*PRO* CAA. If a fieldbus interface is available, it is also initialized. See also illustration on the following page.

If the “Run/Stop” switch is in the “Run” position, the PLC program starts. During the start-up, phase both the red and the green “I/O” LEDs blink. After an error-free system start-up, the green “I/O” LED is on (the red “I/O” LED is off).

PLC Cycle

After an error-free system start-up, the application begins. The PLC runtime system automatically starts to execute the PLC program if the operating mode switch is in the “Run” position. If the operating mode switch is in the “Stop” position, the PLC program does not start.

If a PLC program is executed in the I/O-IPC, a PLC cycle starts by: reading the fieldbus data, the data of the integrated inputs and outputs, the data of the I/O modules and the time data. Afterward, a PLC cycle is executed in the RAM.
After a program cycle has been completed, the fieldbus data, the data of the digital inputs, outputs and I/O modules is updated with new output data. After this, system functions are executed (system diagnostics, communication, time calculations, etc.). If no stop command was received (**“Reset”** button), a new cycle is started by reading the fieldbus data, I/O module data and time data.
The operating mode (“Stop/Run”) is only changed at the end of a PLC cycle.
The cycle time is the time from the start of the PLC program to the next start. If a larger loop is programmed in the PLC program, the cycle time will be longer.

The inputs and outputs are not updated while the program is executed. Updates are only made when a PLC program cycle has been completed. It is therefore not possible to receive a change of physical I/Os when the program loop is executed.

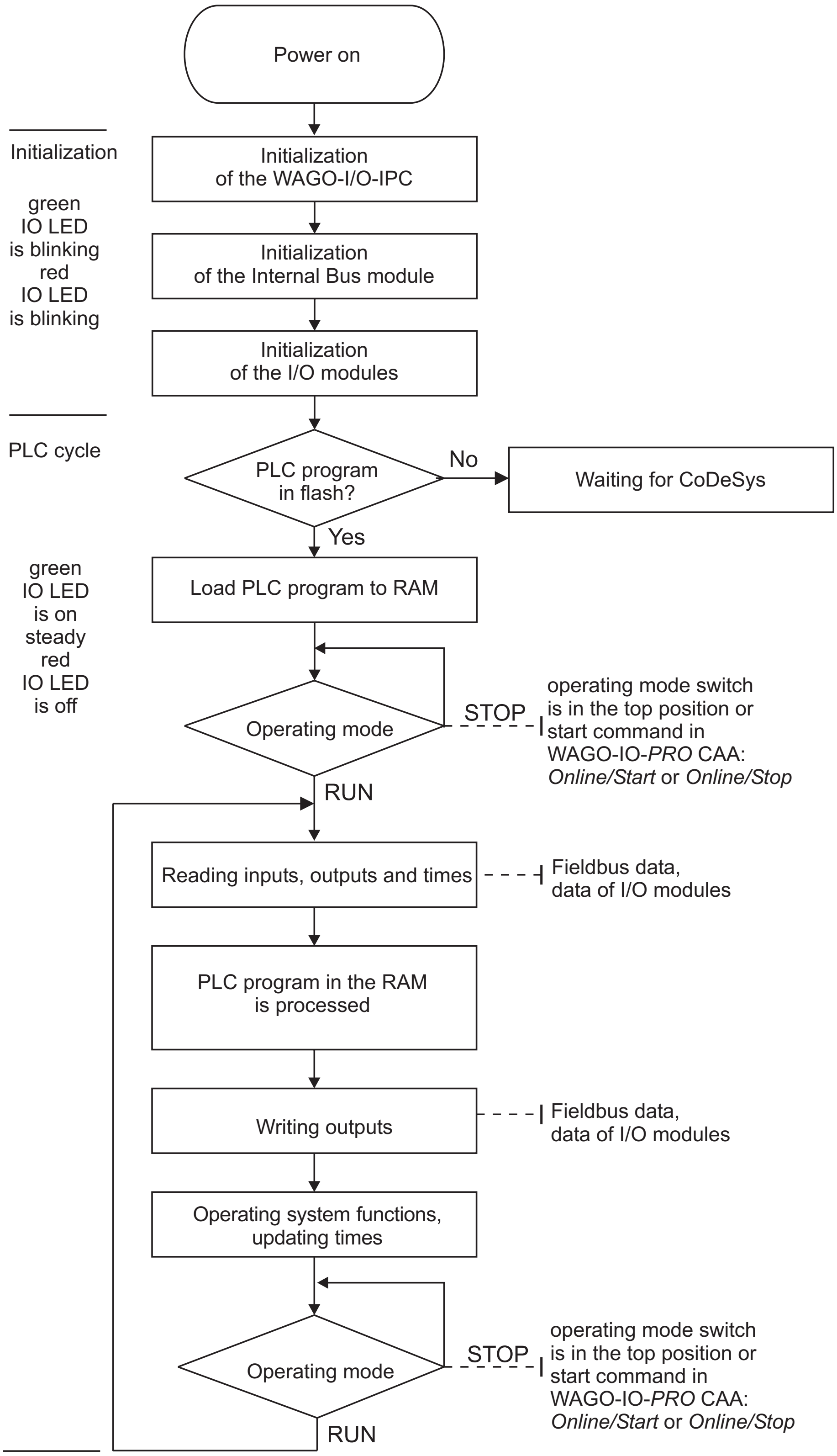


Fig. 3-4: I/O-IPC operating system

g287003d

3.1.7 Linux Operating System

This section provides information on the Linux operating system. However, basic knowledge of the Linux operating system is required.

3.1.7.1 Linux File System

In the operational mode, the I/O-IPC is booted from the flash memory.

The flash memory (Linux device/dev/hda) is partitioned as follows:

Linux partition name:	Size:	Type:	Note:
/dev/hda1	1 MB	FAT	Boot partition
/dev/hda2	19 MB	Linux EXT2	Linux root file system
/dev/hda3	All available space	Linux EXT2	Data memory

Partition sizes for the 32 MB Version

The memory allocation of the four partitions that is given in the tables is only an approximation. The specified values only refer to the default status of the I/O-IPC and may change during operation (e.g., during a firmware update).

In order to see the free disk space on the partitions, use the shell command `df` (disc free).

Linux partition name:	Memory:		File path:
	Total	Used (approx.)	
/root	18525 kB	12282 kB	/
/dev/hda1	1068 kB	968 kB	/boot
/dev/hda3	10593 kB	15 kB	/data
/dev/rd/0	3963 kB	25 kB	/tmp

Partition sizes for the 128 MB Version

Linux partition name:	Memory:		File path:
	Total	Used (approx.)	
/root	72871	12282	/
/dev/hda1	1260	968	/boot
/dev/hda3	47584	4130	/data
/dev/rd/0	3963	27	/tmp

CoDeSys programs including all resources such as bitmaps etc. are transferred to the partition */dev/hda3* or in the directory */data* via the CoDeSys gateway.

If a program download failed, ensure that there is enough free disk space available.

3.1.7.1.1 Boot Partition */dev/hda1*

The boot partition includes:

- the Linux kernel in the bzImage format; configured for I/O-IPC hardware,
- the boot loader ldlinux.sys
- and all configuration files that can be modified using web-based management tools or local configuration tools.

This partition is write protected.

3.1.7.1.2 Root Partition */root*

The root partition includes:

- all files of the root file system
- all available Linux drivers and applications

This partition is write protected.

3.1.7.1.3 User Data Partition */dev/hda3*

This partition is the memory location for files that are generated by CoDeSys and for user data files that can be transferred via FTP.

The partition has read/write access. When something has been written to the partition, a `sync` command is executed in order to guarantee data integrity even after an unexpected shutdown.

Note:
The life of the flash memory is limited and depends on the number of write cycles.

The precise length of a flash memory life cannot be calculated. In order to increase the life of the flash memory, the data is distributed evenly over the entire flash during a write cycle.

3.1.7.1.4 External Mass Storage Devices

The operating system supports external mass storage devices such as USB and CD-ROM drives, USB storage devices and CF cards. Connect such devices using the `mount` command.
As an alternative, these devices can be used to run the I/O-IPC or an application in order to start a special configuration, diagnostics or updates.

3.1.8 Logical Address Spaces of the Inputs and Outputs

Note:
This section provides a description of software structures that are only relevant within the PLC software WAGO-I/O-*PRO* CAA and only if the Modbus protocol is used.

3.1.8.1 Introduction

The process image of the I/O-IPC consists of several areas. These include:

- input data to the local I/O modules at the internal bus
- output data from the local I/O modules at the internal bus
- input variables to the PLC
- output variables from the PLC
- input data to the optionally available fieldbus systems (PROFIBUS, CANopen)
- output data from the optionally available fieldbus systems (PROFIBUS, CANopen).

Every area is assigned an address space in order to access the area in a PLC project using WAGO-I/O-*PRO* CAA.

For Modbus access, this data is transferred to a special Modbus address space. In some cases the same data can be available in different address spaces.

3.1.8.2 Process Interface of the I/O Modules

The input process image (PII) and the output process image (PIO) transfer input and output data both from and to the I/O modules.

Note:
Depending on the specific address of an I/O module in the fieldbus node, the process data of all previous byte-oriented or bit-oriented I/O modules must be considered in determining the position of the I/O module in the process image.

For detailed information on the assignment of I/O modules to logical addresses, please refer to section 3.1.9.3.

3.1.8.3 Data Access within a PLC Program

The method to access process data that is used by the PLC functions is independent of the data source (fieldbus etc.) for all I/O-IPCs. The data is always accessed by an application in accordance with IEC 61131-3. This means that the process data is read at the beginning of the PLC task cycle that processes the data. At the end of the PLC task cycle, the process data is written. While the PLC task is being processed, changes in the process data are not taken into consideration.

3.1.8.4 Data Access via Modbus/TCP (UDP)

Remote access to the entire process image of the IPC is possible via Modbus. Access via Modbus allows the user to write and to read elements of the process image.

Note:
Depending on the specific address of an I/O module in the fieldbus node, the process data of all previous byte-oriented or bit-oriented I/O modules must be considered in determining the position of the I/O module in the process image.

3.1.9 Data Exchange

The I/O-IPC exchanges process data with its environment during operation. The process data is transmitted by the following interfaces:

- Ethernet (Modbus slave, HTTP, CoDeSys web visualization)
- Fieldbus (PROFIBUS, CANopen)
- Internal bus and connected I/O modules.

The I/O-IPC supports a defined number of simultaneous socket connections from different network devices:

- up to 3 connections for HTTP (read out HTML pages of the IPC; e.g., from the PC)
- up to 10 connections via Modbus/TCP (read or write input or output data of the IPC from the PC)
- up to 2 connections for WAGO-I/O-*PRO* CAA (these connections are reserved for application programming via ETHERNET). However, only one programming tool can access the I/O-IPC.

The maximum number of simultaneous connections cannot be exceeded. Additional connections cannot be set up.

3.1.9.1 Memory Areas of the Process Image

The I/O-IPC uses both the Modbus and the PLC functionality to exchange data (see the following illustration).

Data exchange is possible between the Modbus data, PLC data, fieldbus data, and internal bus data.

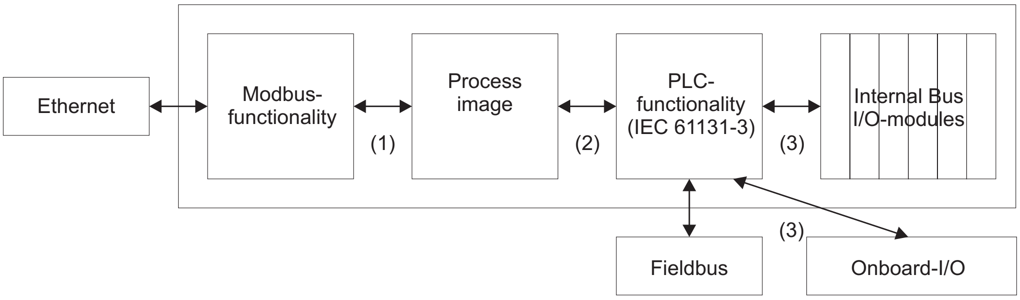


Fig. 3-5: Memory areas and data exchange of the I/O-IPC g287005d

The process image of the I/O-IPC contains data for the I/O modules at the internal bus, of the fieldbus, of the onboard IOs and also the PLC variables. Additionally, there is also the PLC flag data and the residual data (retain).

(1) PLC functionality, as well as the Modbus functionality, allow for reading process image data (see illustration above).
(2) Likewise, the PLC functionality, as well as the Modbus functionality, allow for writing data into the process image.
(3) Reading and writing of internal bus data, fieldbus data and data of the onboard IOs is only supported by the PLC functionality.

Note:
Due to architectural reasons, the use of a PLC program is essential in the 758-870 for proper operation of the Modbus services.
If there is no PLC program, all Modbus service responses are sent to the memory areas PFC-IN, PFC-OUT and the response for flags is the Modbus exception code “ILLEGAL_RESPONSE_LENGTH.”
Only variables that were declared with the “AT %M<Address>” statement can be accessed via Modbus services. Variables that were declared as VAR RETAIN cannot be accessed via Modbus services.

The following table illustrates the allocation of access writes to the physical memory areas:

Abbreviation:	Memory area/ Description:	Access via Modbus:	Access via PLC:	Logical address space for PLC programming:
PII	Input process image Process image of the local input modules (internal bus, I/O modules 1 to 64).	Read	Read	**Word** %IW 0 to %IW255 **Byte** %IB 0 to %IB511
PIO	Output process image Process image of the local output modules (internal bus, I/O modules 1 to 64).	Write	Read/ Write	**Word** %QW 0 to %QW255 **Byte** %QB 0 to %QB511
PLC PII	PLC/Modbus Input process image Process image of the PLC input variables that can be accessed via Modbus.	Read/ Write	Read	**Word** %IW256 to %IW511 **Byte** %IB 512 to %IB 1023
PLC PIO	PLC/Modbus Output process image Image of the PLC output variables that can be accessed remotely via Modbus.	Read	Read/ Write	**Word** %QW256 to %QW511 **Byte** %QB 512 to %QB 1023
OnBrdI	Integrated digital input Image of the digital I/O-IPC input bits (0 & 1) that can be accessed by the user.	Read	Read	**Bit** %IX 2300.0 to %IX 2300.1
OnBrdO	Integrated digital output Image of the digital I/O-IPC output bits (0 & 1) that can be accessed by the user.	Read/ Write	Read/ Write	**Bit** %QX 2300.0 to %QX 2300.1

Abbreviation:	Memory area/ Description:	Access via Modbus:	Access via PLC:	Logical address space:
FBusI	Fieldbus input Image of the input area for the implemented fieldbuses.	Read	Read	**Word** %IW2400 to %IW31767 **Byte** %IB4800 to %IB65535
FBusO	Fieldbus output Image of the output area for the implemented fieldbuses.	Write	Write/ Read	**Word** %QW2400 to %QW31767 **Byte** %QB4800 to %QB65535
Flag variables	8 kB residual memory in the SRAM Declared with "AT %M<Address>".	Read/ Write	Read/ Write	**Word** %MW0 to %MW 4095 **Byte** %MB0 to %MB8190
Retain variables	N/A	N/A	Read/ Write	Symbolically addressable residual memory in the SRAM 120 kBytes

Use bit-oriented addressing and observe that the base address is word-oriented. The bits are in the areas 0 to 15.

Note:
The fieldbus master requires the areas %MB0 to %MB141 in order to file the diagnostic information. Do not write data into this area as it will be overwritten! Only use the memory %MB0 142 and above for flag variables.

3.1.9.2 Variable Declaration in the Flag Memory

If the program variables need to be saved when the IPC power is turned off (stored in the non-volatile memory), use residual variables. Among these are retain variables and persistent variables. All other types of variables are re-initialized when the IPC is rebooted.

RETAIN
Retain variables are represented by the key word RETAIN. These variables keep their values after an abnormal termination, as well as after switching the control on or off regularly.
The runtime system places the variable in the memory.

AT% M<Address>
If a variable is declared with the key phrase AT%M <Address>, the variable is stored in the non-volatile memory.

PERSISTENT
Persistent variables are represented by the code word PERSISTENT. They keep their values after another download. However, in contrast to retain variables, they do not keep their values after a reset or reboot because they are not stored in the retain memory. If persistent variables are to keep their values after a control failure, they must also be declared with the code word VAR RETAIN.

For more detailed information please consult the WAGO-I/O-*PRO* CAA “Help” file.

Examples for permanent storage of DWORD test variables:

```
VAR RETAIN
    Test: DWORD;
END_VAR
```

```
VAR
    Test AT%MW1:
DWORD;
END_VAR
```

```
VAR RETAIN PERSIST
    Test: DWORD;
END_VAR
```

- If a local variable has been declared with RETAIN in a program, this variable will be stored in the residual memory (like a global Retain variable).
- If a local variable in a function block has been declared with RETAIN, the entire instance of this function block is stored in the residual memory (all data of the module); however, only the declared retain variable is treated as such.

3.1.9.3 Process Image

After the power is applied to the IPC, it automatically identifies all I/O modules of the node (data width/bit width > 0). The node can consist of a mixed arrangement of a maximum of 63 analog and digital I/O modules.

Note:
For the number of input and output bits or words of individual modules please refer to the relevant I/O module descriptions starting with section 5.2.1.

The I/O-IPC creates an internal local process image on the basis of the data width, the type of I/O module and the module's position in the node. The local process image is divided into two data zones containing the data received and the data to be sent. The analog I/O modules, and most specialty modules, are loaded into the process image before the digital modules are loaded.

Every I/O module is assigned a place in the process image. The data of the analog modules is sent to the I/O-IPC byte by byte, according to the order in which the modules are connected. After that, the bits of the digital I/O modules are grouped into words and also loaded into the process image. If the amount of digital information exceeds 16 bits, the I/O-IPC automatically starts with a new word.

Note:
Changing a node's physical layout will result in a new process image structure. The process data addresses (data of the inputs and outputs) also change. If I/O modules are added or removed, it is necessary to verify the entire process data.

The process image for the input and output modules is stored in the words 0 to 255. The memory is divided into separate areas for input data and output data. However, in a PLC program both areas are represented with an index of 0 to 255. See also the areas PIO and PII (see section 3.1.9.1).

3.1.9.3.1 Example for an Input Process Image

The following illustration is an example for an input process image. The configuration includes 16 digital and 8 analog inputs. Therefore, the data length of the process image is 9 words (8 words for the analog data and 1 word for the digital inputs).

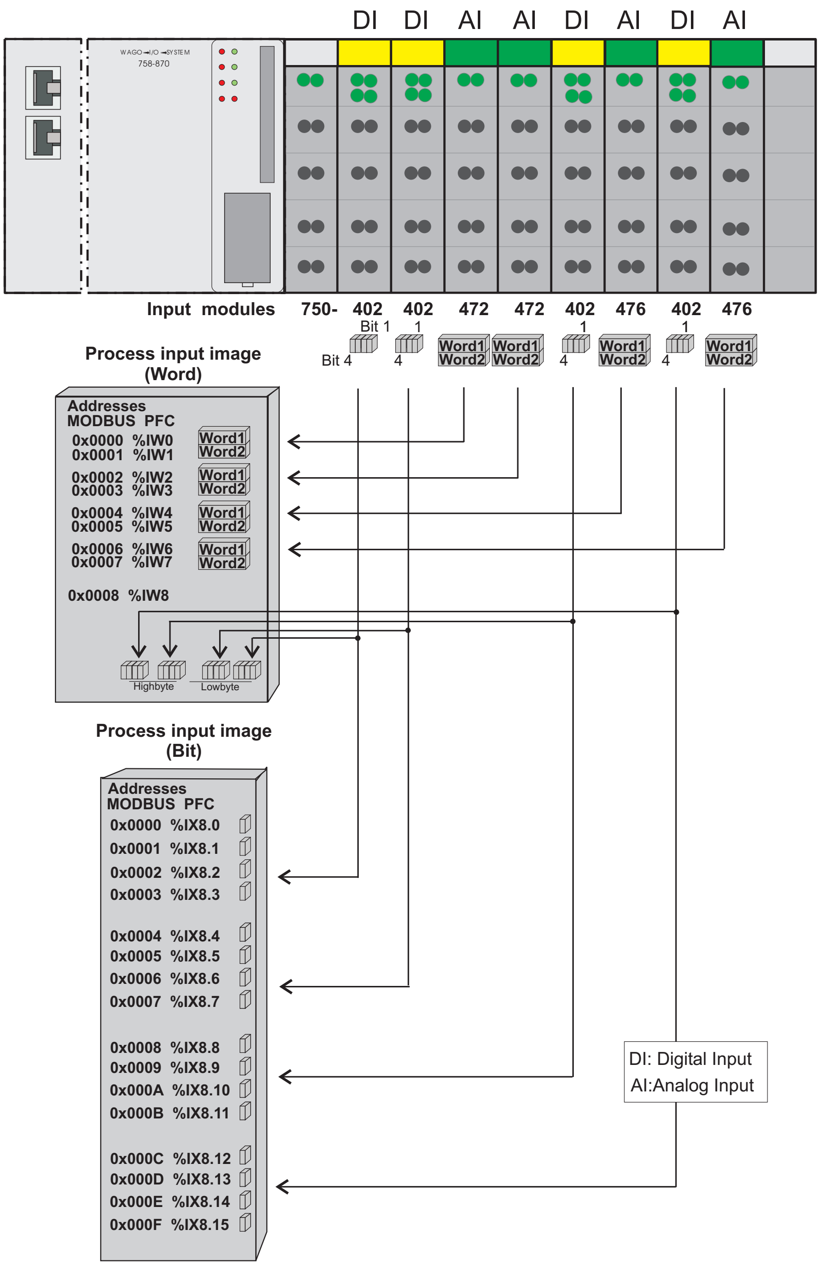


Fig. 3-6: Example for an input process image g287020d

3.1.9.3.2 Example for an Output Process Image

The following illustration is an example for an output process image. The configuration includes two digital and four analog outputs. Therefore, the data length of the process image is five words (four words for the analog data and one word for the digital outputs).

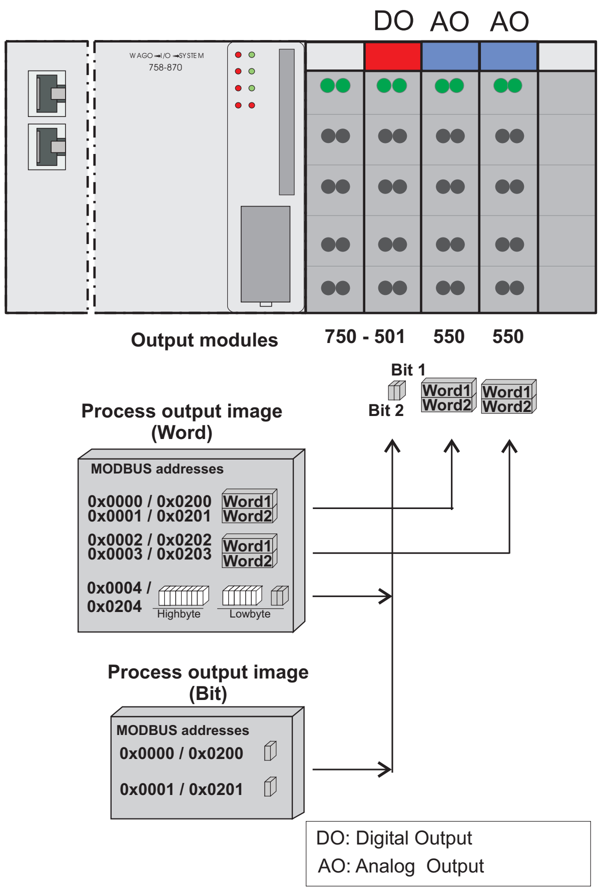


Fig. 3-7: Example for an output process image g287021d

3.1.9.3.3 Example with Different I/O Modules

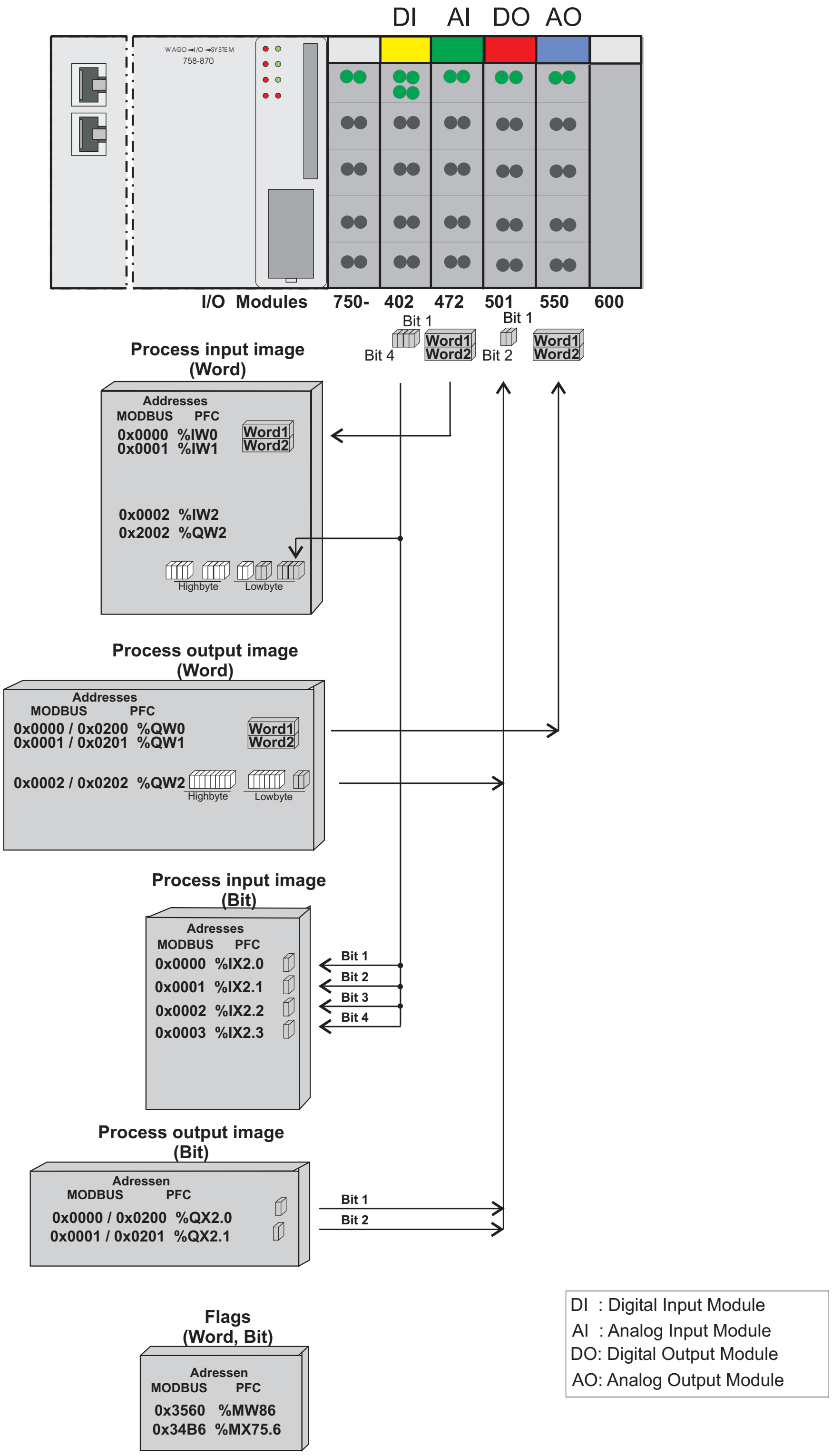


Fig. 3-8: Example: Addressing a fieldbus node g287022d

3.1.9.4 Addressing Data in the Process Image

3.1.9.4.1 Addressing I/O Module Data

When addressing I/O modules, the I/O-IPC transmits the data of the complex modules first (I/O modules that occupy one or more bytes). This data is sent to the I/O-IPC according to the order in which the modules are connected. Therefore, this data occupies the addresses starting with word 0. Then, the bits of the digital I/O modules are grouped into words (16 bits per word) and transmitted according to the order in which the modules are connected. If the amount of digital information exceeds 16 bits, the I/O-IPC automatically starts with a new word.

For detailed information on the number of input and output bits or bytes of a special I/O module, please refer to section 5.2.1.

Note:
If a node is changed, this results in a new process image structure. The process data addresses also change. If I/O modules are added or removed it is necessary to verify the entire process data.

Please find below an overview of the data widths for different I/O modules:

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 thermocouples	Digital output modules with diagnostics (2 bits/channel)
Input modules for resistor sensors	Supply modules with fuse carrier/diagnostics
Pulse width output modules	Solid state power relay
Interface module	Relay output module
Up/down counters	-
I/O modules for angle and distance measurement	-

3.1.9.4.2 Prefix Description for the Syntax of Logical Addresses

Special prefixes allow to access individual memory elements in accordance with IEC 61131-3. The beginning of these prefixes are marked by the percent sign (%). The following is a description of the signs:

Position:	Prefix:	Marking:	Comments:
1	%	Starts the absolute address	-
2	I	Input	
	Q	Output	
	M	Flag	
3	X*	Single bit	Data size
	B	Byte (8 bits)	
	W	Word (16 bits)	
	D	Double-word (32 bits)	
4		Address	
* The "X" character is optional when declaring bit variables.			

Two examples:

Addressing word by word %QW27 (28th word)

Addressing bit by bit %IX1.9 (10th bit in word 2)

Note:
Make sure there are no blank spaces in the character string for the absolute address!

3.1.10 Comparison of Modbus and IEC-61131-3 Addressing

Please find detailed information on the Modbus communication in the manual "Modbus Communication between WAGO Ethernet Couplers and Controllers." It is available at www.wago.us under: *Service → Documentation → Application Notes → Total Overview → A300003*.

3.1.10.1 Accessing the Process Image via Modbus Function Codes

The process image address allocation via Modbus cannot be directly applied to the IEC-61131 addresses. This is due to the fact that the access to an address is defined by the Modbus function code. When addressing in accordance with IEC-61131, the access is defined by the IEC-61331 address standard.

The table describes the access types used to access logical address spaces.

FC:	Name:	Description:
FC1	Read coils	Read several digital outputs
FC2	Read inputs discrete	Read several digital inputs
FC3	Read holding registers	Read several analog inputs and outputs
FC4	Read input registers	Read several analog inputs and outputs
FC5	Write coil	Write a single digital output
FC6	Write single register	Write a single analog output
FC11	Get communication event counter	Communication event counter
FC15	Force multiple coils	Write several digital outputs
FC16	Write multiple registers	Write several analog outputs
FC23	Read/write multiple registers	Read and write operation on analog inputs and outputs

3.1.10.2 Register Services of 758-870

Determine, or change, the state of analog and digital I/O modules with the register services. There are certain function codes available for every function.

Read registers with FC3, FC4 and FC23

Modbus addresses: [dec]	Modbus addresses: [hex]	IEC 61131-3-addresses:	Description:
0–255	0x0000 –0x00FF	%IW0–%IW255	Physical memory of the inputs 256 words of input data
256–511	0x0100 –0x01FF	%QW256–%QW511	Physical memory of the outputs Volatile PLC output variables
512–767	0x0200 –0x02FF	-	Modbus exception (fault): “Illegal data address”
768–1023	0x0300 –0x03FF	%IW256–%IW511	PLC memory of the inputs Volatile PLC input variables
1024–4095	0x0400 –0x0FFF	-	Modbus exception (fault): “Illegal data address”
4096–12287	0x1000 –0x2FFF	-	Modbus configuration register (see section 3.1.10.4)
12288–16384	0x3000 –0x4000	%MW0–%MW4095	Retain memory (8 kB) Non-volatile PLC variables
16385–65535	0x4001 –0xFFFF	-	Modbus exception (fault): “Illegal data address”

Write registers with FC6, FC16 and FC23

Modbus addresses: [dec]	Modbus addresses: [hex]	IEC 61131-3-Addresses:	Description:
0–255	0x0000 –0x00FF	%QW0–%QW255	Physical memory of the outputs 256 words of output data
256–511	0x0100 –0x01FF	%IW256–%IW511	PLC memory of the inputs Volatile PLC input variables
512–767	0x0200 –0x02FF	-	Modbus exception (fault): “Illegal data address”
768–1023	0x0300 –0x03FF	%IW256–%IW511	PLC memory of the inputs Volatile PLC input variables
1024–4095	0x0400 –0x0FFF	-	Modbus exception (fault): “Illegal data address”
4096–12287	0x1000 –0x2FFF	-	Modbus configuration register (see section 3.1.10.4)
12288–16384 Or can be set to max. 24576	0x3000 –0x3FFF 0x5FFF	%MW0–%MW4095 %MW0–%MW12287	Retain memory (8 kB) Non-volatile PLC variables Retain memory (24 kB) Non-volatile PLC variables
16385–65535	0x4001 –0xFFFF	-	Modbus exception (fault): “Illegal data address”

3.1.10.3 Bit Services of 758-870

The digital bit function codes can only determine or change the state of digital I/O modules. Analog modules are ignored or cannot be accessed.

Read coils with FC1 and FC2

Modbus addresses:		IEC 61131-3-Addresses:	Description:
[dec]	[hex]		
0–511	0x0000 –0x01FF	Memory area %IX 0.0–%IX 32.15 + Offset value	Input process image The bit-based Modbus addressing starts with the first digital I/O module on the internal bus. If analog modules are used, the address space that is occupied by these modules is ignored (offset value).
512–1023	0x0200 –0x03FF	-	Modbus exception (fault): “Illegal data address”
1024–1025	0x0400 –0x0401	%IX2300.0–%IX2300.1	Integrated digital inputs
1026–4095	0x0402 –0x0FFF	-	Modbus exception (fault): “Illegal data address”
4096–8191	0x1000 –0x1FFF	%QX256.0–%QX511.15	PLC memory of the outputs
8192–12287	0x2000 –0x2FFF	%IX256.0–%IX511.15	PLC memory of the inputs Volatile PLC input variables
12288–32767	0x3000 –0x7FFF	%MX0.0–%MX1279.15	Retain memory (8 kB)
32768–65535	0x8000 –0xFFFF	-	Modbus exception (fault): “Illegal data address”

Write Coils with FC5 and FC15

Modbus addresses: [dec]	[hex]	IEC 61131-3-Addresses:	Description:
0–511	0x0000 –0x01FF	Memory area %QX 0.0–%QX 32.15 + Offset value	512 bits of digital output data. The bit-based Modbus addressing starts with the first digital I/O module on the internal bus. If analog modules are used, the address space that is occupied by these modules is ignored (offset value).
512–1023	0x0200 –0x03FF	Memory area %QX 256–%QX511 + Offset value	Mirrored memory for read/write access. 512 bits of digital output data. The bit-based Modbus addressing starts with the first digital I/O module on the internal bus. If analog modules are used, the address space that is occupied by these modules is missed (offset value).
1024–1025	0x0400 –0x0401	%QX2300.0– %QX2003.1	Integrated digital outputs
1026–4095	0x0402 –0x0FFF	-	Modbus exception (fault): “Illegal data address”
4096–8191	0x1000 –0x1FFF	%IX256.0–%IX511.15	PLC memory of the inputs Volatile PLC input variables
8192– 12287	0x2000 –0x2FFF	%IX256.0–%IX511.15	PLC memory of the inputs Volatile PLC input variables
12288– 32767	0x3000 –0x7FFF	%MX0.0–%MX1279.15	Retain memory (8 kB) Non-volatile PLC variables
32768– 65535	0x8000 –0xFFFF	-	Modbus exception (fault): “Illegal data address”

3.1.10.4 Modbus Configuration Register

The Modbus configuration registers for **FC3, FC4, FC6 and FC16** can adjust the behavior of the IPC.

Modbus address:		Length:	Access:	Description:
[dec]	[hex]	[Word]		
4128	0x1020	1	R	LED error code
4129	0x1021	1	R	LED error argument
4130	0x1022	1	R	Number of analog outputs in the process image [Bit]
4131	0x1023	1	R	Number of analog inputs in the process image [Bit]
4132	0x1024	1	R	Number of digital outputs in the process image [Bit]
4133	0x1025	1	R	Number of digital inputs in the process image [Bit]
4136	0x1028	1	R/W	IP configuration: BootP (1), DHCP (2) or FIX (4)
4138	0x1030	1	R/W	Enable Modbus connection watchdog
4139	0x1031	3	R	ETH Interface X9: MAC ID
4140	0x1032	3	R	ETH Interface X8: MAC ID
8192	0x2000	1	R	0x0000 (constant)
8193	0x2001	1	R	0xFFFF (constant)
8194	0x2002	1	R	0x1234 (constant)
8195	0x2003	1	R	0xAAAA (constant)
8196	0x2004	1	R	0x5555 (constant)
8197	0x2005	1	R	0x7FFF (constant)
8198	0x2006	1	R	0x8000 (constant)
8199	0x2007	1	R	0x3FFF (constant)
8200	0x2008	1	R	0x4000 (constant)
8213	0x2015	1	R	Modbus server version
8256	0x2040	1	W	Software reset (write 0x55AA or 0xAA55)
8260	0x2044	1	W	Delete Modbus configuration file (write 0x55AA)

3.1.11 I/O-IPC Start-Up

Assigning the I/O-IPC an IP address in the network allows communication with the I/O-IPC. The following two options are available:

- connect a monitor and a USB keyboard to the I/O-IPC and configure it using the local configuration program/sbin/ipconfig.
- configuration of the I/O-IPC without a monitor. In this case, it is necessary to install a BootP server on a local Windows or Linux host.

3.1.11.1 IP Address Assignment

Attention:
The following description serves as an example and only applies to the local start-up of a single Ethernet fieldbus node and a Windows-based computer.
A direct Internet connection should be exclusively established by an authorized network administrator and is therefore not described in this manual.

To assign an IP address via BootP server, complete the following:

1. Write down MAC-ID and set up fieldbus node.
2. Connect the PC and the fieldbus node.
3. Determine the IP address of the host PC under Windows.
4. Assign the IP address to the fieldbus node via BootP server.
5. Allocate address using the WAGO-BootP server.
6. Execute an I/O-IPC network test.
7. Use the DHCP or a fixed IP address.

The following sections provide detailed information on these steps.

3.1.11.1.1 Write Down MAC-ID and Set Up Fieldbus Node

Before setting up the fieldbus node, write down the MAC-ID of the I/O-IPC Ethernet interface (X9). The ID is on a sticker, which is located on both the bottom and on the side of the I/O-IPC.

MAC-ID of the I/O-IPC:

----- ----- ----- ----- ----- -----

3.1.11.1.2 Connect the PC and the fieldbus node

1. Connect the I/O-IPC to a hub with an Ethernet cable type S-UTP, Cat.5e or directly to the PC with a crossover cable. The transmission speed of the I/O-IPC is 10 Mbit or 100 Mbit.

2. Connect the I/O-IPC to the power supply (DC 24 V, power supply unit). As soon as the operating voltage is applied, the initialization starts.

3. When the green MS1 LED blinks, the I/O-IPC is operational.

3.1.11.1.3 Determine the IP Address of the Host PC under Windows

If the PC has already been connected to the Ethernet network, proceed as follows in order to determine the IP address of the Windows PC:

1. Click on the Windows “Start” button and select “Execute.”

2. Enter the command `cmd` and press the **“Enter”** key.
   A command window opens.

3. Enter the command `ipconfig` and press the **“Enter”** key.

4. The IP address, the subnet mask and the standard gateway, including the appropriate parameters, are displayed.

5. Write down the parameters:

   IP address: ----- . ----- . ----- . -----

   Subnet mask: ----- . ----- . ----- . -----

   Standard gateway: ----- . ----- . ----- . -----

6. Now select an IP address for the fieldbus node, it must be in the same local network as the PC.

7. Write down the selected IP address:
   IP address of the fieldbus node: ----- . ----- . ----- . -----

3.1.11.1.4 Assign the IP Address to the Fieldbus Node via the BootP Server

The IP address may also be assigned under different operating systems (e.g., under Linux) or by any other BootP server. Implement the network using a crossover cable, or two parallel cables and a hub.

Editing the BootP Table

In the BootP table, a MAC address is assigned to an IP address. With the BootP server, assign an IP address that is unique in the network to the I/O-IPC. A correctly functioning BootP server is a prerequisite under Microsoft Windows.

1. Start the BootP server by clicking on **Start** in the Windows menu bar and select *Programs* → *WAGO Software* → *WAGO BootP Server* → *WAGO BootP Server*

2. Once the server is started, click on “Edit Bootptab” on the right side of the screen. An editable file opens in the Windows Editor (bootptab.txt). This file is a database for the BootP server. The file includes an example for the IP address assignment. An example command can be found below the line “Example of entry with no gateway.”

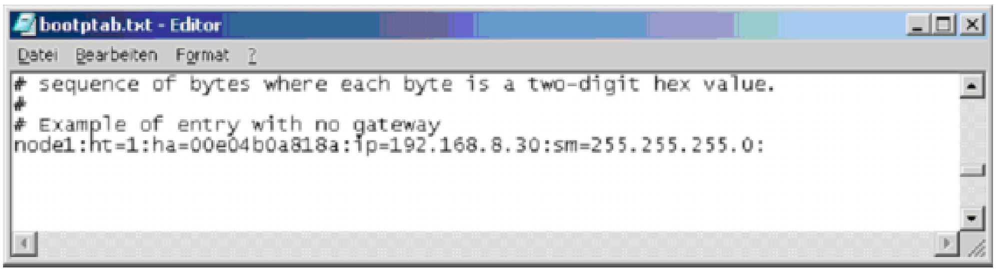


Fig. 3-9: BootP table p287001x

The illustration on the previous page includes the following information:

Parameter:	Indicates:
node1	Any name can be given to the fieldbus node.
ht=1	Enter the hardware type of the network. The hardware type for Ethernet is 1.
ha=0030DE000100 ha=0030DE000200	Enter the hardware address/the MAC ID of the I/O-IPC (hexadecimal).
ip= 10.1.254.100 ip= 10.1.254.200	Enter the IP address of the I/O-IPC (decimal).
T3=0A.01.FE.01	Enter the IP address of the gateway. Write the address in hexadecimal form.
Sm=255.255.0.0	Enter the subnet mask of the subnet (decimal) to which the I/O-IPC belongs.

The local network that is described in this example does not require a gateway.

3. Click on the line “node1:ht=1:ha=0030DE000100:ip=10.1.254.100”. Replace the twelve characters of the hardware address behind “ha=” with the MAC ID of the Ethernet interface X9.

4. If needed, assign another name to the fieldbus node by simply replacing the name “node1” with the new name.

5. In order to assign an IP address to the I/O-IPC, replace the IP address in the example behind “ip=” with the required IP address. Make sure that the three-digit numbers are separated by decimal points.

Note:
In order to address more than one fieldbus node, add one line of setup information for every additional I/O-IPC to the bootptab.txt file. Use the steps 3 to 5 as an I/O-IPC configuration guide.

6. Save the new settings in “bootptab.txt” file. To do so, select the menu item “Save” in the menu “File.”

7. Close the editor.

3.1.11.1.5 Allocate Addresses Using the WAGO-BootP Server

1. After closing the editor, click on the “**Start**” button in the open BootP dialog window. This enables the query and response mechanism of the BootP protocol. A number of messages are displayed in the BootP server message window. The error messages indicate that some services (e.g., Port 67, Port 68) have not been defined in the operating system. Ignore these error messages.

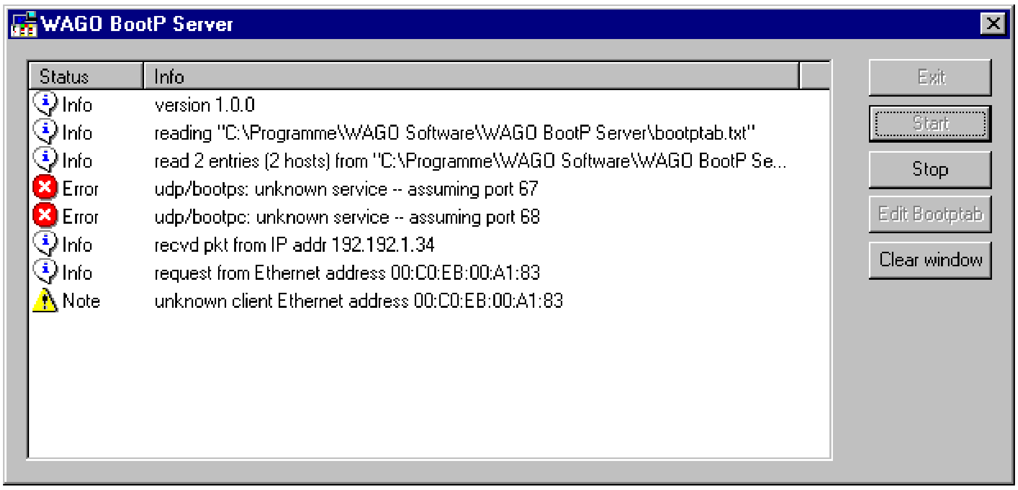


Fig. 3-10: Dialog window of the WAGO BootP server with messages p287002x

2. Restart the I/O-IPC by pressing the **“Reset”** button, or turn off the I/O-IPC supply voltage for two seconds and turn it on again.
   The I/O-IPC sends a response that the IP address has been accepted (no error). The IP address is now available in the I/O-IPC for a limited time and it is not stored permanently. If the I/O-IPC is switched off, it must be assigned with a new IP address unless there is a BootP server in the network.

3. Click on the “**Stop”** button and then on the “**Exit”** button in order to shut down the BootP server.

3.1.11.1.6 I/O-IPC Network Test

In order to test the new IP address of the I/O-IPC, open a command window.

1. Enter the `ping` command and the IP address of the I/O-IPC in the DOS window as follows: `ping` [space] xxx . xxx . xxx . xxx .
   Example: `ping 10.1.254.202`

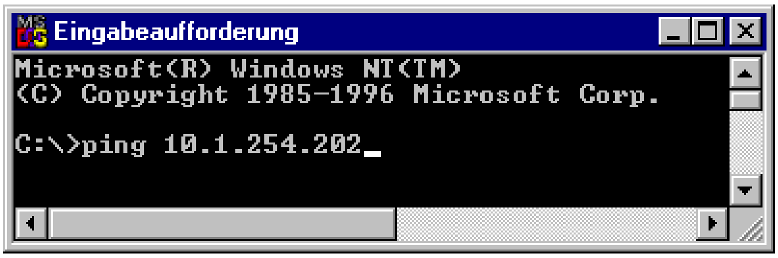


Fig. 3-11: Example for a fieldbus node function test p287003x

2. Press the **"Enter"** key. The PC receives a response from the I/O-IPC that is displayed in the DOS window. If the error message: "Timeout" appears, please compare entries with the assigned IP address and check all connections.

3. Upon successful completion, close the DOS window.

3.1.11.1.7 Using the DHCP or a Fixed IP Address

The BootP in the I/O-IPC is enabled as standard.
If BootP is enabled, the I/O-IPC expects a BootP server to be permanently present; the BootP assigns the IP address.
If the Dynamic Host Configuration Protocol (DHCP) is enabled, the I/O-IPC expects a DHCP server to be permanently present; the DHCP assigns the IP address. If the server is not available, or if the IP addresses cannot be assigned dynamically, it is possible to program the fixed IP addresses. However, if there is no BootP server available after the power has been turned on again, the network of the I/O-IPC remains disabled.

In order to operate the I/O-IPC using the IP configuration that is stored in the flash, the BootP protocol must be disabled.

Note:
If the BootP protocol is disabled after the IP addresses have been assigned, the stored IP address will exist even after the power supply of the I/O-IPC has been turned off.

Use the Web pages that are stored in the I/O-IPC to disable the BootP.

1. Open a Web browser.

2. Enter the IP address of the I/O-IPC into the address bar of the Web browser and press the **"Enter"** key.

3. The "Status information" information window opens and displays I/O-IPC information.

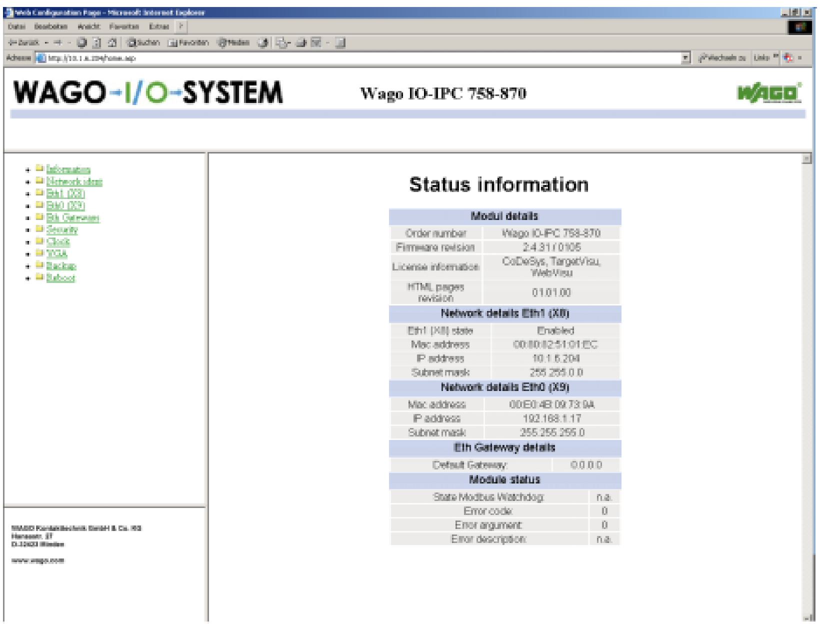


Fig. 3-12: Status information window p2870n1d

4. If a hyperlink in the left navigation bar is clicked, a password is requested. This password access protection includes three different user groups.

5. In order to log in as an administrator, enter the user name **user** and the password **user00**.

Note:
If the I/O-IPC does not display the start page, make sure that the Web browser configuration allows bypassing the proxy server for local addresses.

6. In the navigation bar, click on "Eth0 (X9)" and on "TCP/IP" in order to configure the Ethernet connection X9. The possible options to assign IP addresses that are supported by the I/O-IPC are shown.

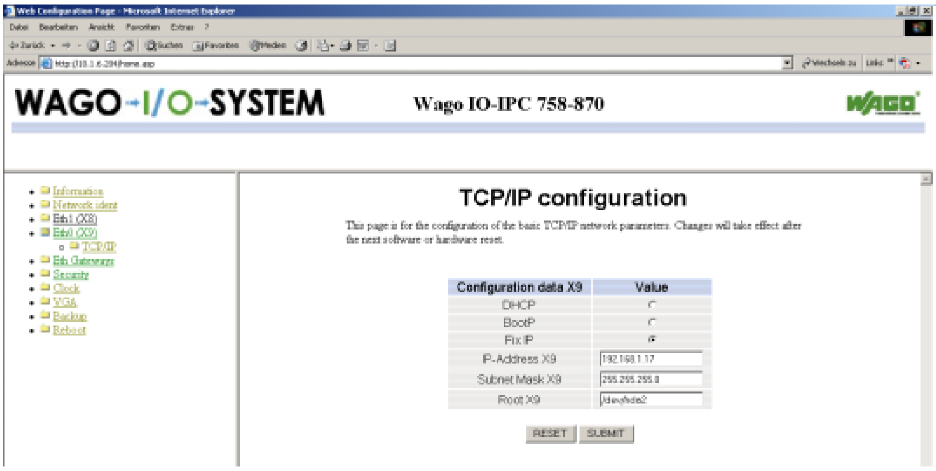


Fig. 3-13: TCP/IP configuration window Eth0 (X9) p2870n2d

7. Click on "DHCP" or "Fix IP" in order to select the desired procedure to allocate the IP address for the Ethernet interface X9. Click on the "**Submit"** button to save the selection.

8. If selecting DHCP, click on “Network ident” in the navigation bar. Enter the PC host name in the “Host name” field so that it can be addressed via its host name in the network.

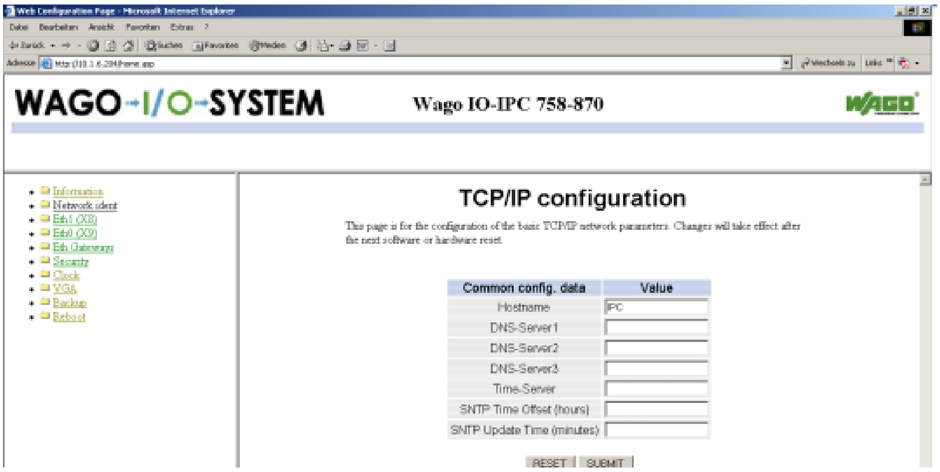


Fig. 3-14: TCP/IP configuration window p2870n4d

In order to configure Ethernet interface X8, proceed as follows:

1. In the left navigation area, click on “Eth1 (X8)” and on “TCP/IP.”

2. Enter the IP address and the subnet mask for the Ethernet interface X8.

3. Select the “Ethl (X8) Enabled” checkbox.

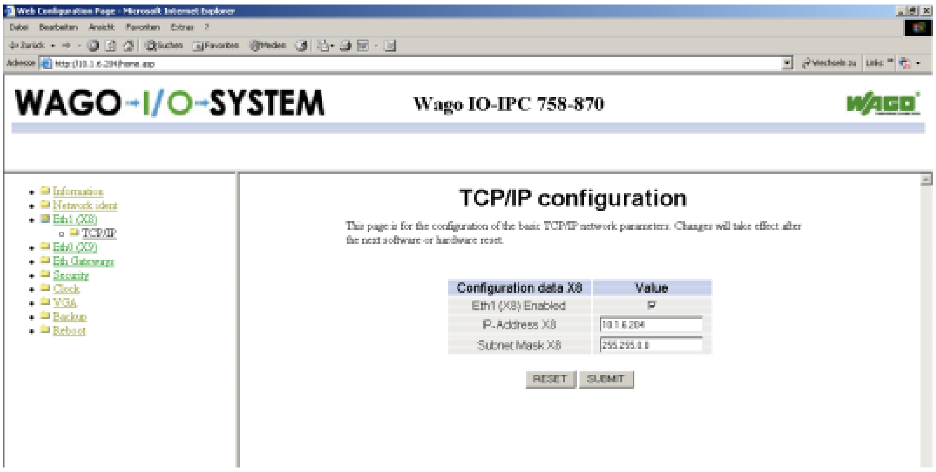


Fig. 3-15: TCP/IP configuration window Eth1 (X8) p2870n3d

4. Click on the “**Submit”** button and restart the I/O-IPC to save the selection of the protocol. To do so, turn off the power supply of the I/O-IPC and turn it on again, or press the **“Reset”** button.

If not configuring the Ethernet interface X8, proceed as follows:

1. In the left navigation area, click on “Eth1 (X8)” and on “TCP/IP.”

2. To disable X8, unselect the checkbox in the column “Value.”
   Click on the “**Submit”** button to save the changes.

3. Restart the I/O-IPC. To do so, turn off the power supply of the I/O-IPC and turn it on again, or press the **“Reset”** button.

3.1.12 Information on Programming with WAGO-I/O-*PRO* CAA

This section only includes programming notes specifically for the PLC program WAGO-I/O-*PRO* CAA in connection with the I/O-IPC. Please find detailed information on the entire range of functions on the CD-ROM WAGO-I/O-*PRO* CAA 2.3.x, which can be ordered from WAGO.

If the I/O-IPC is used as a PLC, it is possible to control the I/O modules at the internal bus and the connected fieldbus devices locally via WAGO-I/O-*PRO* CAA. WAGO-I/O-PRO CAA is an IEC 61131-3 compliant programming and configuration tool for the I/O-IPC. I/O modules that are controlled locally can also be controlled remotely via Ethernet interfaces.

To create a new project, proceed as follows:

1. In order to start WAGO-I/O-PRO CAA click on the **"Start"** menu bar and select: Programs → WAGO Software → CoDeSys for Automation Alliance→ CoDeSys V2.3 → CoDeSys V2.3.
2. CoDeSys opens and displays the project that was last been worked on. If opening CoDeSys for the first time, the "Target settings" window will open.
3. In order to create new projects, click on "File" and select "New."
   The "Target settings" window opens.
4. Select the target system for the project from the target settings.

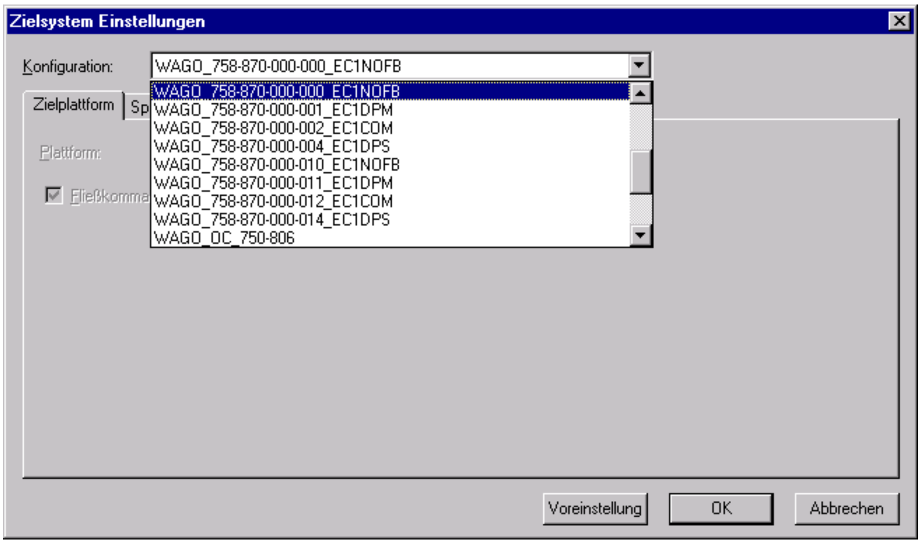


Fig. 3-16: Target settings in WAGO-I/O *PRO* CAA p2870n5d

Types of the I/O-IPC, with different process interfaces for **32-MB** versions that can be selected are:

WAGO_758-870/_EC1NOFB	Internal bus interface, no fieldbus interface
WAGO_758-870/000-001_EC1DPM	Internal bus interface and PROFIBUS (Master)
WAGO_758-870/000-002_EC1COM	Internal bus interface and CANopen (Master)
WAGO_758-870/000-004_EC1DPS	Internal bus interface and PROFIBUS (Slave)

Types of the I/O-IPC, with different process interfaces for **128-MB** versions that can be selected are:

WAGO_758-870/010_EC1NOFB	Internal bus interface, no fieldbus interface
WAGO_758-870/000-011_EC1DPM	Internal bus interface and PROFIBUS (Master)
WAGO_758-870-000-012_EC1COM	Internal bus interface and CANopen (Master)
WAGO_758-870-000-014_EC1DPS	Internal bus interface and PROFIBUS (Slave)

5. After acknowledging by clicking the **“OK”** button, a dialog window opens and requests programming language selection (e.g., IL, LD, etc.).

6. It is now possible to tailor a program to suit specific requirements.

3.1.12.1 WAGO-I/O-*PRO* CAA Library Elements

WAGO-I/O-*PRO* CAA provides several libraries for different programming applications in accordance with IEC 61131-3. They include many modules that make it easier and faster to create an application program. Some libraries are implemented automatically into a new project as standard. The following libraries are available on the CD-ROM WAGO-I/O-*PRO* CAA, version 2.3.x:

Library:	**Description:**
Analyzation.lib	This library contains the module "AnalyzeExpression" to analyze logical expressions.
AnalyzationNEW.lib	This library contains modules to analyze expressions. If a complex expression has the total value FALSE, it is possible to determine those components that contribute to this value.
BusDiag.lib	Contains a function to analyze the fieldbus devices (PROFIBUS, CANopen) that are connected to the I/O-IPC. There is an interaction between the Hilscher.lib and the BusDiag.lib. Therefore, it is only possible to use one library at a time because both use the same interface.
Ec1Lib.lib	Contains a function to read the value set by the DIP switch.
Hilscher.lib	There is an interaction between the Hilscher.lib and the BusDiag.lib. Therefore, it is only possible to use one library at a time because both use the same interface.
Iecsfc.lib	Contains sequential function charts that comply with the IEC standard.
KbusLib.lib	Contains functions to read and write data for the control of the internal bus and of the integrated COM interfaces.
LinuxRTSystemLib.lib	This library contains the function "SysGetLastError" that can be used to retrieve the error number and text describing the error that occurred last.
MiscLib.lib	Contains functions to read and write data for the control of integrated components.
Mod_com.lib	This library contains modules and functions to access the process image of the inputs or the outputs. The following functions are supported: - CRC16 - GET_DIGITAL_OUTPUT_OFFSET0 - GET_DIGITAL_INPUT_OFFSET - KBUS_ERROR_INFORMATION - MOD_COM_VERSION - PI_INFORMATION
NetVarUdp_Lib_V23.lib	Contains functions to support the network variables. The firmware 0107 requires the current version of this library.
Standard.lib	Contains all functions and function blocks required by the IEC61131-3 as standard components for an IEC programming system.

StandExt.lib	Contains auxiliary functions for string processing. The following functions are supported: - STRING_COMPARE - STRING_TO_ASCIIBYTE
Library:	**Description:**
SysLibCallback.lib	This library contains the functions "SysCallbackRegister" and "SysCallbackUnregister" that serve to enable defined callback functions for runtime events. Both are BOOL type functions and they return TRUE if the indicated callback function can be registered or de-registered. They are processed synchronously.
SysLibCom.lib	Serial communication with the target system.
SysLibDir.lib	The functions of this library allow synchronous access to a file directory system of the target system. Entries in the directory can be read and changed.
SysLibFile.lib	Contains functions that support a file system on the target computer.
SysLibIecTasks.lib	Special functions for IEC task handling.
SysLibPlcCtrl.lib	This library is not fully implemented.
SysLibPorts.lib	Communication with external hardware via port addresses; e.g., real-time clock, graphics controller, etc.
SysLibProjectInfo.lib	Reading the project information.
SysLibRTC.lib	The SysRtcGetTime function always displays UTC time, independent of the SNTP offset value. The offset value is only considered in the WBM. It is not possible to check the capacity of the battery via SysRtcCheckBattery function.
SysLibSem.lib	Creating and using semaphores to synchronize tasks.
SysLibSockets.lib	This library supports the socket access for communication via TCP/IP and UDP.ss
SysLibStr.lib	Functions for string handling (copy, compare, etc.).
SysLibTargetVisu.lib	Target visualization
SysLibTime.lib	Additional functions to read the real time clock (see also SysLibRtc.lib) and to view the task. Time evaluation in the CoDeSys task configuration required.
SysLibVisu.lib	Defines structures for object visualization.
SysTaskInfo.lib	To read task information.
System.lib	This library provides information on the current and the average execution time of all IEC tasks. The following functions are supported: - GET_ACT_CYCLE - GET_MAX_CYCLE - GET_MIN_CYCLE - GET_PLC_ACT_CYCLE - GET_PLC_MAX_CYCLE - GET_PLC_MIN_CYCLE - GET_PROGRAM_ID - SYSTEM_VERSION

Library:	Description:
Util.lib	This library contains additional function blocks that can be used for: BCD conversion, bit/byte functions, mathematical help functions, as controller, signal generators, function manipulators and for analog value processing.
Util_no_Real.lib	This library contains additional function blocks that can be used for: BCD conversion, bit/byte functions, mathematical help functions, as controller, signal generators, function manipulators and for analog value processing. However, all function blocks and functions that contain REAL variables are excluded.

After the installation of WAGO-I/O-*PRO* CAA on the host PC, libraries are stored in the following directory:
C:\Programs\WAGO Software\CoDeSysV2.3\Targets\WAGO\ Libraries\io_ipc
If not using version WAGO-I/O-*PRO* CAA 2.3.x, the menu path may differ from the one indicated above.

If using additional libraries in the project, proceed as follows:

1. Select the **“Resources”** tab in the left window.
2. Click on the library manager in the tree structure. Right click the mouse button in the upper-middle window of the library manager and select “Additional Library.”
3. After a library has been selected, it is possible to access the POU (Program Organization Units), the data types and the global variables. They are used just like user-defined program objects.

More information:
For more information on the function blocks and details on the use of the software, please refer to the WAGO-I/O-*PRO* CAA manual. It can also be downloaded from the WAGO Web site: www.wago.com.

3.1.12.2 Transferring an IEC 61131-3 Application from a PC to the I/O-IPC

An IEC 61131-3 application can be transferred from the PC to the I/O-IPC via Ethernet or CF card.

The dialog window “Communication parameter” can be opened through the menu *Online* → *Communication parameter*. Please make sure that the “Communication parameter” window contains the following setup data:
Port=1200, Motorola byteorder=No.
Also verify that the indicated IP address is correct.

More information:
For more information on the communication and details on the use of the software, please refer to the WAGO-I/O-*PRO* CAA manual. It can also be downloaded from the WAGO Web site: www.wago.com.

3.1.13 Information on the Web-Based Management System

The I/O-IPC contains a number of HTML pages that provide information and options for the configuration.

Any Web browser can be used to view the pages.

Note:
Make sure that the Web browser set up allows for bypassing the proxy server for local addresses. The procedure depends on the operating system and on the browser that is used.

Use the TCP/IP link to view the settings for the TCP/IP protocol and change them if required.

3.1.14 CoDeSys Web Visualization

3.1.14.1 Information

The CoDeSys Web visualization is based on Java technology. An applet is a program written in Java. All Java programs require a Java Runtime Environment (JRE). An applet is stored in the file system of a Web server and can be made accessible to browsers via an HTML page.

The Web visualization is created using the graphical editor from CoDeSys. For each of these pages, a description file in the XML format is generated using the information. Find these files in the subfolder *"visu"* of the CoDeSys installation path. There is also the HTML start page "WebVisu.htm," the Java archive "webVisu.jar" in which the applet (webvisu.class) is stored in a compressed form.

In order to view the Web visualization, enter the following URL into the address bar: http://<IP>:8080/webvisu.htm. The browser interprets the file and finds the applet tag. In addition to the pixel width and height of the applet, it also includes the name and the path to the Java sources. The browser uses this information to start the Java runtime environment. After the Java archives "webvisu.jar" and "minml.jar" have been loaded, an instance of the Java class "webvisu.class" is created. The instance analyzes the call parameters UPDATETIME and STARTVISU and requests a list of all visualization pages.
In the next step, the XML description files for all visualization pages are loaded.
Finally, the process data communication for the visualization page that was configured in STARTVISU is established. When delivered, this page is "PLC_VISU."

3.1.14.2 Limitations of the CoDeSys Visualization

The visualization that is integrated in CoDeSys provides the three different versions “HMI,” “TargetVisu” and “WebVisu” that are all supported by the IPC. Depending on the version, there are technological limitations.

Several options of the complex visualization objects “Alarm” and “Trend” are only provided by the “HMI” version. This applies, for example, to sending of e-mails as an response to an alarm or to the navigation through historical trend data as well as the creation of the data.

In contrast to the “HMI,” the Web visualization on the IO-IPC is implemented using much more limited resources. While the “HMI” can rely on most desktop PC resources, take the following limitations into account when the Web visualization is used:

Adaptation to the File System

The overall size of the PLC program, visualization files, bitmaps, log files, configuration files, etc. must fit into the file system.

Process Data Buffer

The Web visualization uses its own protocol to exchange process data between the applet and the control. The process data is transmitted as an ASCII-encoded file. The pipe character (|) is used as a delimiter between two process values. For this reason, the required space of a process data variable in the process data buffer not only depends on the data type, but also on the process value itself. A “WORD” type variable occupies between one byte for the values 0 to 9 and five bytes for values from 10000 upward.
The selected format (ASCII + |) only allows a rough estimation of the space required by the process data in the process data buffer. If the amount of ASCII-encoded process data is exceeded, the Web visualization will no longer function properly.

Computer Performance/Processor Time

The IO-IPC is based on a real-time operating system. High-priority processes such as the PLC program, for example, will interrupt or eliminate low-priority processes. The Web server that is responsible for the Web visualization belongs to this group of low-priority processes.

Make sure when configuring tasks that there is sufficient processor time available for all processes. The “freewheeling” task call option is not suitable in conjunction with the “WebVisu,” as in this case the high-priority PLC program suppresses the Web server. Instead of this, use the “cyclic” task call option with a realistic value.

Network Load

The IO-IPC has one CPU responsible both for running the PLC program and for handling the network traffic. Ethernet communication demands that every telegram received is processed, regardless of whether it is intended for the I/O-IPC or not.

A significant reduction of the network load can be achieved by using switches instead of hubs.

However, broadcast telegrams can only be suppressed by the sender or by means of configurable switches, which feature broadcast limiting. A network monitor such as www.ethereal.com will give an overview of the current network load.

3.1.14.3 FAQs

How can the applet be optimized for special screen resolutions?

In order to optimize the Web visualization for a PDA or a touch panel with a fixed resolution, proceed as follows:
In the “Target system settings,” enter the pixel width and height in the tab “Visualization.” When the visualization is created, the visible area is highlighted in gray. However, the actual pixel width and height of the Web visualization are defined by the attributes “Height” and “Width” of the HTML APPLET tag in the “webvisu.htm” file. Do not forget to also adapt these parameters to the existing resolution.

When delivered, the “webvisu.htm” has the value “99 %” for height and width; i.e., the size is specified using a relative value, that can be processed without any problems by the Java Runtime Environment (JRE) used by Sun and Microsoft. The CrEme from NSIcom, however, interprets the value as an absolute value, so that only 99*99 pixel of the upper-left corner are displayed. Since this area is rarely used, this could give the impression that the visualization is not displayed at all even though the upper-left corner has been displayed correctly.

Which JRE should be used?

It is recommended to use Java2 standard edition Version 1.5.0 (J2SE1.5.0_06) or higher.
Other JREs have also been tested and it is also possible to use MSJVM3810 from Microsoft or CrEme from NSIcom.

Note:
CoDeSys Version 2.3.5.3 does not work with CrEme and MSJVM3810.

Should the Java Cache be used?

After a standard installation, the cache is enabled. If the cache is enabled, the JRE uses it to store applets and Java archives. If the Web visualization is called up a second time, it requires considerably less time to start because the applet (approx. 250 kb) does not need to be reloaded by the network, but is already available in the cache. This is especially advantageous when network connections are slow.

Note:
The Java archives may not be transferred into the cache completely due to network failures. In this case, the cache must be cleared manually or be disabled.

What needs to be observed when using an HTTP proxy for the Web visualization?

If using a proxy server for the Web visualization, both a SOCKS proxy and the HTTP proxy are required. The HTTP proxy provides the HTML pages and the Java archives.
The SOCKS proxy is required for the process data exchange between the applet and the control.

Why does the visualization element "TREND" in the Web visualization only work "Online?"

Select the following settings for the visualization elements: tab *Resources* → *Target Settings*.
Enable the "Web visualization" (select check box) and "Trend data recording within the control." Otherwise, the trend data is stored on the hard drive of the CoDeSys development PC. This would require a permanent connection between the IO-IPC and the CoDeSys gateway. If the connection is interrupted, this may lead to an unpredictable behavior of the IO-IPC.

In the TREND configuration dialog, there is a choice between the operating modes "Online" and "History." The IO-IPC only supports the "Online" operating mode for visualization projects because it is not possible to configure the maximum size (quota) of the trend data files (*.trd). An uncontrolled accumulation of trend data may lead to an unpredictable behavior of the IO-IPC.

In most cases, the use of the "HISTOGRAM" visualization element is the better choice, as this gives full control over the time and number of measurements and thus the amount of memory required.

What needs to be observed when the visualization element "ALARM TABLE" is used in the Web visualization?

The status of this component is best described as an "Add-On"; i.e., an extra that is free of charge and without warranty.

Select the following settings for the visualization elements:
tab *Resources* → *Target Settings*.
Enable the "Web visualization" (select check box) and "Alarm treatment within the control." Otherwise, the alarm data is processed on the CoDeSys development PC. This would require a permanent connection between the I/O-IPC and the CoDeSys gateway. If the connection is interrupted, this may lead to unpredictable IO-IPC behavior.

3.1.15 LED Indication

The operational status of the I/O-IPC or of the node is indicated via nine LEDs. When using the supply module 750-602, the LED “D” is always off.

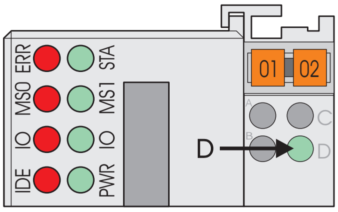


Fig. 3-17: Indicators g287002x

3.1.15.1 Fieldbus status: ERR/STA

The meaning of the ERR and STA LEDs depends on the fieldbus used and is described for the particular fieldbus systems in section 3.2.

3.1.15.2 Module status: MS0/MS1

These LEDs are used by the PLC runtime system during start-up.

LED:		Indicates:
MS0 (red)	**MS1 (green)**	
Off	Off	WAGO-I/O-*PRO* CAA has not yet been started.
Blinks slowly	Off	Initializing
Blinks quickly	Off	An error has occurred, WAGO-I/O-*PRO* CAA is not working correctly.
Off	Blinks (0.5 Hz)	WAGO-I/O-*PRO* CAA has started successfully. If an error occurs, the blinking stops.

After start-up, the LEDs can also be used in a PLC program by the library MiscLib.lib (see section 3.1.12.1). In this case, the table shown above is no longer valid.

3.1.15.3 Internal Bus Status: IO/IO

The IO LEDs indicate the operating status of the internal bus. In the event of a system failure, they also display error codes (blink codes).

LED:		Indicates:
IO (red)	**IO (green)**	
Blinks quickly	Blinks quickly	I/O-IPC start-up.
Blinks quickly	Off	During the start-up of the I/O-IPC: Internal bus is initialized. During start-up, the LED blinks quickly for about 1–2 seconds.
Off	On	I/O-IPC is working properly.
Blinking sequences	Off	After start-up of the I/O-IPC: Errors are indicated by three successive blinking sequences. Between the sequences are short intervals.

During start-up of the I/O-IPC, the red and the green I/O LEDs blink when the internal bus is initialized. After an error-free start-up, the green LED is on.
In the event of a failure, the red I/O LED blinks and displays an error code.
The green IO LED is off.

3.1.15.4 IDE and Power Supply Status: IDE/PWR

The green PWR LED (Power) is on as soon as power is supplied to the I/O system and it is off when the power supply is interrupted.
The IDE LED blinks when the CF card (integrated or external) is accessed.

After a successful initialization the LEDs are automatically managed by the operating system of the I/O-IPC. In this case, the LEDs indicate the following:

LED:		**Indicates:**
IDE (red)	**PWR (green)**	
On/Off	Blinks (1 Hz)	The system has been configured with a fixed IP address and works properly. The red LED indicates IDE activity.
On/Off	Blinks (10 Hz)	The system has been configured with a fixed IP address (DHCP, BootP) and works properly. The red LED indicates IDE activity.
Blinks (10 Hz)	Blinks (10 Hz)	The system is configured for DHCP and BootP. The system could not determine a valid IP address.
Blinks	Off	Undefined error. Possibly invalid configuration.

The hardware function of the LEDs can be overridden by the functions provided by the MiscLib.lib.

3.1.15.5 Power Supply Status of the Supply Module 750-602

The green LED (D) indicates the status of the 24 V power supply of the power jumper contacts:

LED:	**Indicates:**
D (green)	
Off	No power jumper contact operating voltage.
On	Operating voltage for power jumper contacts is available.

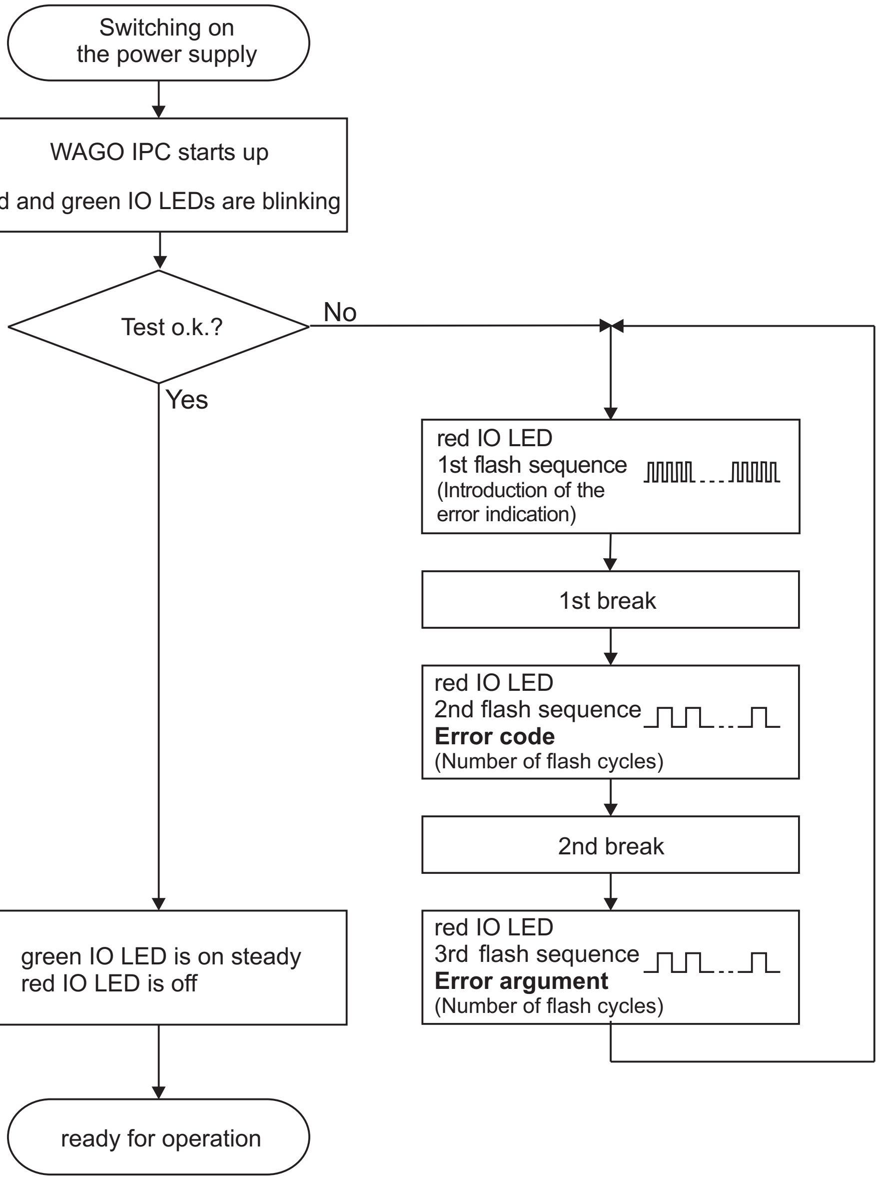


Fig. 3-18: IO LED indication g287004d

If required, the IO LEDs can be used by the library MiscLib.lib (see section 3.1.12.1). In this case, the status messages are no longer sent from the internal bus to the LEDs.

3.1.15.6 Blink Codes of the IO LEDs

Detailed error messages are indicated by blink codes. An error is indicated cyclically by up to three blink sequences.

1. The **Error Display** starts with the first blink sequence (approx. 10 Hz).

2. After a break, the second blink sequence starts (approx. 1 Hz).
   The number of light pulses indicates the **Error Code**.

3. After another break, the third blink sequence starts (approx. 1 Hz).
   The number of light pulses indicates the **Error Argument**.

Example of an error message:

Error: The 13th I/O module was removed.

1. The I/O LED starts the **Error Display** with the first blink sequence (approx. 10 Hz).

2. After the first break, the second blink sequence starts (approx. 1 Hz). The I/O LED blinks four times, thus indicating **Error Code** 4 (data error internal bus).

3. After the second break, the third blink sequence starts. The I/O LED blinks twelve times. **Error Argument** 12 means that the internal bus is interrupted behind the twelfth I/O module.

Error messages and blink codes

If it is not possible to troubleshoot the problems with the following measures, contact the WAGO AUTOMATION Support:

Kontakttechnik GmbH & Co. KG
Technical Support
Tel: +49 571 887-555
Fax: +49 571 887-8555
E-Mail: support@wago.com

Error argument:	Error description:	Solution:
Error Code 1: Hardware and configuration fault		
-	Invalid parameter checksum of the internal bus controller	Switch off the node's supply voltage, replace the I/O-IPC and switch on the supply voltage again.
1	Internal buffer overflow (max. amount of data exceeded) during inline code generation	Turn off the node's supply voltage, reduce the number of modules and turn on the supply voltage again.
2	Data type of the I/O module(s) is not supported	Update the firmware of the I/O-IPC. If the error persists, detect the defective I/O module as follows: Turn off the supply voltage. Plug the end module in the middle of the fieldbus node. Turn on the power supply again. – **If the LED keeps blinking**, turn off the power supply and plug the end module in the middle of the first half of the node (toward the I/O-IPC). – **If the LED does not blink**, turn off the power supply and plug the end module in the middle of the second half of the node (away from the I/O-IPC). Turn on the power supply again. Repeat this procedure until the faulty I/O module is detected. Replace the defective I/O module.
3	Unknown module type of flash program memory	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
4	Error occurred while writing to the flash memory	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
5	Error occurred while erasing a flash sector	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
6	The I/O module configuration after an internal bus reset differs from the one after the last I/O-IPC start-up.	Restart the I/O-IPC by turning off the power supply and turning it on again.
7	Error occurred while writing to the serial EEPROM	Turn off the supply voltage of the node, replace I/O-IPC and turn on the supply voltage again.

Error argument:	Error description:	Solution:
8	Invalid hardware/firmware combination	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
9	Invalid checksum in the serial EEPROM	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
10	Fault while initializing the serial EEPROM	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
11	Error occurred while reading from the serial EEPROM	Turn off the node's supply voltage, reduce the number of modules and turn on the supply voltage again.
12	Access time for the serial EEPROM has been exceeded	Turn off the node's supply voltage, replace I/O-IPC and turn on the supply voltage again.
14	Maximum number of gateway modules or mailbox modules exceeded	Turn off the node's supply voltage, reduce the number of gateway or mailbox modules and switch on the supply voltage again.
Error code 2: not used		
-	-	-

Error code 3: Internal bus protocol error		
-	Internal bus communication failure; defective device could not be detected	If there are internal supply modules (750-613) in the fieldbus node, first make sure that these modules are supplied with power. This is indicated by the “I/O LEDs.” If all I/O modules are connected correctly, or if there are no 750-613 supply modules in the field-bus node, detect the defective I/O module as follows: Turn off the supply voltage of the node. Plug the end module in the middle of the fieldbus node. Turn on the power supply again. **– If the LED keeps blinking**, turn off the power supply and plug the end module in the middle of the first half of the node (toward the I/O-IPC). **– If the LED does not blink**, turn off the power supply and plug the end module in the middle of the second half of the node (away from the I/O-IPC). Turn on the power supply again. Repeat this procedure until the faulty I/O module is de-tected. Replace the defective I/O module. If only one I/O module is left, but the LED is still blinking, then either this module or the I/O-IPC is defective. Replace the defective component.

Error code 4: Physical error internal bus		
-	Internal bus data communication error or interruption of the internal bus at the I/O-IPC	Turn off the node's supply voltage. Plug one I/O module with process data behind the I/O-IPC and write down the error argument that is indicated after power-on. If no error argument is indicated by the red I/O LED, replace the I/O-IPC. Otherwise, detect the defective I/O module as follows: Turn off the supply voltage of the node. Plug the end module in the middle of the fieldbus node. Turn on the power supply again. **– If the LED keeps blinking,** turn off the power supply and plug the end module in the middle of the first half of the node (toward the I/O-IPC). **– If the LED does not blink,** turn off the power supply and plug the end module in the middle of the second half of the node (away from the I/O-IPC). Turn on the power supply again. Repeat this procedure until the faulty I/O module is detected. Replace the defective I/O module. If only one I/O module is left, but the LED is still blinking, then either this module or the I/O-IPC is defective. Replace the defective component.
n*	Interruption of the internal bus behind the nth process data module	Turn off the node's supply voltage, replace the (n+1)th process data module and turn on the supply voltage again.
Error code 5: Internal bus initialization error		
n*	Register communication error during the initialization of the internal bus	Turn off the node's supply voltage, replace the nth process data module and turn on the supply voltage again.
Error code 6: Design error in the node configuration		
5	Maximum size of the process image exceeded	Turn off the supply voltage of the node, reduce the number of modules and turn on the supply voltage again.
Error code 7: not used		
-	-	-
Error code 8: not used		
-	-	-
* The number of light pulses (n) indicate the position of the I/O module. I/O modules without data are not counted (e.g., supply modules without diagnostics).		

Error code 9: CPU exception error		
1	Invalid program statement	Program sequence error. Please contact WAGO-I/O Support.
2	Stack overflow	Program sequence error. Please contact WAGO-I/O Support.
3	Stack underflow	Program sequence error. Please contact WAGO-I/O Support.
4	Invalid event (NMI)	Program sequence error. Please contact WAGO-I/O Support.

3.1.16 Error Response

3.1.16.1 Internal Bus Error

If an internal bus error occurs (e.g., an I/O module is removed), all output modules are turned out. The red I/O LED blinks and generates an error message (see section 3.1.15.3). The error message is displayed by a blink code (error code and error argument).

The PLC program can also detect this error. The I/O-IPC has a special library for this purpose (KbusLib.lib) and is also able to restart the internal bus.

As soon as the internal bus is restarted, the I/O functions are available again.

Examples for the Library KbusLib.lib functions:

The “KbusGetLifeStatus” function is used to detect if the internal bus works normally or if it has been disabled after an error was detected.

The “KbusGetStatusData” function reports the current status of the internal bus. In the event of a failure, the status is provided automatically and can be accessed directly via this function.

If there is no error, or the error has been eliminated by a reset, it is possible to find out the current status by the following calls: “KbusSetStatusTrig,” “KbusGetStatusTrig” and “KbusGetStatusData.”

3.1.16.2 Fieldbus Error

The status of the fieldbus can be monitored via functions of the standard library BusDiag.lib.

The “DiagGetBusState” function provides the general status of the fieldbus, the “DiagGetState” function provides the status of the connected devices.

3.1.16.3 Troubleshooting the CoDeSys Web Visualization

If experiencing problems while working with the CoDeSys Web visualization, use the following table to find the solution. If it is not possible to troubleshoot the problems, contact WAGO AUTOMATION Support:

Error:	Solution:
Internet Explorer reports the error “APPLET NOT INITIATED”	Close all Internet Explorer windows and restart the Explorer. If the error remains, this might be due to a missing or damaged file. Using FTP, check if the entire Java archive “webvisu.jar” is available in the “/PLC” folder of the IO-IPC. Find the original file in the CoDeSys installation path (normally under *C:\Programs\WAGO Software\CoDeSys V2.3\Visu\webvisu.jar*). If necessary, replace the defective file using FTP or force a complete download of all files in CoDeSys via the commands: CLEAN ALL, REBUILD ALL and LOG-IN.
Web visualization is not displayed	Ensure that the JRE has been installed. Check the firewall settings; e.g., if port 8080 is open.
Web visualization freezes. Web visualization stops after a longer period of time.	The intervals between calls in the task configuration are too short. Therefore the Web server of the IO-IPC, which is a low-priority process, is not getting enough computing time or none at all. If no task configuration (explicit) was stored, the PLC_PRG task (implicit) is run with priority 1. As a result, the Web server is not getting enough computing time. If using the Web visualization, always store a task configuration. The call interval should not be shorter than three times the average execution time. When determining the execution time, make sure that the PLC program has reached a steady state.
Web visualization cannot be loaded into the control	There may not be enough space for all data in the file system of the I/O-IPC. Erase the data that is no longer required; e g., via FTP.
Bitmap is not displayed	If there are umlauts (e.g., ä) in the file name, the Web server cannot interpret the name and the file is not transferred to the browser.
Java console reports: “Class not found”	The JRE does not find the entry point for the “webvisu.class” in the Java archive “WebVisu.jar.” The Java archive is probably incomplete. Delete the “WebVisu.jar” from the Java cache or disable the cache. In this case, the I/O-IPC requests the archive (applet) again. If the problem remains, load the project into the IO-IPC again.
Web visualization is static, all process values are “0”	This is caused by process data communication failure. If using a proxy server for the Web visualization, both a SOCKS proxy and an HTTP proxy are required for the process data exchange.

3.1.17 Technical Data

System Data:	
No. of nodes	Limited by ETHERNET specification
Transmission medium	Twisted Pair S-UTP 100 Ω Cat. 5
Buscoupler connection	RJ-45
Maximum Ethernet segment length	100 m between hub station and I/O-IPC; max. length of network limited by ETHERNET specification. The Ethernet connection X 9 is an exception and allows a length of 30 m.
Baud rate	10/100 Mbit/s
Protocols	Modbus/TCP (UDP), HTTP, BootP, DHCP, DNS, SNTP, FTP, SMTP
Programming	WAGO-I/O-PRO CAA
IEC 61131-3-3	IL, LD, FBD, ST, FC
Approvals:	
UL	E175.199, UL 508
Conformity marking	Œ
Technical Data:	
Battery/Life span	Approximately one year at room temperature
Max. number of I/O modules	63
Max. size of the local process image	500 bytes inputs, 500 bytes outputs
Max. size of the process image of the fieldbus master	3584 bytes inputs, 3584 bytes outputs
Max. size of the process image of the PROFIBUS slaves	244 bytes inputs, 244 bytes outputs
Configuration possibility	Via PC
Program memory (32 MB version)	2 MByte
Data memory (32 MB version)	2 MByte
Program memory (128 MB version)	16 MByte
Data memory (128 MB version)	32 MByte
Residual memory	128 kByte (SRAM)
Max. number of socket connections	3 for HTTP, 10 for Modbus/TCP, 2 for WAGO-I/O-*PRO* CAA
Voltage supply	DC 24 V (-15 %–+20 %)
Internal current consumption$_{max.}$	770 mA bei 24 V
Total current for I/O modules$_{max.}$	1A at 5 V
Voltage via power jumper contacts	DC 24 V (-15 %–+20 %)
Current via power jumper contacts$_{max}$	DC 10 A
Dimensions (mm) W x H x L	160 x 65* x 100 (*from upper-edge of DIN rail)

Weight	Approx. 540 g
Environmental Requirements:	
Operating temperature	0 °C–+55 °C
Storage temperature	-10 °C–+85 °C
Standards and Regulations (see section 2.2)	
EMC Immunity to interference	Acc. to EN 50082-2 (95)
EMC Emission of interference	Acc. to EN 50081-1 (93)

3.1.18 Installation Notes

3.1.18.1 24 V Power Supply

The following picture shows the connection of the 24 V power supply:

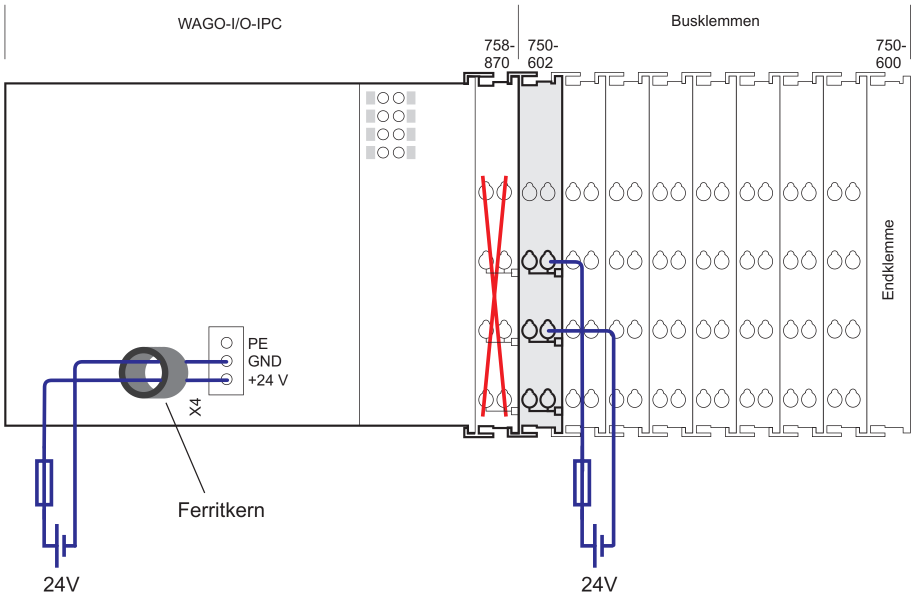


Fig. 3-19: 24 V power supply g287033d

Attention:
The supply voltage for the power jumper contacts (field supply) must not be derived from the I/O-IPC connections!
The 750-602 supply module (included in the delivery) must be used!

3.1.18.2 Mailbox Modules

The I/O-IPC currently supports only three mailbox modules.
As of April 2007.

Mailbox functions are supported by the following WAGO modules:

753-646 KNX/EIB/TP1 module
750-655 AS-Interface master
750-670 Stepper controller RS 422/24 V/20 mA
750-671 Stepper controller 24 V/1.5 A
750-482 2AI 4 ... 20mA, HART

3.1.18.3 OnBoard IOs (X5)

The I/O-IPC has two digital inputs and two digital outputs.

The digital inputs have the following features:

Voltage range	Low: -3 V–+5 V High: +11 V–+30 V (+24 V standard)
Max. current per channel	5 mA
Channels	2
Input impedance	Min. 1.5 kOhm Max. 6 kOhm at 30 V
Characteristics	Optocoupler 2 kV Low pass filter Current limitation Overvoltage protection Reverse voltage protection

The connections to digital inputs are shown in the picture below:

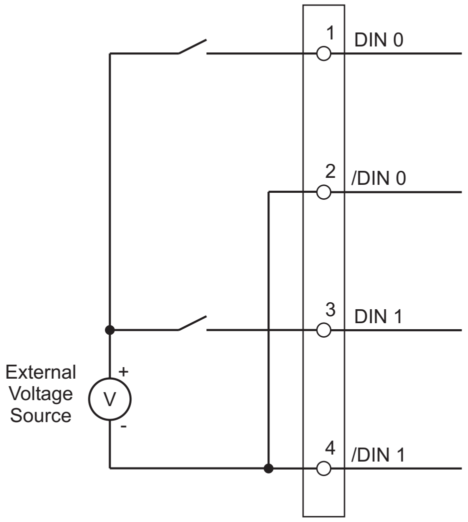


Fig. 3-20: Connection of the integrated inputs g287030e

The digital outputs have the following features:

Voltage range	Depends on external circuit
Max. current per channel	100 mA (typ.) 200 mA (absolute maximum)
Channels	2
Characteristics	Optocoupler 2 kV Current limitation Reverse voltage protection

The following picture shows the connections of the digital outputs:

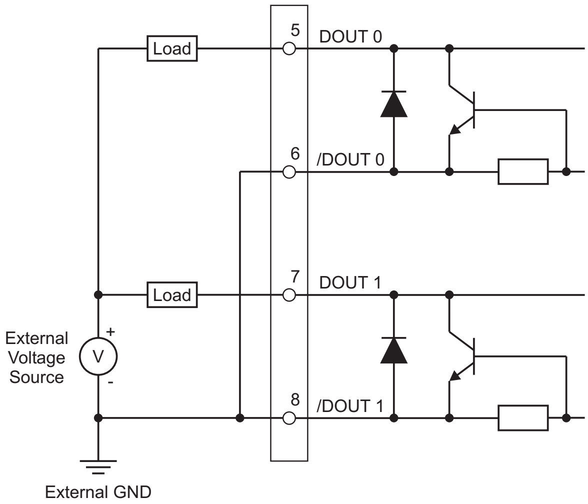


Fig. 3-21: Connection of the integrated outputs g287031e

3.1.18.4 Watchdog Output

The I/O-IPC has a watchdog output. This monitoring channel verifies if the I/O-IPC is operating normally and can detect errors (e.g., operating system crash, hardware failure, etc.).

The watchdog output has the following features:

Timeout time	508 ms
Relay output	
Voltage $_{max.}$	220 V DC
Current $_{max.}$	2 A

The connection of the watchdog output is shown in the picture below:

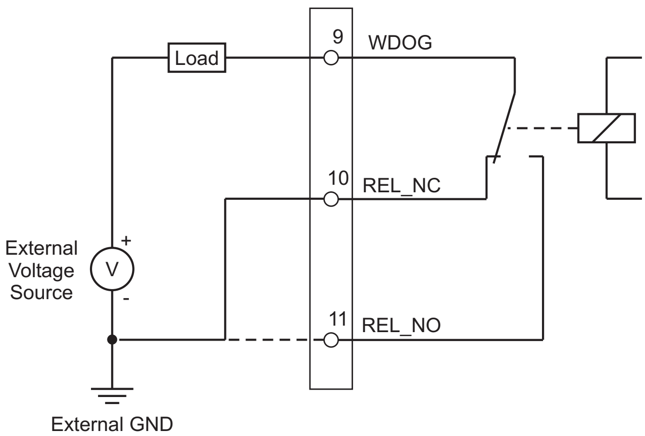


Fig. 3-22: Connection of the watchdog output g287032e

3.1.18.5 Using an Internal Bus Extension

The I/O-IPC supports internal bus extension; a total of 63 I/O modules may be connected to the internal bus.

The manuals for both the 750-627 and 750-628 provide information on how to connect the internal bus extension and what to observe. The manuals are available on the WAGO web site under *Service* → *Documentation* → *WAGO-I/O-System 750* → *System modules*.

Additionally, the “WAGO Extension Setting 759-314” must be installed.
It is available under *Service* → *Downloads* → *Automation*.

3.1.19 Installation Notes for GL Applications

Currently, only the I/O-IPCs with the following item numbers are approved for GL applications:

- 758-870
- 758-870/000-001 (Profibus master)
- 758-870/000-004 (Profibus slave).

Only the following interfaces for these IPC versions have been approved for GL applications:

- PROFIBUS interface (X3)
  The maximum admissible baud rate is 1.5 MBaud
- Ethernet interfaces (X8, X9)
- Power supply 24 V DC (X4).

3.1.19.1 Approved Mounting Position of the IPC

Only one mounting position is approved for the IPC. The IPC must be mounted on a horizontal DIN rail that is attached to a vertical mounting plate.

3.1.19.2 Power Supply

In order to smooth the voltage applied to the WAGO-I/O-IPC and the following I/O modules, it is necessary to use two 750-626 filter modules, as shown in the following illustration. The IO-IPC must only be used in this configuration!

Ferrites or cable length limitations are not necessary.

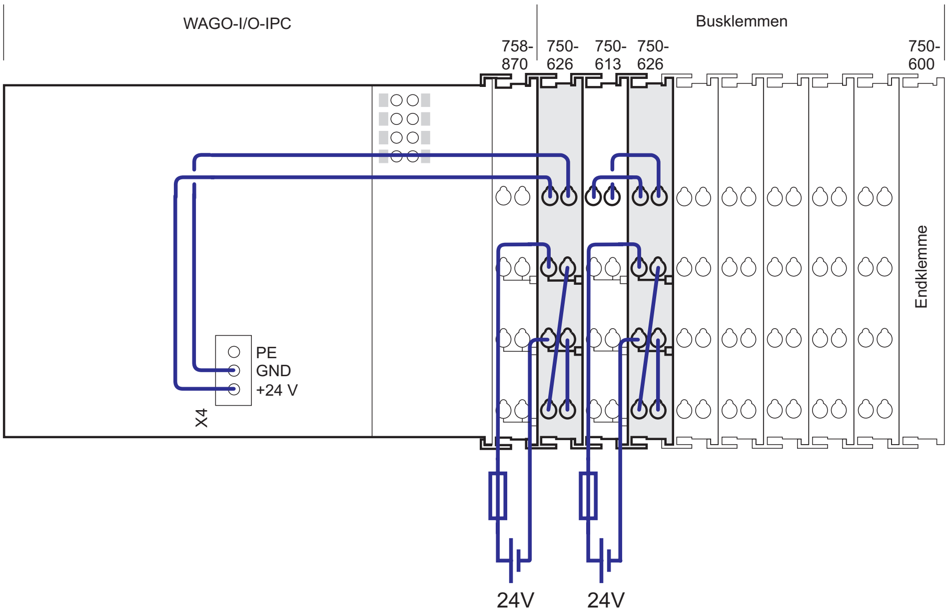


Fig. 3-23: Power supply g287031d

3.2 I/O-IPC 758-870/000-001 and 758-870/000-011

3.2.1 Overview of the Process Data

It is possible to connect up to 125 slaves to the PROFIBUS master. The master can receive up to 3584 bytes of input data from the slaves and send 3584 bytes of output data to the slaves. In order to support this amount of data, it is essential to meet the guidelines for PROFIBUS networks. Depending on the driver performance of the slave with the lowest performance, a repeater must be used after 32 slaves in order to guarantee the required signal quality.

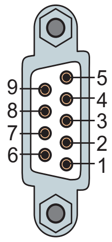
The following table lists the limits for the amount of process data:

Maximum amount of input data from all slaves	3584 bytes
Maximum amount of output data for all slaves	3584 bytes
Available node addresses	1 – 126

3.2.2 Hardware

3.2.2.1 PROFIBUS-DP Fieldbus Interface (X3)

The following table and figure provide pinout information for this interface.



SIGNAL:	PIN:
NC	1
NC	2
PB_+	3
PB_ENA	4
PB_GND	5
PB_+5V	6
NC	7
PB_–	8
NC	9

3.2.2.2 Address Switch

The address switch has no function for the PROFIBUS master. The fieldbus address is set via WAGO-I/O-*PRO* CAA control configuration.

3.2.2.3 LEDs

ERR:	STA:	Description:
Off	Off	PROFIBUS master is not configured PROFIBUS master has not received the token from the PROFIBUS network
Off	On	PROFIBUS master holds the token and is ready to send telegrams
On	Off	PROFIBUS master has a short circuit
On	On	Communication problem between the PROFIBUS master and at least one PROFIBUS-DP slave
Off	Blinks (0.5 to 100 Hz)	PROFIBUS master exchanges the token with another master in the PROFIBUS network

3.2.3 Configuring the First Project with PROFIBUS

The PROFIBUS network must first be configured before the application can access it:

1. The devices are configured directly in the WAGO-I/O-PRO *CAA* programming software. To do so, click on “PLC configuration” in the resources tree structure.

2. Add the PROFIBUS master to the control configuration. To do so, click the right mouse button on “PLC Configuration” and select “Append Subelement” → “EC1-DEB-DPM...”

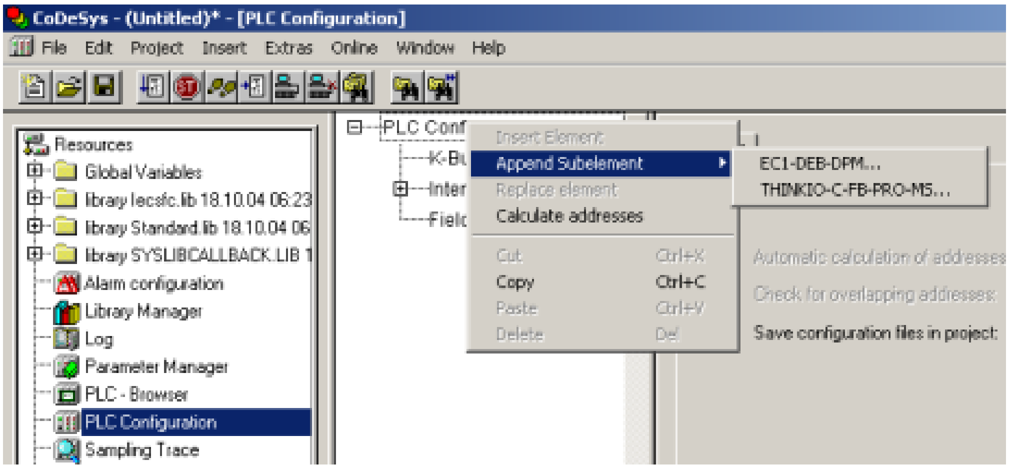


Fig. 3-24: Adding the PROFIBUS master p287033d

Adding PROFIBUS slaves (750-343 modules are shown in this illustration) as “Append Subelement” is done in a similar manner.

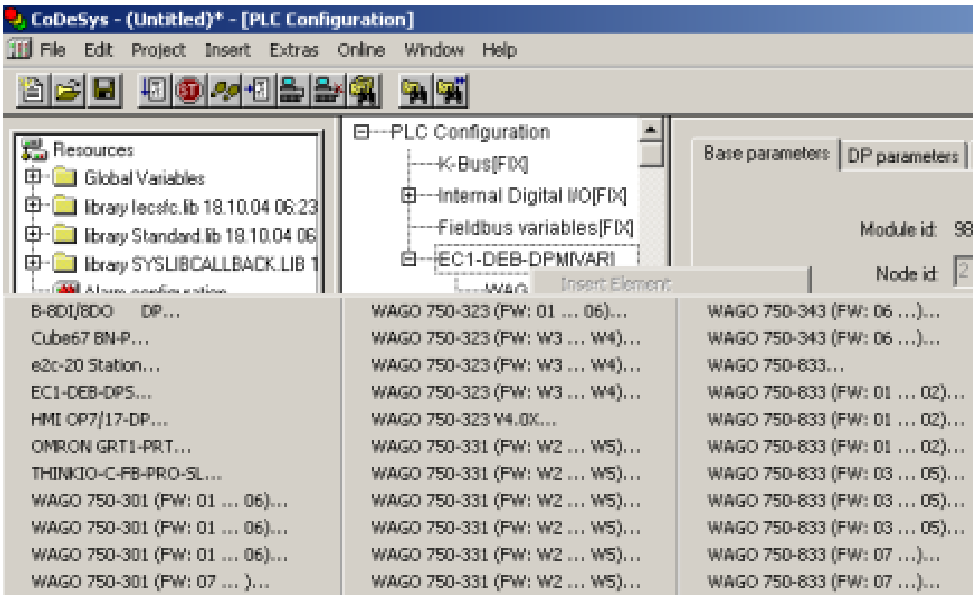


Fig. 3-25: Adding the PROFIBUS slaves p287034d

Note:
The GSD files of current WAGO-I/O-System components are automatically integrated into the target files of the WAGO-I/O-IPC 758-870/000-001. To connect devices that are not from WAGO, the appropriate GSD files must be used. This is done with the menu item Extras/Add configuration file.

3. In the configuration tree, click on the coupler 750-343.

4. Click on the “Inputs/Outputs” tab. Transfer the I/O modules of the PROFIBUS node from the selection list to the configuration according to their physical order from left to right.

Note:
Before selecting the node’s modules, the coupler’s first entry must be “750-343 No Pi Channel” (which is found under “empty modules”).

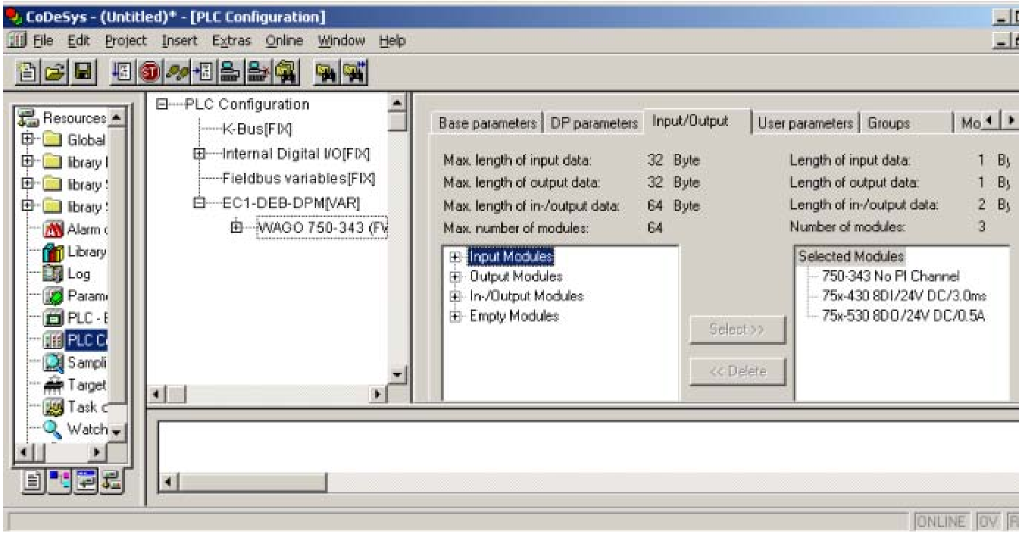


Fig. 3-26: Selecting the I/O modules p287035d

Note:
I/O modules without process data (e.g., supply modules, end modules, etc.) are irrelevant when configuring the PROFIBUS and are not included in the GSD file selection list.

5. Click on "WAGO 750-343" in the navigation area.

6. Click on the "Base parameter" tab to see the default addresses of the process data in the node. No changes need to be done here for the example program.

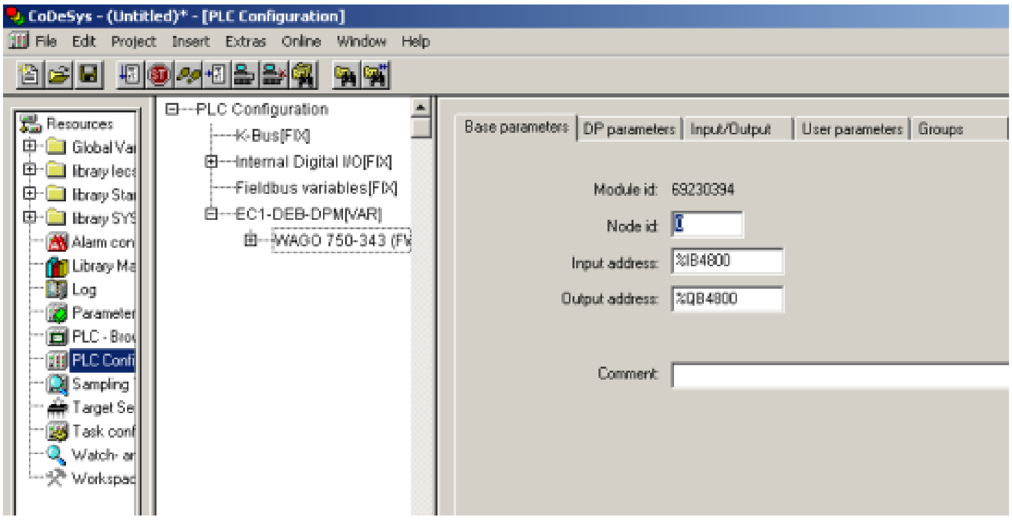


Fig. 3-27: Basic parameters, address allocation p287036d

7. Click on the "DP Parameters" tab to enter the PROFIBUS station address. Entry "2" is automatically generated for the first node. This entry must match the node address set on the PROFIBUS slave DIP switch.

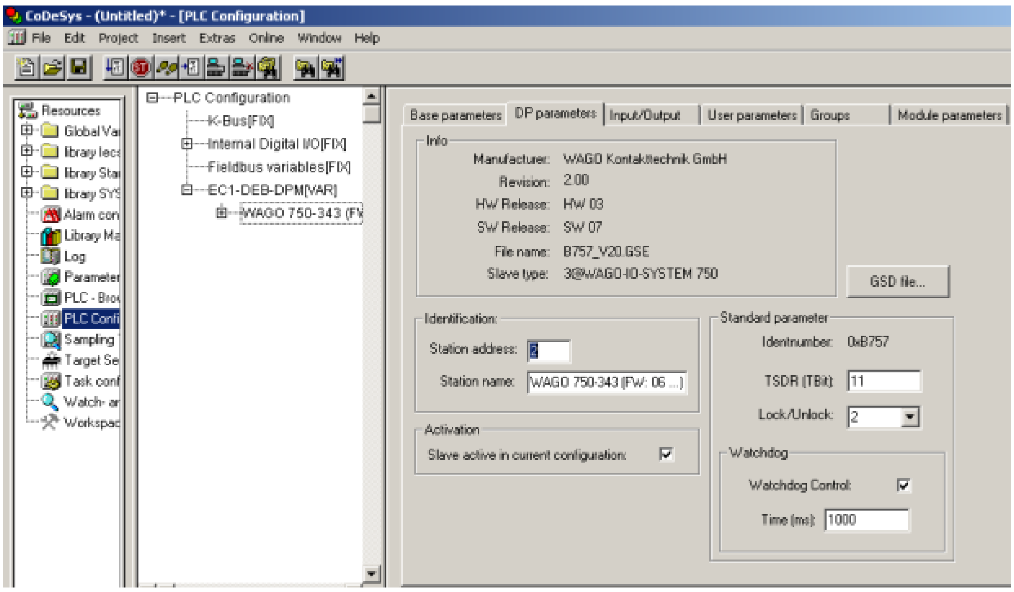


Fig. 3-28: DP parameters p287037d

The basic parameters of the PROFIBUS node have now been set.

3.2.4 Accessing the PROFIBUS Data

Using a small example program, the process image of the WAGO-IO-IPC 758-870/000-001 will look like the following (byte declaration):

PB_Input_Byte AT %IB4800: BYTE;

PB_Output_Byte AT %QB4800: BYTE;

The following picture shows a simple assignment program. The inputs are directly assigned to the outputs.

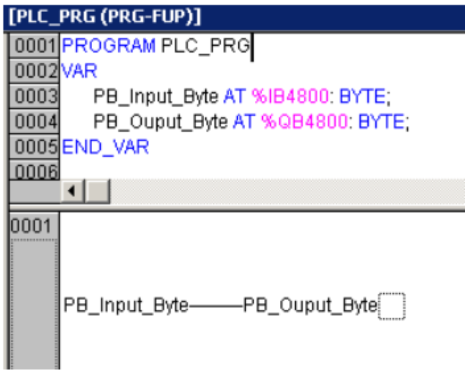


Fig. 3-29: Declaration and statement sections of the PLC-PRG in FBD p287038d

If the test program can be compiled without any problems, it can be loaded into the WAGO-I/O-IPC 758-870/000-001.

To compile, click on “Project” in the menu bar and select “Compile all.”

PLC program download to the 758-870/000-001

Please verify that the simulation mode is deactivated.

1. Click “Online” in the menu and select “Communication parameter” in order to create a new communication channel.

2. Click on the “**New**” button.
   The “Communication parameter: New channel” window will open.

3. Select “TCP/IP” and click on “**OK**.”

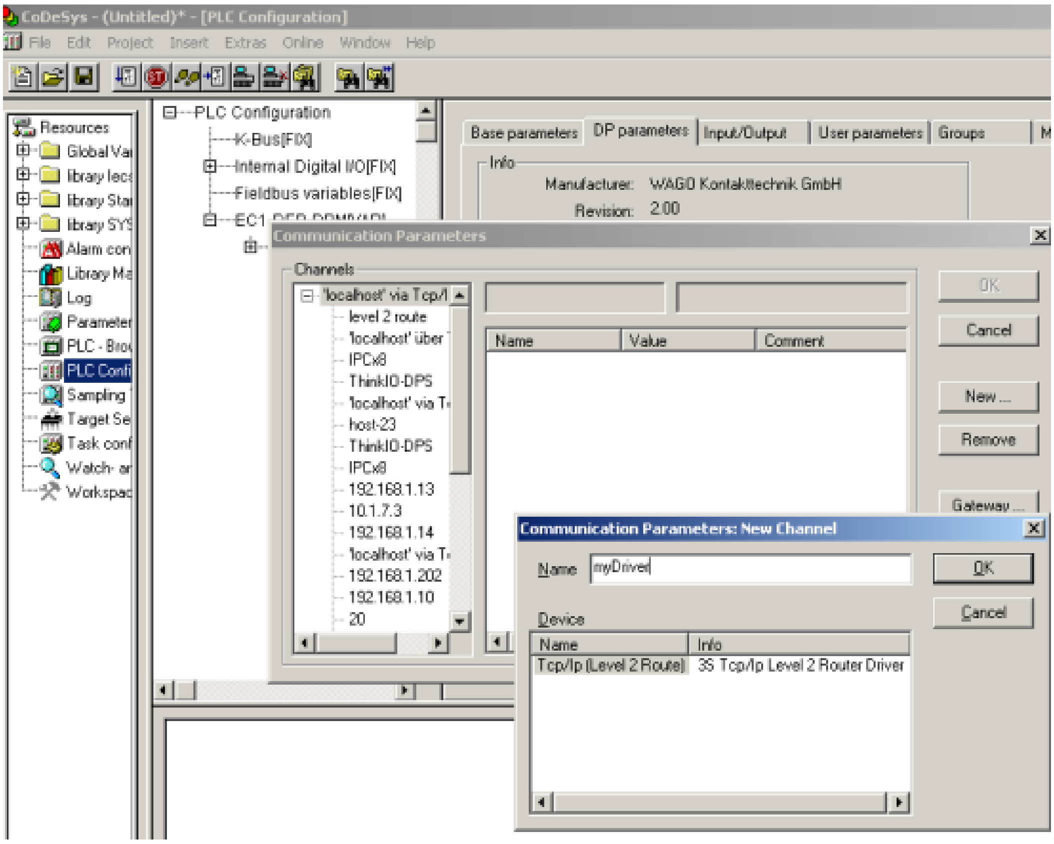


Fig. 3-30: Communication parameter: New channel p287039d

4. Enter the IP address of the coupler (e.g., 192.168.1.2) under "Address."
   To do so, double click on the field next to "Address."

5. Enter the value 1200 under "Port."
   To do so, double click on the field next to "Port."

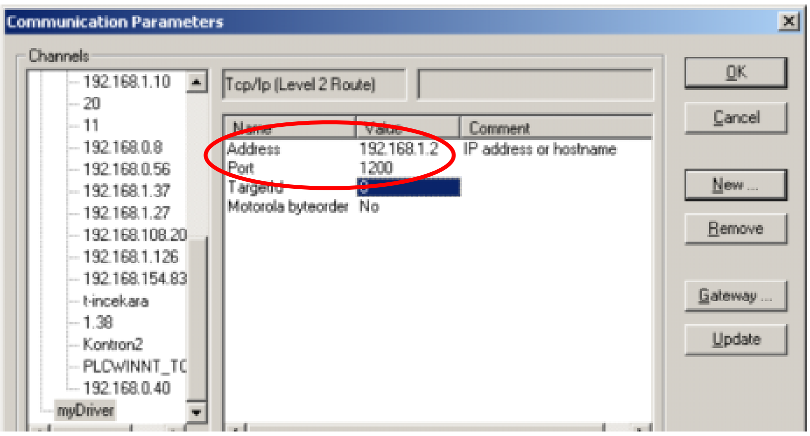


Fig. 3-31: Creating a new communication channel p287040d

6. Transfer the program by clicking "Online" in the menu bar and selecting "Login."

7. To start the program, click on "Online" and select "Start."

Note:
With the demo version of WAGO-I/O-*PRO* CAA, it is not possible to store programs permanently; i.e., the program must be downloaded and restarted after a power failure.

PROFIBUS is automatically initialized when the user program starts.

Note:
If PROFIBUS is stopped (e.g., by pulling the PROFIBUS cable out) the WAGO-I/O-IPC 758-870/000-001 must be reset by using the menu "Online" → "Reset (origin)" — the program will have to be reloaded.

3.3 I/O-IPC 758-870/000-002 and 758-870/000-012

3.3.1 Overview of the Process and Configuration Data

When using the CANopen fieldbus, it is possible to connect up to 125 slaves. In order to support this amount of data, it is essential to meet the guidelines for CANopen networks. Depending on the driver performance of the slave with the lowest performance, a repeater after 110 slaves is needed in order to guarantee the required signal quality.

The memory space in the fieldbus controller chip for the storage of configuration data is limited. 9600 bytes RAM are available for dynamic configuration.

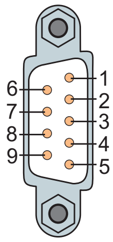
The following table lists the limits for the amount of process and configuration data:

Maximum amount of configuration data per slave	9600 bytes
Maximum amount of configuration data	11446 bytes
Maximum amount of input data from all slaves	3584 bytes
Maximum amount of output data for all slaves	3584 bytes
Available node addresses	1–126

3.3.2 Hardware

3.3.2.1 CANopen Fieldbus Interface (X3)

The following table and figure provide pinout information for this interface.



SIGNAL:	PIN:
NC	1
CAN_–	2
CAN_GND	3
NC	4
NC	5
NC	6
CAN_+	7
NC	8
CAN_+5V	9

3.3.2.2 Address Switch

The address switch has no function for the CANopen master. The fieldbus address is set via WAGO-I/O-*PRO* CAA control configuration.

3.3.2.3 LEDs

ERR:	STA:	Description:
Off	Off	CANopen master is operational and can send or receive telegrams
Off	on	CANopen master sends a telegram
on	-	Communication problem between the CANopen master and at least one CANopen node

3.3.3 Configuring the First Project with CANopen

The CANopen network must first be configured before the application can access it.

1. The devices are configured directly in the programming software WAGO-I/O-*PRO* CAA. To do so, click on "PLC configuration" in the resources tree structure.

2. Add the CANopen master to the control configuration. To do so, click right mouse button on "PLC Configuration" and select "Append CanMaster."

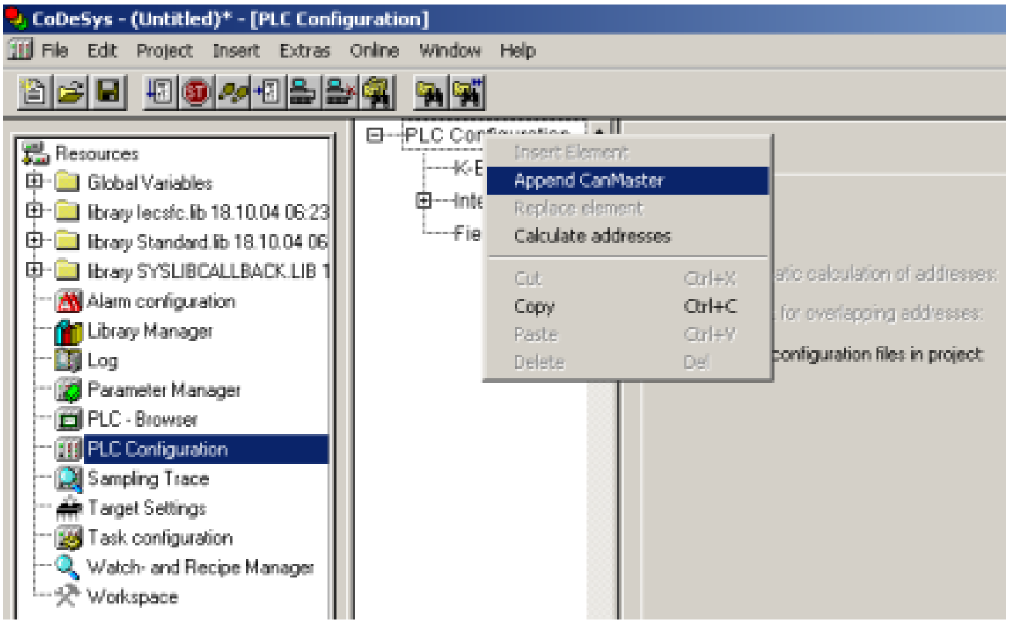


Fig. 3-32: Adding the CANopen master p287040d

3. Adapt the CAN parameters to requirements.

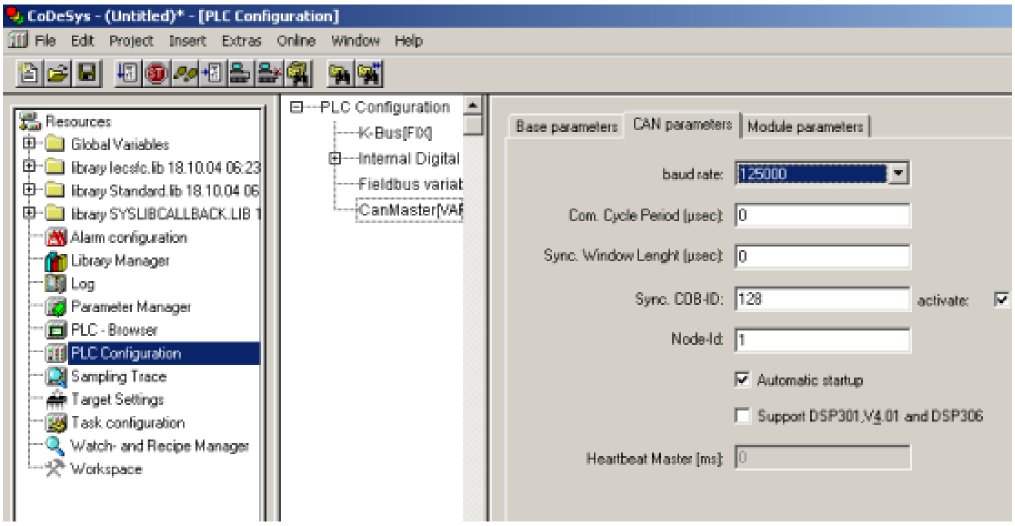


Fig. 3-33: Setting the CAN parameters p287041d

- Com. Cycle Period (µsec):
  Used to set the time interval between two Sync telegrams.
  Enter the value in µsec; however, it is analyzed in milliseconds.
  Therefore, the lower bits have no significance.

- Sync. Window Length (µsec):
  This line has no relevance and is independent of the entries made.

4. Adding CANopen slaves (shown here with 750-348) as "Sub elements" to the CANopen master is done in a similar manner.

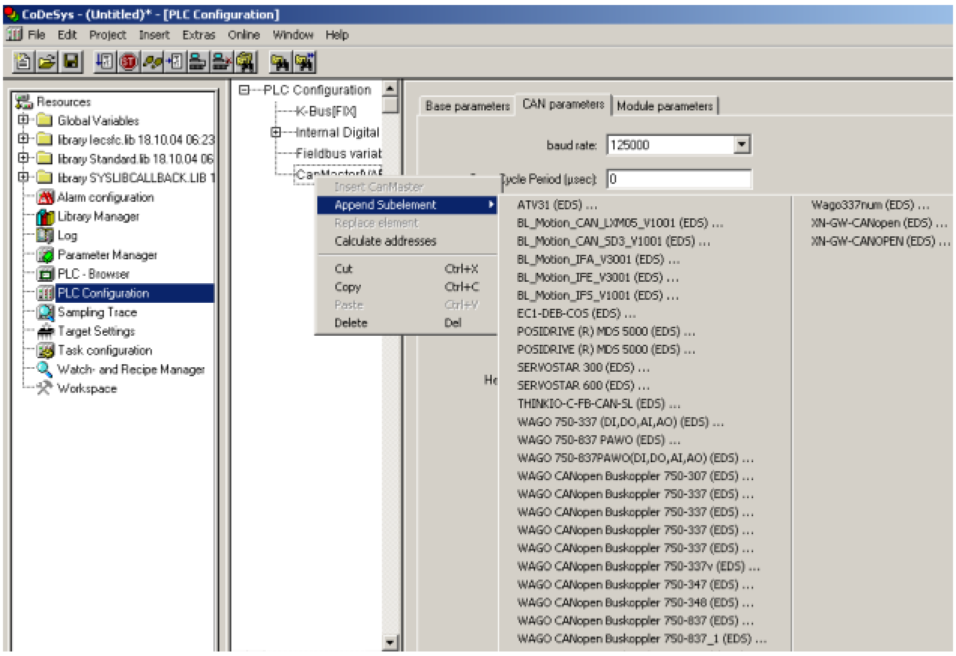


Fig. 3-34: Adding the CANopen slaves p287042d

Note:
The EDS files of current WAGO-I/O-System components are automatically integrated into the target files of the WAGO-I/O-IPC 758-870/000-002. To connect devices that are not from WAGO, the appropriate EDS files must be used. This is done via menu item Extras/Add configuration file.

5. In the configuration tree, click on the CANopen coupler 750-348. Click on the “CAN Parameters” tab. Set the “Node ID,” the “Device Type” and the “Node guarding.”

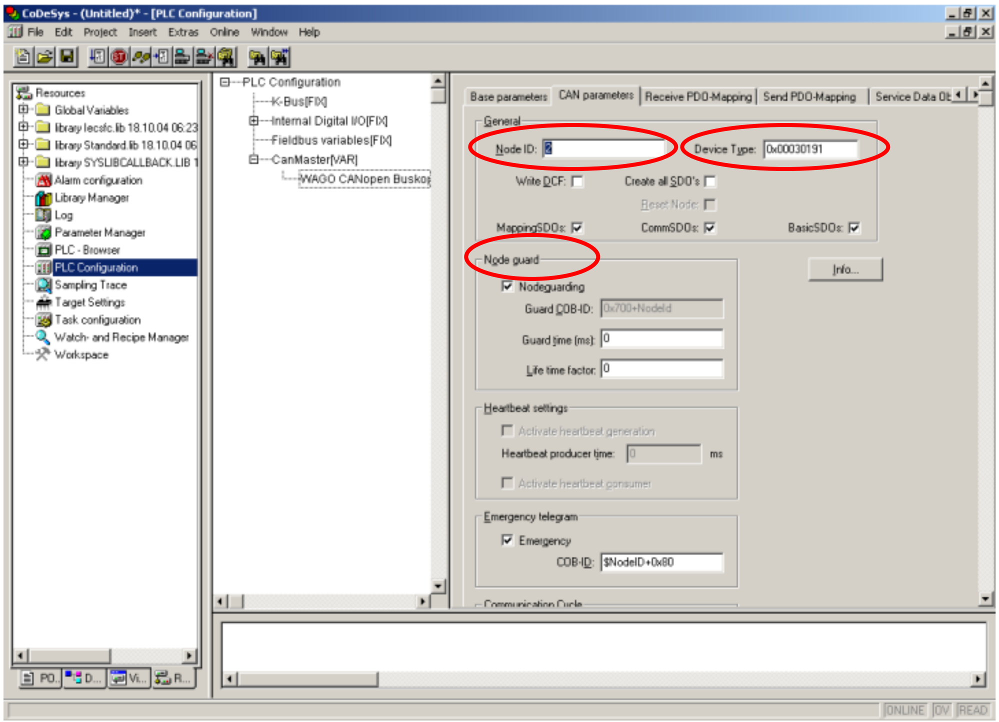


Fig. 3-35: CAN parameters p287043d

- Device Type
  The default value (Object 0x1000) from the EDS file is used. If the default value is different, it must be adapted. Please refer to section 3.3.5 for more information.

The transmission of any of the three following Service Data Objects (SDO) of the slave configuration may be disabled and are not sent by the master:

- MappingSDOs:
  Objects 0x1600–1620
  Objects 1A00–1A20

- CommSDOs:
  Objects 0x1400–1420
  Objects 1800–1820

- BasicsSDOs:
  Objects 0x100C–1017

The “Node ID” must match the node address set with the address switch on a CANopen slave. In the case of modular slaves, such as the WAGO-I/O-SYSTEM 750, the device type depends on the I/O modules (see section 3.3.5).

6. Click on the “Receive PDO Mapping” tab (slave output data) and on the “Send PDO Mapping” tab (slave input data) in order to map the data that is to be transmitted.

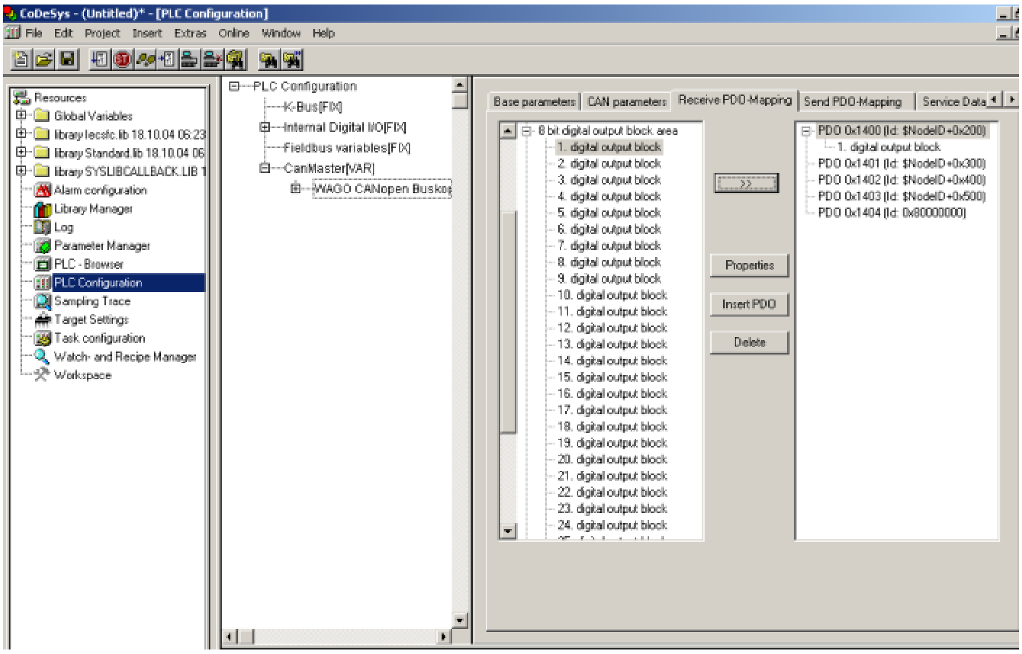


Fig. 3-36: Select output data p287044d

The basic configuration of the CANopen node is now completed.

3.3.4 Accessing CANopen Data

The variables for the CANopen slave are declared in the control configuration. The variable can be addressed in the entire project via this name (global variable).

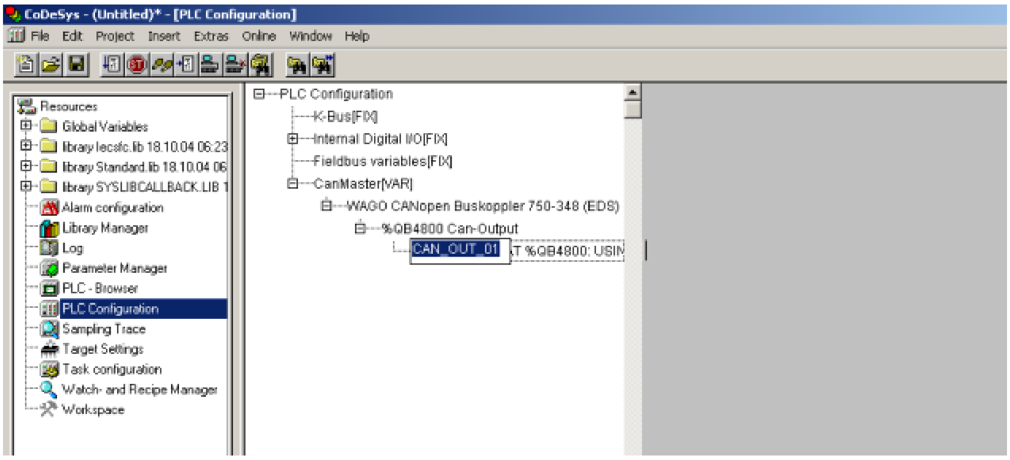


Fig. 3-37: Variable declaration p287045d

A simple assignment can be made in the statement section of the program (see arrow in the following illustration).

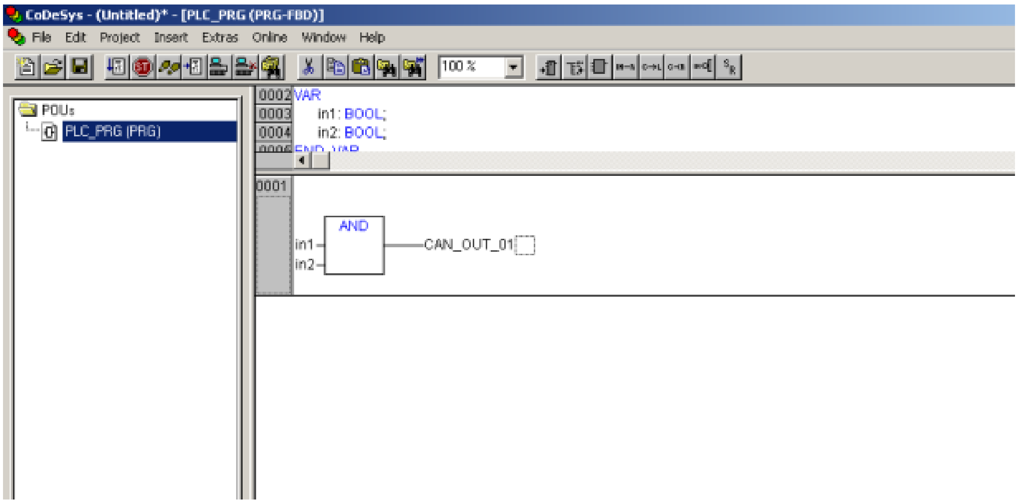


Fig. 3-38: WAGO-I/O-*PRO* CAA, test program 2.

If the test program can be compiled without any problems, it can be loaded into the WAGO-I/O-IPC 758-870/000-002. The CAN bus is started automatically after the download of the user program.

3.3.5 Device Type

During the CAN fieldbus start-up, the CANopen master first reads the device type of the slave. If the value does not match the value of the configuration or the EDS file, the configuration of this slave is aborted.

For slaves, such as the WAGO-I/O-SYSTEM 750, the device type depends on the I/O modules.

Structure:

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

With bit:
1 = 1, if at least 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

Note:
Please find additional information on the device type (Object 0x1000) in the CANopen coupler manual that can be downloaded from the WAGO Web site.

This results in 15 different values for the device type of device profile DS-401:

1. 0x00010191 digital inputs
2. 0x00020191 digital outputs
3. 0x00030191 digital inputs and outputs
4. 0x00040191 analog inputs
5. 0x00050191 digital inputs and analog inputs
6. 0x00060191 digital outputs and analog inputs
7. 0x00070191 digital inputs and outputs and analog inputs
8. 0x00080191 analog outputs
9. 0x00090191 digital inputs and analog outputs
10. 0x000A0191 digital outputs and analog outputs
11. 0x000B0191 digital inputs and outputs and analog outputs
12. 0x000C0191 analog inputs and outputs
13. 0x000D0191 digital inputs and analog inputs and outputs
14. 0x000E0191 digital outputs and analog inputs and outputs
15. 0x000F0191 digital inputs and outputs and analog inputs and outputs

Alternatively, it is also possible to adapt the default value in the EDS file. The EDS files are in the directory *C:\Programs\WAGO Software\CoDeSys V2.3\Targets\WAGO\PLCconf\IO_EC1COM* and can be opened using a text editor (e.g., Notepad). In order to adapt the device type, it is necessary to edit the default value of the object 0x1000.

```
[1000]
SubNumber=
ParameterName=device type
ObjectType=0x07
DataType=0x0007
AccessType=ro
LowLimit=
HighLimit=
DefaultValue=0x00030191
PDOMapping=0
```

3.3.6 Enabling the Analog Inputs

Due to the default value (= FALSE) that is specified by the device profile DS-401-object 0x6423 (Analog Input Global Interrupt Enable), analog input changes are not transmitted. This prevents the CAN bus from being overloaded with CAN messages.
In order to enable the PDO transmission of the analog inputs, the value of object 0x6423 must set to 0x01.

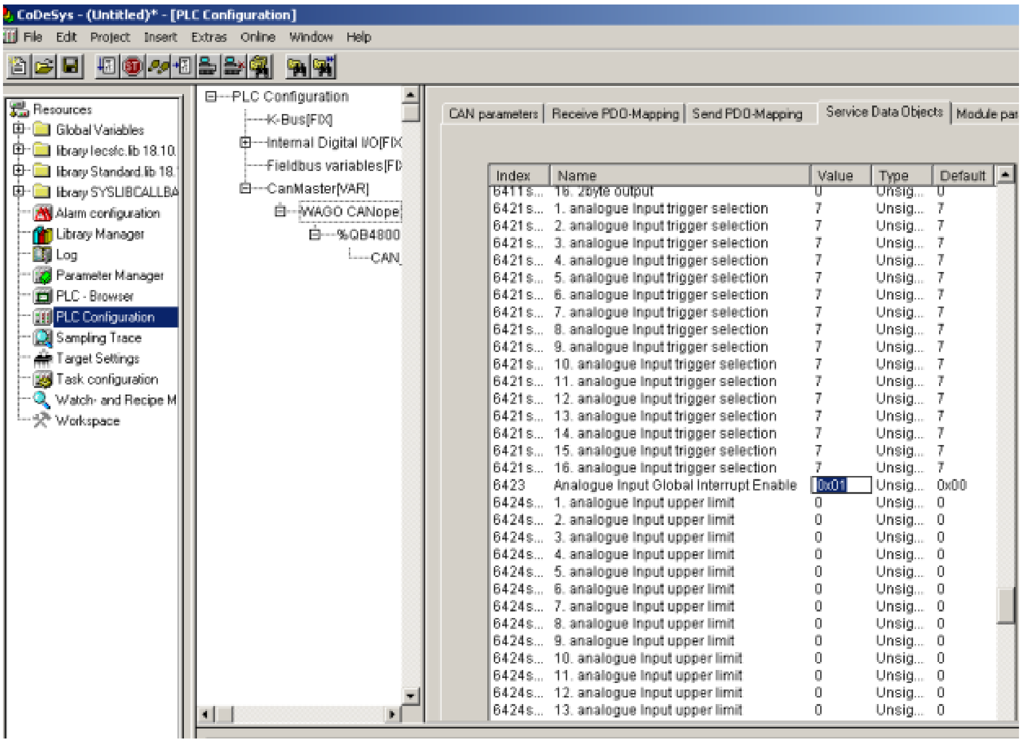


Fig. 3-39: Enabling the analog inputs p287046d

3.4 I/O-IPC 758-870/000-004 and 758-870/000-014

3.4.1 Overview of the Process Data

The maximum amount of data that PROFIBUS slaves can send to and receive from the master is 244 bytes. This makes it essential to meet the guidelines for PROFIBUS networks.

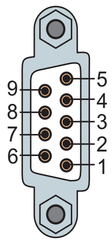
The following table lists the limits for the amount of process data:

Maximum amount of input data from all slaves	244 bytes
Maximum amount of output data for all slaves	244 bytes

3.4.2 Hardware

3.4.2.1 PROFIBUS-DP Fieldbus Interface (X3)

The following table and figure provide pinout information for this interface.



SIGNAL:	PIN:
NC	1
NC	2
PB_+	3
PB_ENA	4
PB_GND	5
PB_+5V	6
NC	7
PB_–	8
NC	9

3.4.2.1.1 Address Switch

The address switch is used to set the fieldbus address for the PROFIBUS slave. When setting the address, make sure that the address in CoDeSys and the address set with the address switch are identical.

3.4.2.2 LEDs

ERR:	STA:	Description:
Off	Off	The PROFIBUS slave does not receive any parameter or configuration data from the PROFIBUS master.
Off	On	The PROFIBUS slave receives parameter and configuration data from the PROFIBUS master.
On	-	CoDeSys is not synchronous with the bus cycle despite the synchronization setting.

4 Fieldbus Master Device Diagnostics

This section requires strong familiarity with the CoDeSys programming tool. It only describes the procedure to create diagnostics using the PROFIBUS master as an example.

The procedure is identical for a CANopen master. Only the EXTENDEDINFO of the DiagGetState() function block is changed. You can find the EXTENDEDINFO for CANopen in section 4.8.

Configured slaves; e.g., a fieldbus coupler or a fieldbus controller, are a prerequisite for diagnostics in fieldbus networks. In order to execute diagnostics, you need to program diagnostic information in CoDeSys. The diagnostic information for the PROFIBUS master is stored in the flag area %MB0–%MB141.

4.1 Description of DiagGetBusState() and DiagGetState()

In order to program diagnostics, you need the following function blocks from the BusDiag.lib library:

- DiagGetBusState() for bus diagnostics
  This function block provides general information for every slave on the bus, such as the number of slaves.

- DiagGetState() for device diagnostics
  This function block provides detailed information on every slave, such as diagnostics.

4.2 Enabling the I/O Module Channels for Diagnostics

In order to execute slave diagnostics (fieldbus coupler or controller), every I/O module must be enabled for diagnostics. This is only required for I/O modules that feature diagnostic functions. To enable the I/O module channels, proceed as follows:

1. In CoDeSys, double click on "System configuration."
2. Click on the slave type (fieldbus coupler or controller).
3. Click on the "Inputs and outputs" tab. Select the I/O modules that provide diagnostic functions. Click on the **SELECT >>** button.
4. Click on the **PROPERTIES** button. The "Module properties" dialog opens.
5. Double click on "disabled" in the "Value" column and enter "enabled."

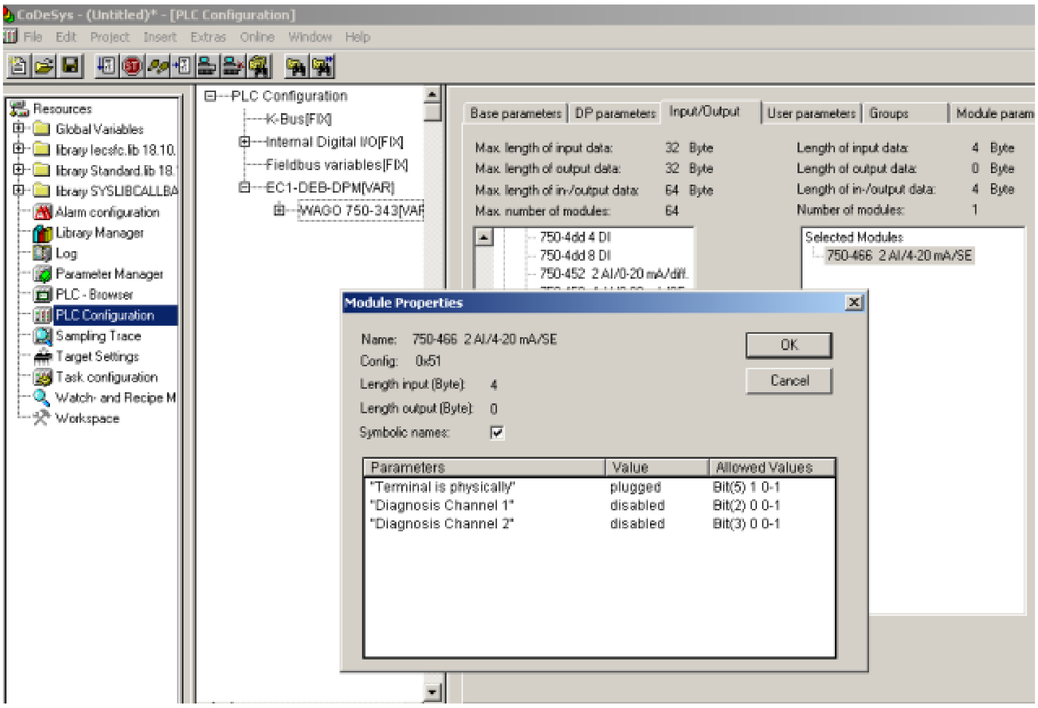


Fig. 4-1: Enabling the I/O Module Channels for Diagnostics p287047d

4.3 Assigning a Diagnostic Address to the PROFIBUS Master

The diagnostic address of the PROFIBUS master needs to be entered in the variable window at “AT%MB0”.

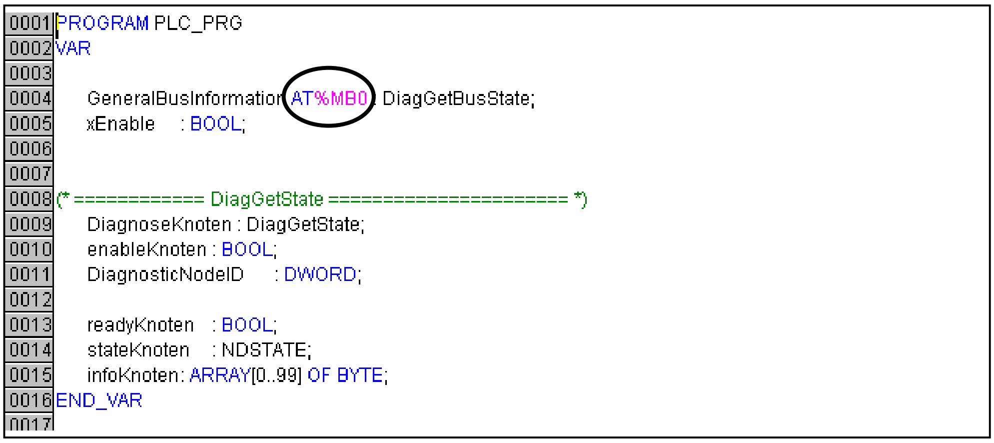


Fig. 4-2: Offline view of the variable window in CoDeSys p287048d

In order to display the diagnostic address, proceed as follows:

1. In the menu item “Control configuration,” click on “EC1-DEB-DPM [VAR].”

2. Select the “Base parameters” tab. The diagnostic address is next to the field “Diagnostic address.” In this example it is %MB0.

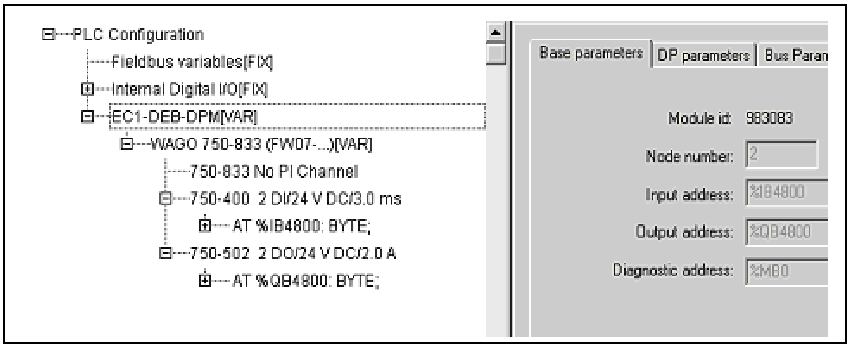


Fig. 4-3: Reading the fieldbus address p287049d

4.4 Creating Diagnostic Functions in CoDeSys 2.7.x

In order to execute bus diagnostics or device diagnostics for the slaves, it is necessary to integrate the BusDiag.lib library into CoDeSys. From this library, call the function block DiagGetBusState() for bus diagnostics and DiagGetState() for device diagnostics. To do so, proceed as follows:

Integrate the BusDiag.lib library into CoDeSys as described below:

1. In the menu bar, click on “Window” and select “Library management.”

2. In the menu bar, click on “Add library” and select BusDiag.lib.

From the BusDiag.lib library, call the DiagGetBusState() function block:

1. In the menu bar, click on the "Component" symbol.

2. Press the **“F2”** key. The “Input help” dialog opens. Click on the option “Standard function block” and select the DiagGetBusState() function block.

3. Create an instance of the DiagGetBusState() function block. To do so, enter a name above the function block. In this example it is “GeneralBus-Information.”

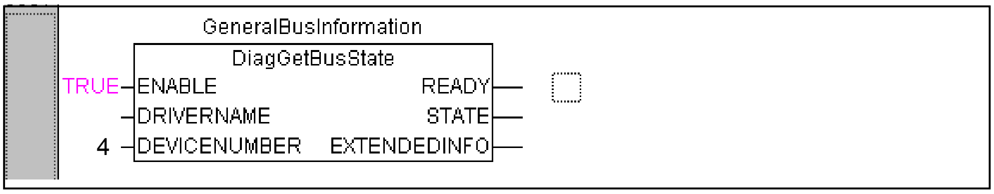


Fig. 4-4: Function block DiagGetBusState() p287050d

From the BusDiag.lib library, call the DiagGetBusState() function block for slave diagnostics:

1. Repeat the steps 1–3 of the previous page.

2. Create an instance of the DiagGetState() function block. In this example it is “DiagnosticsNode.”

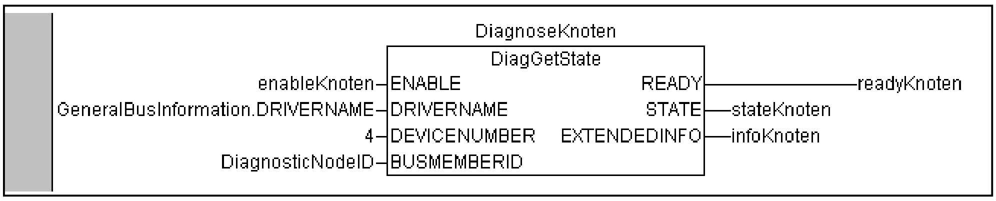


Fig. 4-5: Function block DiagGetState() p287051d

In this example, both function blocks are called during the execution of the program. In order not to extend the cycle time during the execution of the program, do not set the DiagGetState() input “ENABLE” to “TRUE” until diagnostics have been performed.

4.5 Performing Bus Diagnostics Via DiagGetBusState()

To perform bus diagnostics, proceed as follows:

1. Log in to CoDeSys. To do so, click on “Online” in the menu bar and select “Log in.” The variable window displays the information on the variables (online view).

2. Run the PLC program by clicking on “Online” in the menu bar and then clicking on “Start.” The DiagGetBusState() function block is called and the diagnostic information is placed in the EXTENDEDINFO array).

The EXTENDEDINFO array provides information on the PROFIBUS slave status in the online view of the variable window. For every slave, an entry is reserved in the array. The slave address is assigned to the array index. In this example, there are slaves with station addresses 5 and 6 that provide the diagnostic information 5 and 1.

Note:
The diagnostic information is only displayed for the duration of one program cycle. If the diagnostic information must be available for a longer period of time, it is necessary to write the appropriate program.

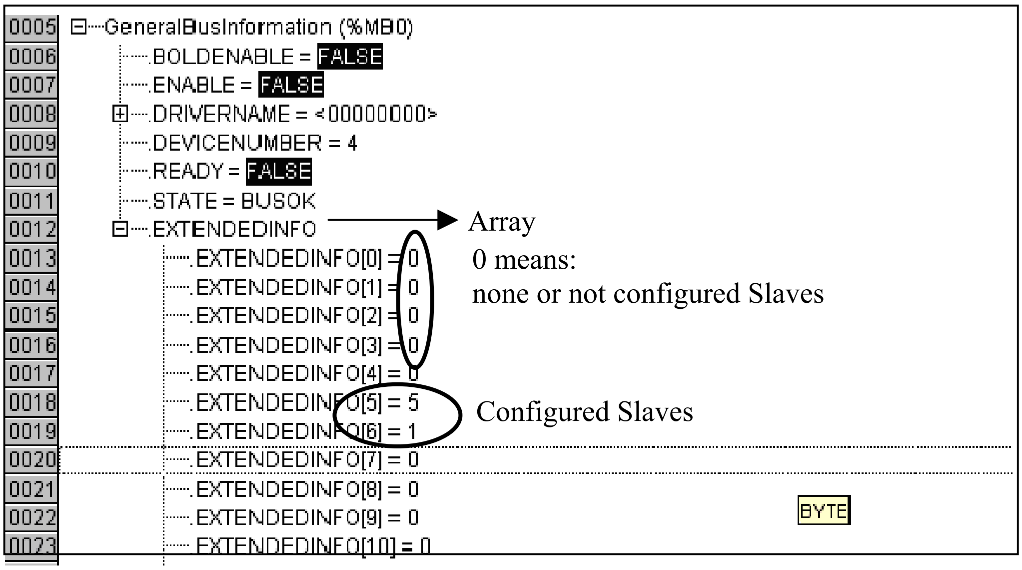


Fig. 4-6: Online view of the variable window(decimal notation) p287052d

Display the diagnostic information of the EXTENDEDINFO array in binary form. To do so, click right mouse button in the variable window and select “binary.” A binary display makes the interpretation of the diagnostic bits easier.

Compare the last three least significant bits of the diagnostic information from the array elements EXTENDEDINFO[x] with the bits listed in the following table.

Example:

The slave with PROFIBUS address 5 provides diagnostic information of the value 5. The decimal digit 5 in binary digits is 101.

1 0 1

Second bit:		First bit:		Zero bit:	
1	0	1	0	1	0
The module can provide diagnostic information.	The module cannot provide diagnostic information.	Bus device is enabled.	Bus device is disabled.	Bus device is configured.	Bus device is not configured.

The slave with PROFIBUS address 5 is configured, disabled and provides diagnostic information.

If an I/O module provides diagnostic information, as it does in this example (bit 2 provides the value 1), then perform device diagnostics. Please see section 4.6.

4.6 Performing Device Diagnostics Via DiagGetState()

If bus diagnostics have shown that an I/O module provides diagnostic information, perform device diagnostics for this slave. Call the DiagGetStatus() function block as follows:

1. Specify the slave that provides the diagnostic information at the input variable BUSMEMBERID. In this example it is the slave with the fieldbus address 5.

2. To read the diagnostic information, set the “ENABLE” input to “TRUE.” The diagnostic information is transferred to the “EXTENDEDINFO” array.

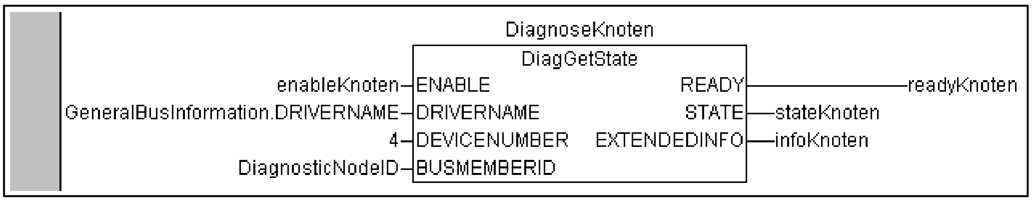


Fig. 4-7: Diagnostic call DiagGetState() p287053d

- DRIVERNAME:
  The input parameter DRIVERNAME is specified by the instance data of the DiagGetBusState function block.

- DEVICENUMBER:
  The DEVICENUMBER of the IPC is always 4.

4.7 Analyzing PROFIBUS Diagnostics of Individual I/O Modules

The array elements [0] to [22] listed in the illustration below are reserved for PROFIBUS standard diagnostics. From array element [23] onward, they include diagnostic information of the WAGO I/O modules.

The amount of diagnostic information per channel of an I/O module (not PROFIsafe) is 3 bytes.

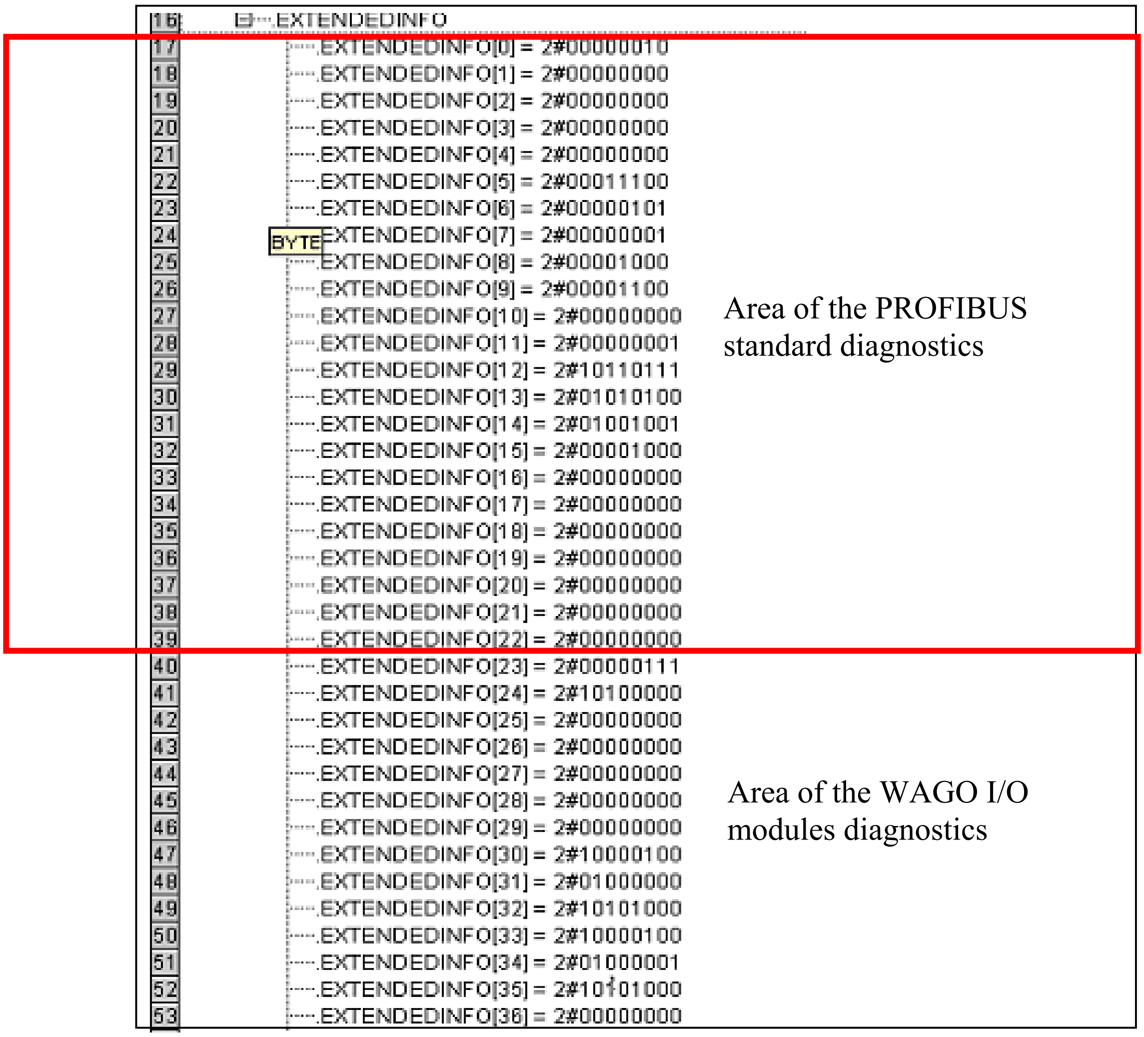


Fig. 4-8: Online view of the EXTENDEDINFO array (binary notation) p287054d

The diagnostic information of the EXTENDEDINFO array is described on the following page.

Description of the Diagnostic Information of the Function Block DiagGetState.EXTENDEDINFO for PROFIBUS

This section describes the EXTENDEDINFO[0–22] of the PROFIBUS standard diagnostics.

EXTENDEDINFO[0]:	Slave address
EXTENDEDINFO[1]:	0 (free)
EXTENDEDINFO[2]:	0 (free)
EXTENDEDINFO[3]:	0 (free)
EXTENDEDINFO[4]:	0 (free)
EXTENDEDINFO[5]:	x, length of the diagnostics
EXTENDEDINFO[6]:	5 (free)
EXTENDEDINFO[7]:	1 (free)
EXTENDEDINFO[8]:	Station status 1
EXTENDEDINFO[9]:	Station status 2
EXTENDEDINFO[10]:	Station status 3
EXTENDEDINFO[11]:	Master address
EXTENDEDINFO[12–13]:	The slave station uses the 2 bytes for its ID number.
EXTENDEDINFO[14…(8+x-1)]:	Ext_Diag_Data (x > 6, for x ≤ 6 no extended information available from the slave)

Station status 1:

Bit 1:	Slave does not respond.
Bit 2:	Slave is not ready.
Bit 3:	Slave is incorrectly parameterized.
Bit 4:	Used for extended diagnostics.
Bit 5:	Slave has detected unknown command.
Bit 6:	Response of the slave is incorrect.
Bit 7:	Last parameter telegram incorrect.
Bit 8:	Slave is parameterized from a different master.

Station status 2:

Bit 1:	Slave must be parameterized
Bit 2:	Slave provides diagnostic information until this bit is zero again.
Bit 3:	1
Bit 4:	Watchdog enabled
Bit 5:	Freeze command enabled
Bit 6:	Sync command enabled
Bit 7:	Reserved
Bit 8:	Slave not designed

Station status 3:

Bit 1:	Reserved
Bit 2:	Reserved
Bit 3:	Reserved
Bit 4:	Reserved
Bit 5:	Reserved
Bit 6:	Reserved
Bit 7:	Reserved
Bit 8:	Slave has more diagnostic data than it can send

Master Address:

This byte contains the master address of the master that has parameterized the slave. If a slave was not parameterized, the value is 255.

Ext_Diag_Data:

This is an extended diagnostic buffer. The values are defined in the manual of the slave station or can be read in the PROFIBUS specification.

Description of the Diagnostic Information for the WAGO I/O MOdules

From array element EXTENDEDINFO[23] onward, diagnostic information of the WAGO I/O modules is present. Decoding of the diagnostic information is explained using I/O module 750-466 with two channels as an example.

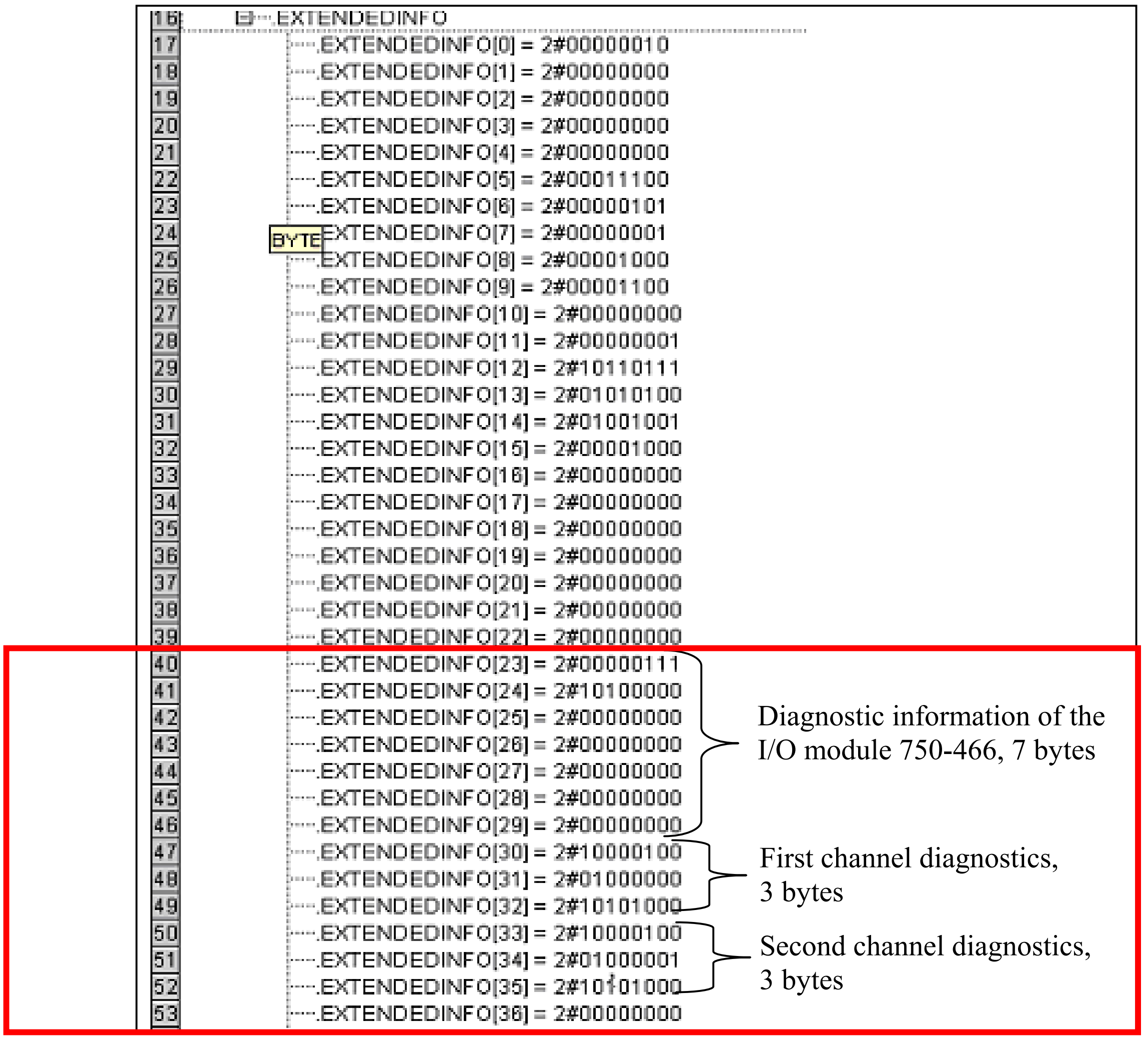


Fig. 4-9: Online view of the EXTENDEDINFO array (binary notation) p287054d

Channel 1 of I/O Module 750-466

Array element:	Content:	Indicates:
30	1000**0100**	Slot for the module. In this example the third I/O module, slot 4. **Note**: The first slot is reserved for the fieldbus coupler and controller including the supply module. The first I/O module is in the second slot.

Array element:	Content:	Indicates:
31	**01**000000	The first two bits indicate the type of channel: **01** Input channel **10** Output channel **11** Input and output channel The other bits indicate the channel number.
32	**101**01000	The first three bits indicate the type of channel: **000** Not assigned **001** 1 bit **010** 2 bit **011** 4 bit **100** 1 byte **101** 1 word **110** 2 words The other bits indicate I/O module errors.
33–35	**Channel 2**	

Channel 2 of I/O module 750-466

Array element:	Content:	Indicates:
33–35	See channel 1	

Examples for Diagnostic Messages of I/O Modules

Item number	Type of channel	Fault type	Indicates:
750-: 462, 465, 466, 472, 474, 479 and 480	101	0.0111' 0.1000' 1.0000' 1.1111'	Value has fallen below lower-limit value. Upper-limit value exceeded. Parameterization error I/O module defective.

For more information on channel-related diagnostics, please refer to the manual for the corresponding I/O module.

4.8 Description of the DiagGetState.EXTENDEDINFO for CANopen

This section describes the EXTENDEDINFO[0–16] of the CANopen standard diagnostics.

EXTENDEDINFO[0]:	Slave address
EXTENDEDINFO[1]:	0 (free)
EXTENDEDINFO[2]:	0 (free)
EXTENDEDINFO[3]:	0 (free)
EXTENDEDINFO[4]:	0 (free)
EXTENDEDINFO[5]:	48, length of the diagnostics
EXTENDEDINFO[6]:	5 (free)
EXTENDEDINFO[7]:	1 (free)
EXTENDEDINFO[8]:	**Station status** Information of the master on the slave
EXTENDEDINFO[9-10]:	**Additional info** Additional information read from the slave OBJ1000H
EXTENDEDINFO[11-12]:	**Device profile** Profile number read from the slave OBJ1000H
EXTENDEDINFO[13]:	**Station status** Feedback of the slave status from the “Node guarding” protocol
EXTENDEDINFO[14]:	**Actual error** Online error information
EXTENDEDINFO[15]:	**Emcy_ Entries** Number of the received upcoming emergency telegrams of the slave
EXTENDEDINFO[16]:	**Emcy_Data**[...]

Emcy_Data[...] – all information per emergency telegram:

EXTENDEDINFO[16]:	Error Code LSB
EXTENDEDINFO[17]:	Error Code MSB
EXTENDEDINFO[18]:	Error Register
EXTENDEDINFO[19]:	Additional Code
EXTENDEDINFO[20]:	Additional Code
EXTENDEDINFO[21]:	Additional Code
EXTENDEDINFO[22]:	Additional Code
EXTENDEDINFO[23]:	Additional Code

Station status:

Bit 1:	Slave does not respond
Bit 2:	Error memory full
Bit 3:	Slave is incorrectly parameterized
Bit 4:	“Node guarding” protocol is enabled
Bit 5:	Reserved
Bit 6:	Reserved
Bit 7:	Reserved
Bit 8:	Slave is disabled

Add Info:

These two bytes are read during start-up of the slave. In the “CAN in Automation” draft standard, these two bytes are described as extended information on the type of slave. For example, in DS-401 (E1 module) these two bytes specify if the slave supports digital inputs or outputs.

Profile number:

These two bytes are read during start-up of the slave. There are several profile numbers that are each described in their own specification. Example:

Device profile for I/O modules: 401

Device profile for drives: 402

Station status:

If the “Node guarding” protocol is enabled for this slave, the status is indicated in the array element EXTENDEDINFO[13].
The following values are specified in the CANopen specification.

Status value:	Description:
1	Disconnected
2	Connecting
3	Preparing
4	Prepared
5	Operational
127	Pre-operational

Actual error:

This byte shows the actual online error information of the slave. See the following table “err_event”:

Err_event:	Indicates:	Error Source:	Help:
30	Guarding failed	Node	Check if node is connected
31	Node has change state and is no longer operational	Node	Reset node
32	Sequence error in guarding protocol	Node	Reset node
33	No response to a configured protocol	Node	Check if node is able to handle remote frames
34	No response of the node during its configuration	Node	Check if node is connected and operational
35	The master configured node profile number is different from the node profile number	Project planning	Check the supported profile number of the node: I/O, encoder, etc.
36	The master configured node device type is different from the node device type	Project planning	Check the supported services of the node
37	Unknown SDO response received	Node	Node not compatible to CiA protocol specification
38	Length indicator of received SDO message unequal 8	Node	Node not compatible to CiA protocol specification
39	Node not handled, node in stop	Device	-

Emcy_Entries:

This byte contains the number of stored emergency telegrams of the following data field.

Emcy_Data:

The emergency telegrams are stored in this field. See the manual of the corresponding slave.

5 I/O Modules

5.1 Overview

All listed bus modules, in the overview below, are available for modular applications with the WAGO-I/O-SYSTEM 750.
For detailed information on the I/O modules and the module variations, please refer to the manuals for the I/O modules.
You will find these manuals on CD ROM „ELECTRONICC Tools and Docs“ (Item No.: 0888-0412) or at http://www.wago.com under Documentation.

More Information
Current information on the modular WAGO-I/O-SYSTEM is available at http://www.wago.com.

5.1.1 Digital Input Modules

Tab. 5-1: Digital input modules

DI DC 5 V	
750-414	4 Channel, DC 5 V, 0.2 ms, 2- to 3-conductor connection, high-side switching
DI DC 5(12) V	
753-434	8 Channel, DC 5(12) V, 0.2 ms, 1-conductor connection, high-side switching
DI DC 24 V	
750-400, 753-400	2 Channel, DC 24 V, 3.0 ms, 2- to 4-conductor connection; high-side switching
750-401, 753-401	2 Channel, DC 24 V, 0.2 ms, 2- to 4-conductor connection; high-side switching
750-410, 753-410	2 Channel, DC 24 V, 3.0 ms, 2- to 4-conductor connection; high-side switching
750-411, 753-411	2 Channel, DC 24 V, 0.2 ms, 2- to 4-conductor connection; high-side switching
750-418, 753-418	2 Channel, DC 24 V, 3.0 ms, 2- to 3-conductor connection; high-side switching; diagnostics and confirmation
750-419	2 Channel, DC 24 V, 3.0 ms, 2- to 3-conductor connection; high-side switching; diagnostics
750-421, 753-421	2 Channel, DC 24 V, 3.0 ms, 2- to 3-conductor connection; high-side switching; diagnostics
750-402, 753-402	4 Channel, DC 24 V, 3.0 ms, 2- to 3-conductor connection; high-side switching

750-432, 753-432	4 Channel, DC 24 V, 3.0 ms, 2-conductor connection; high-side switching
750-403, 753-403	4 Channel, DC 24 V, 0.2 ms, 2- to 3-conductor connection; high-side switching
750-433, 753-433	4 Channel, DC 24 V, 0.2 ms, 2-conductor connection; high-side switching
750-422, 753-422	4 Channel, DC 24 V, 2- to 3-conductor connection; high-side switching; 10 ms pulse extension
750-408, 753-408	4 Channel, DC 24 V, 3.0 ms, 2- to 3-conductor connection; low-side switching
750-409, 753-409	4 Channel, DC 24 V, 0.2 ms, 2- to 3-conductor connection; low-side switching
750-430, 753-430	8 Channel, DC 24 V, 3.0 ms, 1-conductor connection; high-side switching
750-431, 753-431	8 Channel, DC 24 V, 0.2 ms, 1-conductor connection; high-side switching
750-436	8 Channel, DC 24 V, 3.0 ms, 1-conductor connection; low-side switching
750-437	8 Channel, DC 24 V, 0.2 ms, 1-conductor connection; low-side switching
DI AC/DC 24 V	
750-415, 753-415	4 Channel, AC/DC 24 V, 2-conductor connection
750-423, 753-423	4 Channel, AC/DC 24 V, 2- to 3-conductor connection; with power jumper contacts
DI AC/DC 42 V	
750-428, 753-428	4 Channel, AC/DC 42 V, 2-conductor connection
DI DC 48 V	
750-412, 753-412	2 Channel, DC 48 V, 3.0ms, 2- to 4-conductor connection; high-side switching
DI DC 110 V	
750-427, 753-427	2 Channel, DC 110 V, configurable high-side or low-side switching
DI AC 120 V	
750-406, 753-406	2 Channel, AC 120 V, 2- to 4-conductor connection; high-side switching
DI AC 120(230) V	
753-440	4 Channel, AC 120(230) V, 2-conductor connection; high-side switching
DI AC 230 V	
750-405, 753-405	2 Channel, AC 230 V, 2- to 4-conductor connection; high-side switching

DI NAMUR	
750-435	1 Channel, NAMUR EEx i, proximity switch acc. to DIN EN 50227
750-425, 753-425	2 Channel, NAMUR, proximity switch acc. to DIN EN 50227
750-438	2 Channel, NAMUR EEx i, proximity switch acc. to DIN EN 50227
DI Intruder Detection	
750-424, 753-424	2 Channel, DC 24 V, intruder detection

5.1.2 Digital Output Modules

Tab. 5-2: Digital output modules

DO DC 5 V	
750-519	4 Channel, DC 5 V, 20mA, short-circuit-protected; high-side switching
DO DC 12(14) V	
753-534	8 Channel, DC 12(14) V, 1A, short-circuit-protected; high-side switching
DO DC 24 V	
750-501, 753-501	2 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching
750-502, 753-502	2 Channel, DC 24 V, 2.0 A, short-circuit-protected; high-side switching
750-506, 753-506	2 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching; diagnostics
750-507, 753-507	2 Channel, DC 24 V, 2.0 A, short-circuit-protected; high-side switching; diagnostics; no longer available, replaced by 750-508!
750-508	2 Channel, DC 24 V, 2.0 A, short-circuit-protected; high-side switching; diagnostics; replacement for 750-507
750-535	2 Channel, DC 24 V, EEx i, short-circuit-protected; high-side switching
750-504, 753-504	4 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching
750-531, 753-531	4 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching
750-532	4 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching; diagnostics
750-516, 753-516	4 Channel, DC 24 V, 0.5 A, short-circuit-protected; low-side switching
750-530, 753-530	8 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching
750-537	8 Channel, DC 24 V, 0.5 A, short-circuit-protected; high-side switching; diagnostics
750-536	8 Channel, DC 24 V, 0.5 A, short-circuit-protected; low-side switching
DO AC 120(230) V	
753-540	4 Channel, AC 120(230) V, 0.25 A, short-circuit-protected; high-side switching

DO AC/DC 230 V	
750-509, 753-509	2 Channel solid state relay, AC/DC 230 V, 300 mA
750-522	2 Channel solid state relay, AC/DC 230 V, 500 mA, 3 A (< 30 s)
DO Relay	
750-523	1 Channel, AC 230 V, AC 16 A, potential-free, 1 make contact
750-514, 753-514	2 Channel, AC 125 V , AC 0.5 A , DC 30 V, DC 1 A, potential-free, 2 changeover contacts
750-517, 753-517	2 Channel, AC 230 V, 1 A, potential-free, 2 changeover contacts
750-512, 753-512	2 Channel, AC 230 V, DC 30 V, AC/DC 2 A, non-floating, 2 make contacts
750-513, 753-513	2 Channel, AC 230 V, DC 30 V, AC/DC 2 A, potential-free, 2 make contacts

5.1.3 Analog Input Modules

Tab. 5-3: Analog input modules

AI 0 - 20 mA	
750-452, 753-452	2 Channel, 0 - 20 mA, differential input
750-465, 753-465	2 Channel, 0 - 20 mA, single-ended
750-472, 753-472	2-Channel, 0 - 20 mA, 16 bit, single-ended
750-480	2-Channel, 0 - 20 mA , differential input
750-453, 753-453	4 Channel, 0 - 20 mA, single-ended
AI 4 - 20 mA	
750-454, 753-454	2 Channel, 4 - 20 mA, differential input
750-474, 753-474	2 Channel, 4 - 20 mA, 16 bit, single-ended
750-466, 753-466	2 Channel, 4 - 20 mA, single ended
750-485	2 Channel, 4 - 20 mA, EEx i, single-ended
750-492, 753-492	2 Channel, 4 - 20 mA, isolated differential input
750-455, 753-455	4 Channel, 4 - 20 mA, single-ended
AI 0 - 1 A	
750-475, 753-475	2-Channel, 0 - 1 A AC/DC, differential input
AI 0 - 5 A	
750-475/020-000, 753-475/020-000	2-Channel, 0 - 5 A AC/DC, differential input

AI 0 - 10 V	
750-467, 753-467	2 Channel, DC 0 - 10 V, single-ended
750-477, 753-477	2 Channel, AC/DC 0 - 10 V, differential input
750-478, 753-478	2 Channel, DC 0 - 10 V, single-ended
750-459, 753-459	4 Channel, DC 0 - 10 V, single-ended
750-468	4 Channel, DC 0 - 10 V, single-ended
AI DC ± 10 V	
750-456, 753-456	2 Channel, DC ± 10 V, differential input
750-479, 753-479	2 Channel, DC ± 10 V, differential measurement input
750-476, 753-476	2 Channel, DC ± 10 V, single-ended
750-457, 753-457	4 Channel, DC ± 10 V, single-ended
AI DC 0 - 30 V	
750-483, 753-483	2 Channel, DC 0 -30 V, differential measurement input
AI Resistance Sensors	
750-461, 753-461	2 Channel, resistance sensors, PT100 / RTD
750-481/003-000	2 Channel, resistance sensors, PT100 / RTD, EEx i
750-460	4 Channel, resistance sensors, PT100 / RTD
AI Thermocouples	
750-462	2 Channel, thermocouples, line break detection, sensor types: J, K, B, E, N, R, S, T, U
750-469, 753-469	2 Channel, thermocouples, line break detection, sensor types: J, K, B, E, N, R, S, T, U, L
AI Others	
750-491	1 Channel for resistor bridges (strain gauge)

5.1.4 Analog Output Modules

Tab. 5-4: Analog output modules

AO 0 - 20 mA	
750-552, 753-552	2 Channel, 0 - 20 mA
750-585	2 Channel, 0 - 20 mA, EEx i
750-553, 753-553	4 Channel, 0 - 20 mA
AO 4 - 20 mA	
750-554, 753-554	2 Channel, 4 - 20 mA
750-554, 753-554	4 Channel, 4 - 20 mA
AO DC 0 - 10 V	
750-550, 753-550	2 Channel, DC 0 - 10 V
750-560	2 Channel, DC 0 - 10 V, 10 bit, 100 mW, 24 V
750-559, 753-559	4 Channel, DC 0 - 10 V
AO DC ± 10 V	
750-556, 753-556	2 Channel, DC ± 10 V
750-557, 753-557	4 Channel, DC ± 10 V

5.1.5 Special Modules

Tab. 5-5: Special modules

Counter Modules	
750-404, 753-404	Up / down counter, DC 24 V, 100 kHz
750-638, 753-638	2 Channel, up / down counter, DC 24 V/ 16 bit / 500 Hz
Frequency Measuring	
750-404/000-003, 753-404/000-003	Frequency measuring
Pulse Width Module	
750-511	2-channel pulse width module, DC 24 V, short-circuit-protected, high-side switching
Distance and Angle Measurement Modules	
750-630	SSI transmitter interface
750-631	Incremental encor interface, differential inputs
750-634	Incremental encor interface, DC 24 V
750-637	Incremental encor interface RS 422, cam outputs
750-635, 753-635	Digital pulse interface, for magnetostrictive distance sensors
Serial Interfaces	
750-650, 753	Serial interface RS 232 C
750-653, 753	Serial interface RS 485
750-651	TTY-Serial interface, 20 mA Current Loop
750-654	Data exchange module
DALI / DSI Master Module	
750-641	DALI / DSI master module
AS interface Master Module	
750-655	AS interface master module
Radio Receiver Module	
750-642	Radio receiver EnOcean
MP Bus Master Module	
750-643	MP bus (multi point bus) master module
Vibration Monitoring	
750-645	2 Channel vibration velocity / bearing condition monitoring VIB I/O

PROFIsafe Modules	
750-660/000-001	8FDI 24V DC PROFIsafe; PROFIsafe 8 channel digital input module
750-665/000-001	4FDO 0.5A / 4FDI 24V DC PROFIsafe; PROFIsafe 4 channel digital input and output module
750-666/000-001	1FDO 10A / 2FDO 0.5A / 2FDI 24V PROFIsafe; PROFIsafe power switch module
RTC Module	
750-640	RTC module
KNX / EIB TP1 Module	
750-646	KNX / EIB /TP1 module – device mode / router mode

5.1.6 System Modules

Tab. 5-6: System modules

Module Bus Extension	
750-627	Module bus extension, end module
750-628	Module bus extension, coupler module
DC 24 V Power Supply Modules	
750-602	DC 24 V, passive
750-601	DC 24 V, max. 6.3 A, without diagnostics, with fuse-holder
750-610	DC 24 V, max. 6.3 A, with diagnostics, with fuse-holder
750-625	DC 24 V, EEx i, with fuse-holder
DC 24 V Power Supply Modules with bus power supply	
750-613	Bus power supply, 24 V DC
AC 120 V Power Supply Modules	
750-615	AC 120 V, max. 6.3 A without diagnostics, with fuse-holder
AC 230 V Power Supply Modules	
750-612	AC/DC 230 V without diagnostics, passive
750-609	AC 230 V, max. 6.3 A without diagnostics, with fuse-holder
750-611	AC 230 V, max. 6.3 A with diagnostics, with fuse-holder
Filter Modules	
750-624	Filter module, field side power supply
750-626	Filter module, system and field side power supply
Field Side Connection Module	
750-603, 753-603	Field side connection module, DC 24 V
750-604, 753-604	Field side connection module, DC 0 V
750-614, 753-614	Field side connection module, AC/DC 0 ... 230 V
Separation Modules	
750-616	Separation module
750-621	Separation module with power contacts
Binary Spacer Module	
750-622	Binary spacer module
End Module	
750-600	End module, to loop the internal bus

5.2 Process Data Architecture for MODBUS/TCP

With some I/O modules, the structure of the process data is fieldbus specific.

In the case of a coupler/controller with MODBUS/TCP, the process image uses a word structure (with word alignment). The internal mapping method for data greater than one byte conforms to the Intel format.

The following section describes the process image for various WAGO-I/O-SYSTEM 750 and 753 I/O modules when using a coupler/controller with MODBUS/TCP.

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

For the PFC process image of the programmable fieldbus controller is the the structure of the process data mapping identical.

5.2.1 Digital Input Modules

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

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

Some digital modules have an additional diagnostic bit per channel in the Input Process Image. The diagnostic bit is used for detecting faults that occur (e.g., wire breaks and/or short circuits).

1 Channel Digital Input Module with Diagnostics

750-435

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

2 Channel Digital Input Modules

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

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

2 Channel Digital Input Modules with Diagnostics

750-419, -421, -424, -425, 753-421, -424, -425

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

2 Channel Digital Input Module with Diagnostics and Output Process Data

750-418, 753-418

The 750-418, 753-418 digital input module supplies a diagnostic and acknowledge bit for each input channel. If a fault condition occurs, the diagnostic bit is set. After the fault condition is cleared, an acknowledge bit must be set to re-activate the input. The diagnostic data and input data bit is mapped in the Input Process Image, while the acknowledge bit is in the Output Process Image.

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

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Acknowledgement bit Q 2 Channel 2	Acknowledgement bit Q 1 Channel 1	0	0

4 Channel Digital Input Modules

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

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

8 Channel Digital Input Modules

750-430, -431, -436, -437, 753-430, -431, -434

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

5.2.2 Digital Output Modules

Digital output modules use one bit of data per channel to control the output of the corresponding channel. These bits are mapped into the Output Process Image.

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

1 Channel Digital Output Module with Input Process Data

750-523

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						not used	Status bit „Manual Operation“

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						not used	controls DO 1 Channel 1

2 Channel Digital Output Modules

750-501, -502, -509, -512, -513, -514, -517, -535, (and all variations),
753-501, -502, -509, -512, -513, -514, -517

Output Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						controls DO 2 Channel 2	controls DO 1 Channel 1

2 Channel Digital Input Modules with Diagnostics and Input Process Data

750-507 (-508), -522, 753-507

The 750-507 (-508), -522 and 753-507 digital output modules have a diagnostic bit for each output channel. When an output fault condition occurs (i.e., overload, short circuit, or broken wire), a diagnostic bit is set. The diagnostic data is mapped into the Input Process Image, while the output control bits are in the Output Process Image.

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnostic bit S 2 Channel 2	Diagnostic bit S 1 Channel 1

Output Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						controls DO 2 Channel 2	controls DO 1 Channel 1

750-506, 753-506

The 750-506, 753-506 digital output module has 2-bits of diagnostic information for each output channel. The 2-bit diagnostic information can then be decoded to determine the exact fault condition of the module (i.e., overload, a short circuit, or a broken wire). The 4-bits of diagnostic data are mapped into the Input Process Image, while the output control bits are in the Output Process Image.

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnostic bit S 3 Channel 2	Diagnostic bit S 2 Channel 2	Diagnostic bit S 1 Channel 1	Diagnostic bit S 0 Channel 1

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				not used	not used	controls DO 2 Channel 2	controls DO 1 Channel 1

4 Channel Digital Output Modules

750-504, -516, -519, -531, 753-504, -516, -531, -540

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

4 Channel Digital Output Modules with Diagnostics and Input Process Data

750-532

The 750-532 digital output modules have a diagnostic bit for each output channel. When an output fault condition occurs (i.e., overload, short circuit, or broken wire), a diagnostic bit is set. The diagnostic data is mapped into the Input Process Image, while the output control bits are in the Output Process Image.

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnos-tic bit S 3 Channel 4	Diagnos-tic bit S 2 Channel 3	Diagnos-tic bit S 1 Channel 2	Diagnos-tic bit S 0 Channel 1

Diagnostic bit S = '0' no Error
Diagnostic bit S = '1' overload, short circuit, or broken wire

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

8 Channel Digital Output Module

750-530, -536, 753-530, -434

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
controls DO 8 Channel 8	controls DO 7 Channel 7	controls DO 6 Channel 6	controls DO 5 Channel 5	controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

8 Channel Digital Output Modules with Diagnostics and Input Process Data

750-537

The 750-537 digital output modules have a diagnostic bit for each output channel. When an output fault condition occurs (i.e., overload, short circuit, or broken wire), a diagnostic bit is set. The diagnostic data is mapped into the Input Process Image, while the output control bits are in the Output Process Image.

Input Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Diagnostic bit S 7 Channel 8	Diagnostic bit S 6 Channel 7	Diagnostic bit S 5 Channel 6	Diagnostic bit S 4 Channel 5	Diagnostic bit S 3 Channel 4	Diagnostic bit S 2 Channel 3	Diagnostic bit S 1 Channel 2	Diagnostic bit S 0 Channel 1

Diagnostic bit S = '0' no Error
Diagnostic bit S = '1' overload, short circuit, or broken wire

Output Process Image							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
controls DO 8 Channel 8	controls DO 7 Channel 7	controls DO 6 Channel 6	controls DO 5 Channel 5	controls DO 4 Channel 4	controls DO 3 Channel 3	controls DO 2 Channel 2	controls DO 1 Channel 1

5.2.3 Analog Input Modules

The hardware of an analog input module has 16 bits of measured analog data per channel and 8 bits of control/status. However, the coupler/controller with MODBUS/TCP does not have access to the 8 control/status bits. Therefore, the coupler/controller with MODBUS/TCP can only access the 16 bits of analog data per channel, which are grouped as words and mapped in Intel format in the Input Process Image.

When digital input modules are also present in the node, the analog input data is always mapped into the Input Process Image in front of the digital data.

1 Channel Analog Input Module

750-491, (and all variations)

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Measured Value U_D
1	D3	D2	Measured Value U_{ref}

2 Channel Analog Input Modules

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, -476, -477, -478, -479, -480, -481, -483, -485, -492, (and all variations),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, -476, -477, -478, -479, -483, -492, (and all variations)

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Measured Value Channel 1
1	D3	D2	Measured Value Channel 2

4 Channel Analog Input Modules

750-453, -455, -457, -459, -460, -468, (and all variations),
753-453, -455, -457, -459

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Measured Value Channel 1
1	D3	D2	Measured Value Channel 2
2	D5	D4	Measured Value Channel 3
3	D7	D6	Measured Value Channel 4

5.2.4 Analog Output Modules

The hardware of an analog output module has 16 bits of measured analog data per channel and 8 bits of control/status. However, the coupler/controller with MODBUS/TCP does not have access to the 8 control/status bits. Therefore, the coupler/controller with MODBUS/TCP can only access the 16 bits of analog data per channel, which are grouped as words and mapped in Intel format in the Output Process Image.

When digital output modules are also present in the node, the analog output data is always mapped into the Output Process Image in front of the digital data.

2 Channel Analog Output Modules

750-550, -552, -554, -556, -560, -585, (and all variations),
753-550, -552, -554, -556

Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Output Value Channel 1
1	D3	D2	Output Value Channel 2

4 Channel Analog Output Modules

750-553, -555, -557, -559, 753-553, -555, -557, -559

Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Output Value Channel 1
1	D3	D2	Output Value Channel 2
2	D5	D4	Output Value Channel 3
3	D7	D6	Output Value Channel 4

5.2.5 Specialty Modules

WAGO has a host of Specialty I/O modules that perform various functions. With individual modules beside the data bytes also the control/status byte is mapped in the process image. The control/status byte is required for the bi-directional data exchange of the module with the higher-ranking control system. The control byte is transmitted from the control system to the module and the status byte from the module to the control system.
This allows, for example, setting of a counter with the control byte or displaying of overshooting or undershooting of the range with the status byte.

Further information
For detailed information about the structure of a particular module's control/status byte, please refer to that module's manual. Manuals for each module can be found on the Internet under:
http://www.wago.com.

Counter Modules

750-404, (and all variations except of /000-005),
753-404, (and variation /000-003)

The above Counter Modules have a total of 5 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 1 byte of control/status). The counter value is supplied as 32 bits. The following tables illustrate the Input and Output Process Image, which has a total of 3 words mapped into each image. Word alignment is applied.

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	S	Status byte
1	D1	D0	Counter Value
2	D3	D2	

Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	C	Control byte
1	D1	D0	Counter Setting Value
2	D3	D2	

750-404/000-005

The above Counter Modules have a total of 5 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 1 byte of control/status). The two counter values are supplied as 16 bits. The following tables illustrate the Input and Output Process Image, which has a total of 3 words mapped into each image. Word alignment is applied.

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	S	Status byte
1	D1	D0	Counter Value of Counter 1
2	D3	D2	Counter Value of Counter 2

Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	C	Control byte
1	D1	D0	Counter Setting Value of Counter 1
2	D3	D2	Counter Setting Value of Counter 2

750-638, 753-638

The above Counter Modules have a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of counter data and 2 bytes of control/status). The two counter values are supplied as 16 bits. The following tables illustrate the Input and Output Process Image, which has a total of 4 words mapped into each image. Word alignment is applied.

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	S0	Status byte of Counter 1
1	D1	D0	Counter Value of Counter 1
2	-	S1	Status byte of Counter 2
3	D3	D2	Counter Value of Counter 2

Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	C0	Control byte of Counter 1
1	D1	D0	Counter Setting Value of Counter 1
2	-	C1	Control byte of Counter 2
3	D3	D2	Counter Setting Value of Counter 2

Pulse Width Modules

750-511, (and all variations)

The above Pulse Width modules have a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of channel data and 2 bytes of control/status). The two channel values are supplied as 16 bits. Each channel has its own control/status byte. The following table illustrates the Input and Output Process Image, which has a total of 4 words mapped into each image. Word alignment is applied.

Input and Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	C0/S0	Control/Status byte of Channel 1
1	D1	D0	Data Value of Channel 1
2	-	C1/S1	Control/Status byte of Channel 2
3	D3	D2	Data Value of Channel 2

Serial Interface Modules with alternative Data Format

750-650, (and the variations /000-002, -004, -006, -009, -010, -011, -012, -013)
750-651, (and the variations /000-002, -003)
750-653, (and the variations /000-002, -007)

Note:
With the freely parametrizable variations /003 000 of the serial interface modules, the desired operation mode can be set. Dependent on it, the process image of these modules is then the same, as from the appropriate variation.

The above Serial Interface Modules with alternative data format have a total of 4 bytes of user data in both the Input and Output Process Image (3 bytes of serial data and 1 byte of control/status). The following table illustrates the Input and Output Process Image, which have a total of 2 words mapped into each image. Word alignment is applied.

Input and Output Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	D0	C/S	Data byte	Control/Status byte
1	D2	D1	Data bytes	

Serial Interface Modules with Standard Data Format

750-650/000-001, -014, -015, -016
750-651/000-001
750-653/000-001, -006

The above Serial Interface Modules with Standard Data Format have a total of 6 bytes of user data in both the Input and Output Process Image (5 bytes of serial data and 1 byte of control/status). The following table illustrates the Input and Output Process Image, which have a total of 3 words mapped into each image. Word alignment is applied.

Input and Output Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	D0	C/S	Data byte	Control/Status byte
1	D2	D1	Data bytes	
2	D4	D3		

Data Exchange Module

750-654, (and the variation /000-001)

The Data Exchange modules have a total of 4 bytes of user data in both the Input and Output Process Image. The following tables illustrate the Input and Output Process Image, which has a total of 2 words mapped into each image. Word alignment is applied.

Input and Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Data bytes
1	D3	D2	

SSI Transmitter Interface Modules

750-630, (and all variations)

The above SSI Transmitter Interface modules have a total of 4 bytes of user data in the Input Process Image, which has 2 words mapped into the image. Word alignment is applied.

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	D1	D0	Data bytes
1	D3	D2	

Incremental Encoder Interface Modules

750-631

The above Incremental Encoder Interface modules have 5 bytes of input data and 3 bytes of output data. The following tables illustrate the Input and Output Process Image, which have 4 words into each image. Word alignment is applied.

Input Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	-	S	not used	Status byte
1	D1	D0	Counter word	
2	-	-	not used	
3	D4	D3	Latch word	

Output Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	-	C	not used	Control byte
1	D1	D0	Counter Setting word	
2	-	-	not used	
3	-	-	not used	

750-634

The above Incremental Encoder Interface module has 5 bytes of input data (6 bytes in cycle duration measurement mode) and 3 bytes of output data. The following tables illustrate the Input and Output Process Image, which has 4 words mapped into each image. Word alignment is applied.

Input Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	-	S	not used	Status byte
1	D1	D0	Counter word	
2	-	(D2)*)	not used	(Periodic time)
3	D4	D3	Latch word	

*) If cycle duration measurement mode is enabled in the control byte, the cycle duration is given as a 24-bit value that is stored in D2 together with D3/D4.

Output Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	-	C	not used	Control byte
1	D1	D0	Counter Setting word	
2	-	-	not used	
3	-	-		

750-637

The above Incremental Encoder Interface Module has a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of encoder data and 2 bytes of control/status). The following table illustrates the Input and Output Process Image, which have 4 words mapped into each image. Word alignment is applied.

Input and Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	C0/S0	Control/Status byte of Channel 1
1	D1	D0	Data Value of Channel 1
2	-	C1/S1	Control/Status byte of Channel 2
3	D3	D2	Data Value of Channel 2

750-635, 753-635

The above Digital Pulse Interface module has a total of 4 bytes of user data in both the Input and Output Process Image (3 bytes of module data and 1 byte of control/status). The following table illustrates the Input and Output Process Image, which have 2 words mapped into each image. Word alignment is applied.

Input and Output Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	D0	C0/S0	Data byte	Control/Status byte
1	D2	D1	Data bytes	

RTC Module

750-640

The RTC Module module has a total of 6 bytes of user data in both the Input and Output Process Image (4 bytes of module data and 1 byte of control/status and 1 byte ID for command). The following table illustrates the Input and Output Process Image, which have 3 words mapped into each image. Word alignment is applied.

Input and Output Process Image				
Offset	Byte Destination		Remark	
	High Byte	Low Byte		
0	ID	C/S	Command byte	Control/Status byte
1	D1	D0	Data bytes	
2	D3	D2		

DALI/DSI Master Module

750-641

The DALI/DSI Master module has a total of 6 bytes of user data in both the Input and Output Process Image (5 bytes of module data and 1 byte of control/status). The following tables illustrate the Input and Output Process Image, which have 3 words mapped into each image. Word alignment is applied.

Input Process Image				
Offset	**Byte Destination**		**Remark**	
	High Byte	**Low Byte**		
0	D0	S	DALI Response	Status byte
1	D2	D1	Message 3	DALI Address
3	D4	D3	Message 1	Message 2

Output Process Image				
Offset	**Byte Destination**		**Remark**	
	High Byte	**Low Byte**		
0	D0	C	DALI command, DSI dimming value	Control byte
1	D2	D1	Parameter 2	DALI Address
3	D4	D3	Command-Extension	Parameter 1

EnOcean Radio Receiver

750-642

The EnOcean radio receiver has a total of 4 bytes of user data in both the Input and Output Process Image (3 bytes of module data and 1 byte of control/status). The following tables illustrate the Input and Output Process Image, which have 2 words mapped into each image. Word alignment is applied.

Input Process Image				
Offset	**Byte Destination**		**Remark**	
	High Byte	**Low Byte**		
0	D0	S	Data byte	Status byte
1	D2	D1	Data bytes	

Output Process Image				
Offset	**Byte Destination**		**Remark**	
	High Byte	**Low Byte**		
0	-	C	not used	Control byte
1	-	-	not used	

MP Bus Master Module

750-643

The MP Bus Master Module has a total of 8 bytes of user data in both the Input and Output Process Image (6 bytes of module data and 2 bytes of control/status). The following table illustrates the Input and Output Process Image, which have 4 words mapped into each image. Word alignment is applied.

<table>
<tr><th colspan="5">Input and Output Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2" colspan="2">Remark</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>C1/S1</td><td>C0/S0</td><td>extended Control/Status byte</td><td>Control/Status byte</td></tr>
<tr><td>1</td><td>D1</td><td>D0</td><td rowspan="3" colspan="2">Data bytes</td></tr>
<tr><td>2</td><td>D3</td><td>D2</td></tr>
<tr><td>3</td><td>D5</td><td>D4</td></tr>
</table>

Vibration Velocity/Bearing Condition Monitoring VIB I/O

750-645

The Vibration Velocity/Bearing Condition Monitoring VIB I/O has a total of 12 bytes of user data in both the Input and Output Process Image (8 bytes of module data and 4 bytes of control/status). The following table illustrates the Input and Output Process Image, which have 8 words mapped into each image. Word alignment is applied.

<table>
<tr><th colspan="5">Input and Output Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">byte Destination</th><th rowspan="2" colspan="2">Remark</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>-</td><td>C0/S0</td><td>Not used</td><td>Control/Status byte (log. Channel 1, Sensor input 1)</td></tr>
<tr><td>1</td><td>D1</td><td>D0</td><td colspan="2">Data bytes (log. Channel 1, Sensor input 1)</td></tr>
<tr><td>2</td><td>-</td><td>C1/S1</td><td>Not used</td><td>Control/Status byte (log. Channel 2 Sensor input 2)</td></tr>
<tr><td>3</td><td>D3</td><td>D2</td><td colspan="2">Data bytes (log. Channel 2 Sensor input 2)</td></tr>
<tr><td>4</td><td>-</td><td>C2/S2</td><td>Not used</td><td>Control/Status byte (log. Channel 3 Sensor input 1)</td></tr>
<tr><td>5</td><td>D5</td><td>D4</td><td colspan="2">Data bytes (log. Channel 3 Sensor input 1)</td></tr>
<tr><td>6</td><td>-</td><td>C3/S3</td><td>Not used</td><td>Control/Status byte (log. Channel 4 Sensor input 2)</td></tr>
<tr><td>7</td><td>D7</td><td>D6</td><td colspan="2">Data bytes (log. Channel 4 Sensor input 2)</td></tr>
</table>

KNX/EIB/TP1 Module

753-646

The KNX/TP1 module appears in router and device mode with a total of 24-byte user data within the input and output area of the process image, 20 data bytes and 2 control/status bytes. Even though the additional bytes S1 or C1 are transferred as data bytes, they are used as extended status and control bytes. The opcode is used for the read/write command of data and the triggering of specific functions of the KNX/EIB/TP1 module. Word-alignment is used to assign 12 words in the process image. Access to the process image is not possible in router mode. Telegrams can only be tunneled.

In device mode, access to the KNX data can only be performed via special function blocks of the IEC application. Configuration using the ETS engineering tool software is required for KNX.

Input Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	S0	Status byte
1	S1	OP	extended Status byte / Opcode
2	D1	D0	Data byte 0 / 1
3	D3	D2	Data byte 2 / 3
4	D5	D4	Data byte 4 / 5
5	D7	D6	Data byte 6 / 7
6	D9	D8	Data byte 8 / 9
7	D11	D10	Data byte 10 / 11
8	D13	D12	Data byte 12 / 13
9	D15	D14	Data byte 14 / 15
10	D17	D16	Data byte 16 / 17
11	D19	D18	Data byte 18 / 19

Output Process Image			
Offset	Byte Destination		Remark
	High Byte	Low Byte	
0	-	C0	Control byte
1	C1	OP	extended Control byte / Opcode
2	D1	D0	Data byte 0 / 1
3	D3	D2	Data byte 2 / 3
4	D5	D4	Data byte 4 / 5
5	D7	D6	Data byte 6 / 7
6	D9	D8	Data byte 8 / 9
7	D11	D10	Data byte 10 / 11
8	D13	D12	Data byte 12 / 13
9	D15	D14	Data byte 14 / 15
10	D17	D16	Data byte 16 / 17
11	D19	D18	Data byte 18 / 19

AS-interface Master Module

750-655

The length of the process image of the AS-interface master module can be set to fixed sizes of 12, 20, 24, 32, 40 or 48 bytes.
It consists of a control or status byte, a mailbox with a size of 0, 6, 10, 12 or 18 bytes and the AS-interface process data, which can range from 0 to 32 bytes.

The AS-interface master module has a total of 6 to maximally 24 words data in both the Input and Output Process Image. Word alignment is applied.

The first Input and output word, which is assigned to an AS-interface master module, contains the status / control byte and one empty byte.
Subsequently the mailbox data are mapped, when the mailbox is permanently superimposed (Mode 1).

In the operating mode with suppressable mailbox (Mode 2), the mailbox and the cyclical process data are mapped next.
The following words contain the remaining process data.

<table>
<tr><th colspan="5">Input and Output Process Image</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Byte Destination</th><th rowspan="2" colspan="2">Remark</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>-</td><td>C0/S0</td><td>not used</td><td>Control/Status byte</td></tr>
<tr><td>1</td><td>D1</td><td>D0</td><td rowspan="5" colspan="2">Mailbox (0, 3, 5, 6 or 9 words) / Process data (0-16 words)</td></tr>
<tr><td>2</td><td>D3</td><td>D2</td></tr>
<tr><td>3</td><td>D5</td><td>D4</td></tr>
<tr><td>...</td><td>...</td><td>...</td></tr>
<tr><td>max. 23</td><td>D45</td><td>D44</td></tr>
</table>

5.2.6 System Modules

System Modules with Diagnostics

750-610, -611

The 750-610 and 750-611 Supply Modules provide 2 bits of diagnostics in the Input Process Image for monitoring of the internal power supply.

Input Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnostic bit S 2 Fuse	Diagnostic bit S 1 Voltage

Binary Space Module

750-622

The Binary Space Modules 750-622 behave alternatively like 2 channel digital input modules or output modules and seize depending upon the selected settings 1, 2, 3 or 4 bits per channel. According to this, 2, 4, 6 or 8 bits are occupied then either in the process input or the process output image.

Input or Output Process Image							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
(Data bit DI 8)	(Data bit DI 7)	(Data bit DI 6)	(Data bit DI 5)	(Data bit DI 4)	(Data bit DI 3)	Data bit DI 2	Data bit DI 1

6 Fieldbus Communication

6.1 ETHERNET

6.1.1 General

ETHERNET is a technology, which has been proven and established as an effective means of data transmission in the field of information technology and office communication. Within a short time ETHERNET has also made a successful breakthrough in the area of private PC networks throughout the world.

This technology was developed in 1972 by Dr. Robert M. Metcalfe, David R. Boggs, Charles Thacker, Butler W. Lampson, and Xerox (Stanford, Ct.). Standardization (IEEE 802.3) took place in 1983.

ETHERNET predominantly uses coaxial cables or twisted pair cables as a transmission medium. Connection to ETHERNET, often already existing in networks, (LAN, Internet) is easy and the data exchange at a transmission rate of 10 Mbps or for some couplers/controllers also 100 Mbps is very fast.

ETHERNET has been equipped with higher level communication software in addition to standard IEEE 802.3, such as TCP/IP (Transmission Control Protocol / Internet Protocol) to allow communication between different systems. The TCP/IP protocol stack offers a high degree of reliability for the transmission of information.

In the ETHERNET based (programmable) fieldbus couplers and controllers developed by WAGO, usually various application protocols have been implemented on the basis of the TCP/IP stack.

These protocols allow the user to create applications (master applications) with standardized interfaces and transmit process data via an ETHERNET interface.

In addition to a series of management and diagnostic protocols, fieldbus specific application protocols are implemented for control of the module data, depending upon the coupler or controller, e. g. MODBUS TCP (UDP), EtherNet/IP, BACnet, KNXNET/IP, PROFINET, Powerlink, Sercos III or others.

Information such as the fieldbus node architecture, network statistics and diagnostic information is stored in the ETHERNET (programmable) fieldbus couplers and controllers and can be viewed as HTML pages via a web browser (e.g., Microsoft Internet-Explorer, Netscape Navigator) being served from the HTTP server in the couplers and controllers.

Furthermore, depending on the requirements of the respective industrial application, various settings such as selection of protocols, TCP/IP, internal clock and security configurations can be performed via the web-based management system. However, you can also load web pages you have created yourself into the couplers/controllers, which have an internal file system, using FTP.

The WAGO ETHERNET TCP/IP fieldbus node does not require any additional master components other than a PC with a network card. So, the fieldbus node can be easily connected to local or global networks using the fieldbus connection. Other networking components such as hubs, switches or repeaters can also be used. However, to establish the greatest amount of "determinism" a switch is recommended.

The use of ETHERNET as a fieldbus allows continuous data transmission between the plant floor and the office. Connection of the ETHERNET TCP/IP fieldbus node to the Internet even enables industrial processing data for all types of applications to be called up world-wide. This makes site independent monitoring, visualization, remote maintenance and control of processes possible.

6.1.2 Network Architecture – Principles and Regulations

A simple ETHERNET network is designed on the basis of one PC with a network interface card (NI), one crossover connection cable (if necessary), one ETHERNET fieldbus node and one 24 V DC power supply for the coupler/controller voltage source.

Each fieldbus node consists of a (programmable) fieldbus coupler or controller and a number of needed I/O modules.

Sensors and actuators are connected to the digital or analog I/O modules on the field side. These are used for process signal acquisition or signal output to the process, respectively.

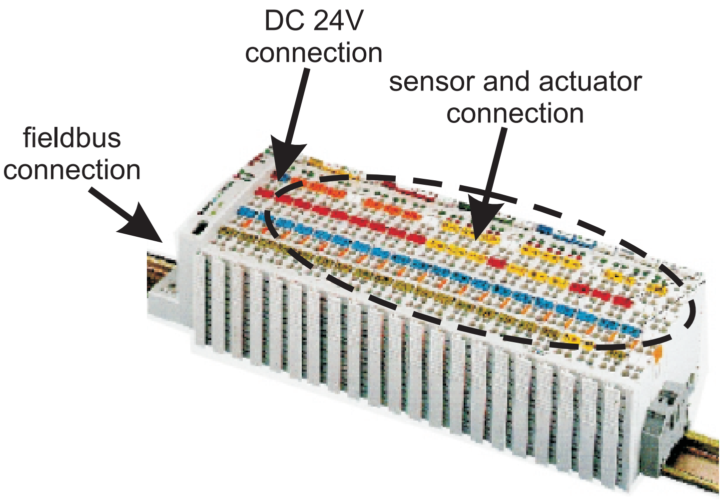


Fig. 6-1. Connection Example and Principle of a Fieldbus Node for a Network Architecture
1Netwerkknotene

Fieldbus communication between master application and (programmable) fieldbus coupler or controller takes place using the implemented fieldbus specific application protocol, e. g. MODBUS TCP (UDP), EtherNet/IP, BACnet, KNXNET/IP, PROFINET, Powerlink, Sercos III or others.

6.1.2.1 Transmission Media

General ETHERNET transmission standards

For transmitting data the ETHERNET standard supports numerous technologies with various parameters (e.g., transmission speed, medium, segment length and type of transmission).

1Base5	Uses a 24 AWG UTP (twisted pair cable) for a 1Mbps baseband signal for distances up to 500 m (250 m per segment) in a physical star topology.
10Base2	Uses a 5 mm 50 Ohm coaxial cable for a 10Mbps baseband signal for distances of up to 185 m in a physical bus topology (often referred to as Thin ETHERNET or ThinNet).
10Base5	Uses a 10 mm 50 Ohm coaxial cable for a 10Mbps baseband signal for distances of up to 500 m in a physical bus topology (often referred to as Thick ETHERNET).
10Base-F	Uses a fiber-optic cable for a 10Mbps baseband signal for distances of up to 4 km in a physical star topology. (There are three sub-specifications: 10Base-FL for fiber-optic link, 10Base-FB for fiber-optic backbone and 10Base-FP for fiber-optic passive).
10Base-T	Uses a 24 AWG UTP or STP/UTP (twisted pair cable) for a 10Mbps baseband signal for distances up to 100 m in a physical star topology.
10Broad36	Uses a 75 Ohm coaxial cable for a 10Mbps baseband signal for distances of up to 1800 m (or 3600 m with double cables) in a physical bus topology.
100BaseTX	Specifies a 100 Mbps transmission with a twisted pair cable of Category 5 and RJ45-connectors. A maximum segment of 100 meters may be used.

Tab. 6-1: ETHERNET Transmission Standards

Beyond that there are still further transmission standards, for example: 100Base-T4 (Fast ETHERNET over twisted conductors), 100Base-FX (Fast ETHERNET over fiber-optic cables) or P802.11 (Wireless LAN) for a wireless transmission.

The media types are shown with their IEEE shorthand identifiers. The IEEE identifiers include three pieces of information.
The first item, for example, “10”, stands for the media.
The third part of the identifier provides a rough indication of segment type or length. For thick coaxial cable, the “5” indicates a 500 meter maximum length allowed for individual thick coaxial segments. For thin coaxial cable, the “2” is rounded up from the 185 meter maximum length for individual thin coaxial segments. The “T” and “F” stand for ‘twisted pair’ and ‘fiber optic’, and simply indicate the cable type.

10Base-T, 100BaseTX

Either the 10BaseT standard or 100BaseTX can be used for the WAGO ETHERNET fieldbus node.
The network architecture is very easy and inexpensive to assemble with S-UTP cable as transmission medium or with cables of STP type.
Both types of cable can be obtained from any computer dealer.

S-UTP cable (screened unshielded twisted pair) is single-shielded cable of Category 5 with overall shield surrounding all twisted unshielded conductor pairs and an impedance of 100 ohm.
STP cable (shielded twisted pair) is cable of Category 5 with stranded and individually shielded conductor pairs; no overall shield is provided.

Wiring of the fieldbus nodes

Maybe, a crossover cable is required for direct connection of a fieldbus node to the network card of the PC.

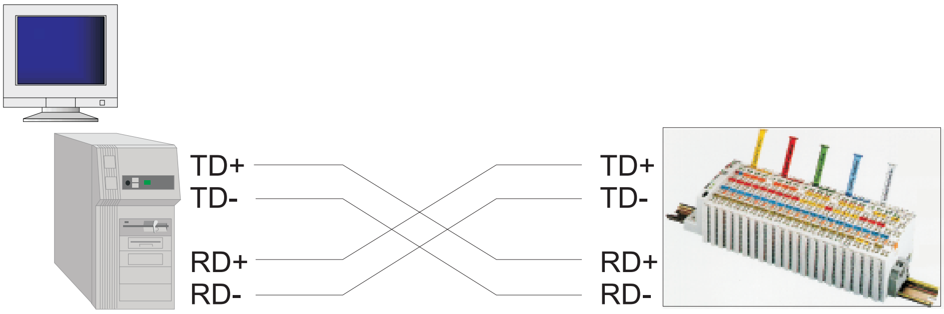


Fig. 6-2: Direct Connection of a Node with Crossover Cable g012906d

If several fieldbus nodes are to be connected to a network card, the fieldbus nodes can be connected via an ETHERNET switch or hub with straight through/parallel cables.

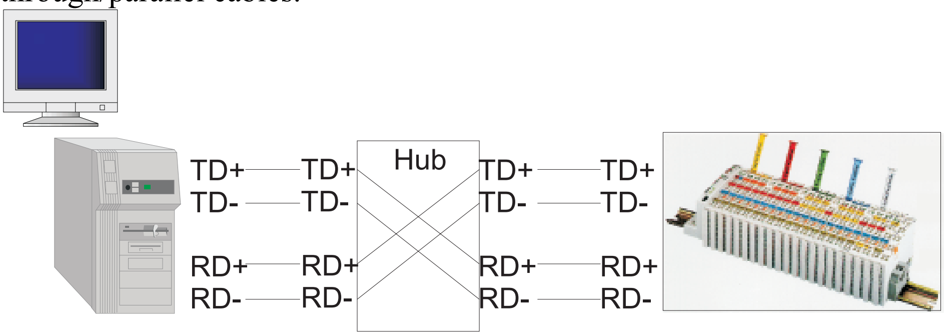


Fig. 6-3: Connection of a Node by means of a Hub with Parallel cables g012908d

An ETHERNET switch is a device that allows all connected devices to transmit and receive data with each other. The switch can also be viewed as a "data traffic cop" where the hub "polices" the data coming in and going out of the individual ports, so the data will only be transmitted to the required node. WAGO recommends using a switch rather then a hub, this will allow for a more deterministic architecture.

Attention
The cable length between the node and the hub cannot be longer than 100 m (328 ft.) without adding signal conditioning systems (i.e., repeaters). Various possibilities are described in the ETHERNET standard for networks covering larger distances.

6.1.2.2 Network Topologies

In the case of 10Base-T, or 100BaseTX several stations (nodes) are connected using a star topology according to the 10Base-T ETHERNET Standard.

Therefore, this manual only deals with the star topology, and the tree topology for larger networks in more detail.

Star Topology

A star topology consists of a network in which all nodes are connected to a central point via individual cables.

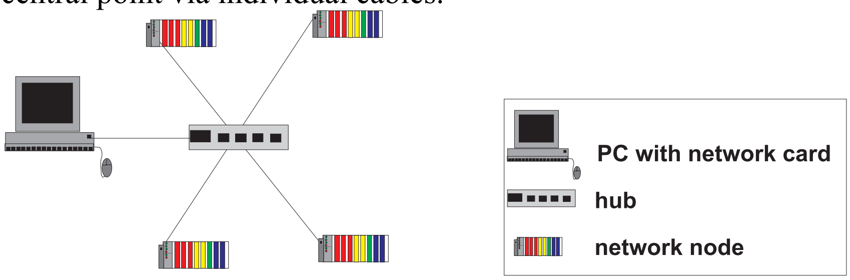


Fig. 6-4: Star Topology G012903e

A star topology offers the advantage of allowing the extension of an existing network. Stations can be added or removed without network interruption. Moreover, in the event of a defective cable, only the network segment and the node connected to this segment is impaired. This considerably increases the fail-safe of the entire network.

Tree Topology

The tree topology combines characteristics of linear bus and star topologies. It consists of groups of star-configured workstations connected to a linear bus backbone cable. Tree topologies allow for the expansion of an existing network, and enables schools, etc. to configure a network to meet their needs.

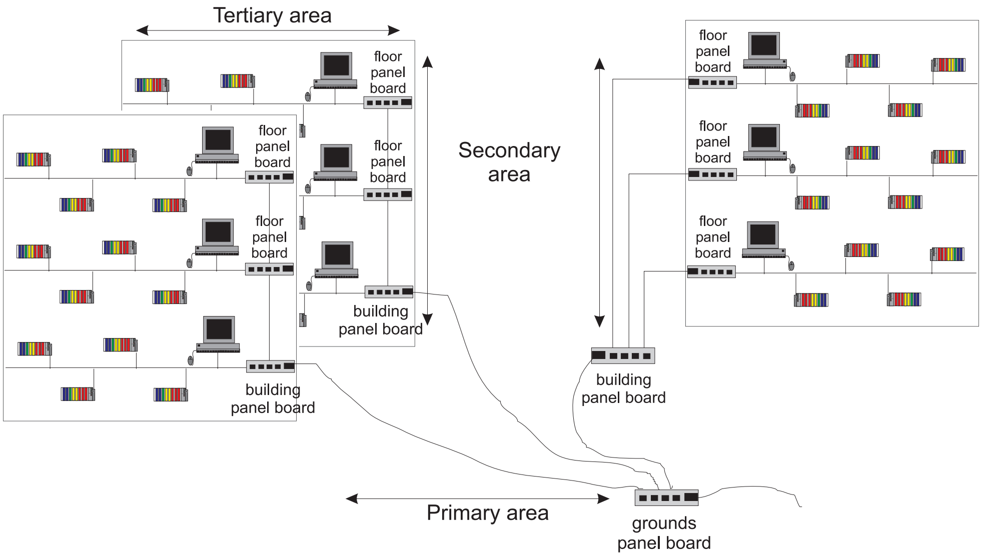


Fig. 6-5: Tree Topology G012904e

5-4-3 Rule

A consideration in setting up a tree topology using ETHERNET protocol is the
5-4-3 rule. One aspect of the ETHERNET protocol requires that a signal sent out on the network cable must reach every part of the network within a specified length of time. Each concentrator or repeater that a signal goes through adds a small amount of time. This leads to the rule that between any two nodes on the network there can only be a maximum of 5 segments connected through 4 repeators/concentrators. In addition, only 3 of the segments may be populated (trunk) segments if they are made of coaxial cable. A populated segment is one that has one or more nodes attached to it. In Figure 5-5, the 5-4-3 rule is adhered to. The furthest two nodes on the network have 4 segments and 3 repeators/concentrators between them.

This rule does not apply to other network protocols or ETHERNET networks where all fiber optic cabling or a combination of a backbone with UTP cabling is used. If there is a combination of fiber optic backbone and UTP cabling, the rule is simply translated to 7-6-5 rule.

Cabling guidelines

"Structured Cabling" specifies general guidelines for network architecture of a LAN, establishing maximum cable lengths for the grounds area, building and floor cabling.

The "Structured Cabling" is standardized in EN 50173, ISO 11801 and TIA 568-A. It forms the basis for a future-orientated, application-independent and cost-effective network infrastructure.

The cabling standards define a domain covering a geographical area of 3 km and for an office area of up to 1 million square meters with 50 to 50,000 terminals. In addition, they describe recommendations for setting up of a cabling system.

Specifications may vary depending on the selected topology, the transmission media and coupler modules used in industrial environments, as well as the use of components from different manufacturers in a network. Therefore, the specifications given here are only intended as recommendations.

6.1.2.3 Coupler Modules

There are a number of hardware modules that allow for flexible arrangement for setting up an ETHERNET network. They also offer important functions, some of which are very similar.

The following table defines and compares these modules and is intended to simplify the correct selection and appropriate application of them.

Module	Characteristics/application	ISO/OSI layer
Repeater	Amplifier for signal regeneration, connection on a physical level.	1
Bridge	Segmentation of networks to increase the length.	2
Switch	Multiport bridge, meaning each port has a separate bridge function. Logically separates network segments, thereby reducing network traffic. Consistent use makes ETHERNET collision-free.	2 (3)
Hub	Used to create star topologies, supports various transmission media, does not prevent any network collisions.	2
Router	Links two or more data networks. Matches topology changes and incompatible packet sizes (e.g. used in industrial and office areas).	3
Gateway	Links two manufacturer-specific networks which use different software and hardware (i.e., ETHERNET and Interbus-Loop).	4-7

Tab. 6-2: Comparison of Coupler Modules for Networks

6.1.2.4 Transmission Mode

Some ETHERNET based WAGO couplers/controllers support both 10Mbit/s and 100Mbit/s for either full or half duplex operation. To guarantee a safe and fast transmission, both these couplers/controllers and their link partners must be configured for the same transmission mode.

Note
A faulty configuration of the transmission mode may result in a link loss condition, a poor network performance or a faulty behavior of the coupler/controller.

The IEEE 802.3u ETHERNET standard defines two possibilities for configuring the transmission modes:

- Static configuration
- Dynamic configuration

6.1.2.4.1 Static Configuration of the Transmission Mode

Using static configuration, both link partners are set to static transmission rate and duplex mode. The following configurations are possible:

- 10 Mbit/s, half duplex
- 10 Mbit/s, full duplex
- 100 Mbit/s, half duplex
- 100 Mbit/s, full duplex

6.1.2.4.2 Dynamic Configuration of the Transmission Mode

The second configuration option is the autonegotiation mode which is defined in the IEEE 802.3u standard. Using this mode, the transmission rate and the duplex mode are negotiated dynamically between both communication partners. Autonegotiation allows the device to automatically select the optimum transmission mode.

Note
To ensure a correct dynamic configuration process, the operation mode for the autonegotiation of both communication partners must be supported and activated.

6.1.2.4.3 Errors Occurring when Configuring the Transmission Mode

Invalid configurations are listed below:

Problem	Cause	Symptoms
Mismatch of the transmission rate	Occurs when configuring one link partner with 10 Mbit/s and the other one with 100 Mbit/s.	Link failure
Duplex mode mismatch	Occurs when one link partner is running in full-duplex and the other in half-duplex mode.	Faulty or discarded data packets as well as collisions on the medium.
Mismatch using autonegotiation	Occurs when one link partner is running in auto-negotiation mode and the other one is using a static configuration of the transmission mode in full-duplex operation.	The link partner, which is in autonegotiation mode, determines the network speed via the parallel detection procedure and sets the duplex mode to half-duplex. If the device is operating in full-duplex mode with static configuration, a duplex mode mismatch will occur (see above).

6.1.2.5 Important Terms

Data security

If an internal network (Intranet) is to be connected to the public network (e.g., the Internet) then data security is an extremely important aspect.

Undesired access can be prevented by a **Firewall**.
Firewalls can be implemented in software or network components. They are interconnected in a similar way to routers as a switching element between Intranets and the public network. Firewalls are able to limit or completely block all access to the other networks, depending on the access direction, the service used and the authenticity of the network user.

Real-time ability

Transmission above the fieldbus system level generally involves relatively large data quantities. The permissible delay times may also be relatively long (0.1...10 seconds).
However, real-time behavior within the fieldbus system level is required for ETHERNET in industry.
In ETHERNET it is possible to meet the real-time requirements by restricting the bus traffic (< 10 %), by using a master-slave principle, or also by implementing a switch instead of a hub.
MODBUS/TCP is a master/slave protocol in which the slaves only respond to commands from the master. When only one master is used, data traffic over the network can be controlled and collisions avoided.

Shared ETHERNET

Several nodes linked via a hub share a common medium. When a message is sent from a station, it is broadcast throughout the entire network and is sent to each connected node. Only the node with the correct target address processes the message. Collisions may occur and messages have to be repeatedly transmitted as a result of the large amount of data traffic. The delay time in a Shared ETHERNET cannot be easily calculated or predicted.

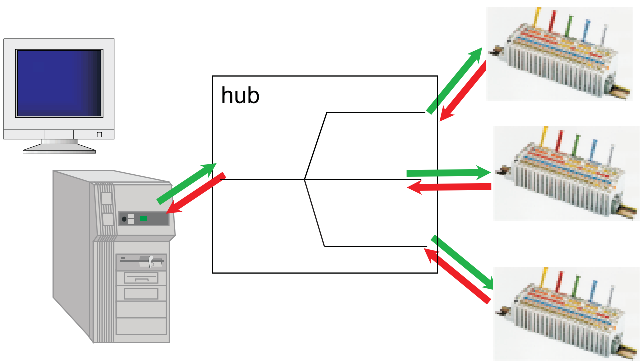


Fig. 6-6: Principle of Shared ETHERNET G012910e

Deterministic ETHERNET

The TCP/IP software or the user program in each subscriber can limit transmittable messages to make it possible to determine real-time requirements. At the same time the maximum medium message rate (datagrams per second), the maximum medium duration of a message, and the minimum time interval between the messages (waiting time of the subscriber) is limited.

Therefore, the delay time of a message is predictable.

Switched ETHERNET

In the case of Switched Ethernet, several fieldbus nodes are connected by a switch. When data from a network segment reaches the switch, it saves the data and checks for the segment and the node to which this data is to be sent. The message is then only sent to the node with the correct target address. This reduces the data traffic over the network, extends the bandwidth and prevents collisions. The runtimes can be defined and calculated, making the Switched Ethernet deterministic.

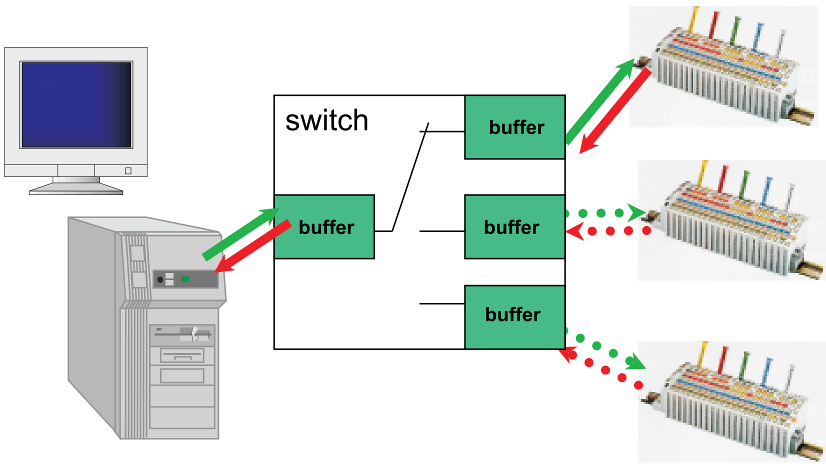


Fig. 6-7: Principle of Switched ETHERNET G012909e

6.1.3 Network Communication

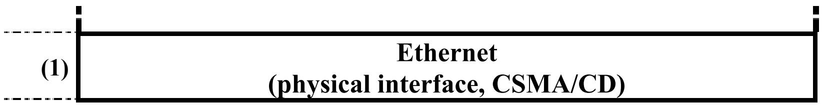
Fieldbus communication between master application and (programmable) fieldbus coupler or controller usually takes place using an implemented field-bus specific application protocol, e. g. MODBUS TCP (UDP), EtherNet/IP, BACnet, KNXNET/IP, PROFINET, Powerlink, Sercos III or others.

The protocol layer model helps with an example (MODBUS and EtherNet/IP) to explain the classification and interrelationships between the communication and application protocols.
In this example, the fieldbus communication can take place using either the MODBUS protocol or EtherNet/IP.

6.1.3.1 Protocol layer model

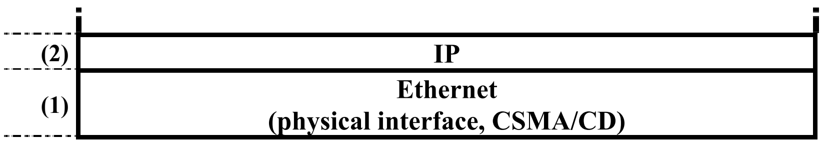
(1) Ethernet:

The Ethernet hardware forms the basis for the physical exchange of data. The exchanged data signals and the bus access procedure CSMA/CD are defined in a standard.



(2) IP:

For the communication the Internet Protocol (IP) is positioned above the Ethernet hardware. This bundles the data to be transmitted in packets along with sender and receiver address and passes these packets down to the Ethernet layer for physical transmission. At the receiver end, IP accepts the packets from the Ethernet layer and unpacks them.

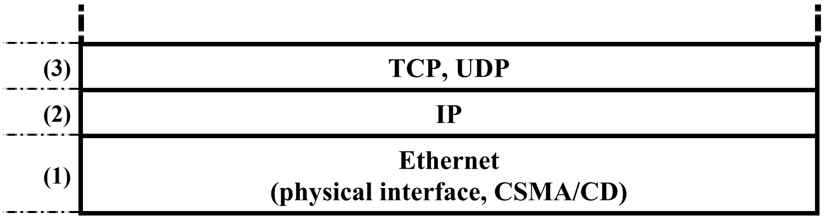


(3) TCP, UDP:

a) TCP: (Transmission Control Protocol)
The TCP protocol, which is positioned above the IP layer, monitors the transport of the data packets, sorts their sequence and sends repeat requests for missing packets. TCP is a connection-oriented transport protocol.
The TCP and IP protocol layers are also jointly described as the TCP/IP protocol stack or TCP/IP stack.

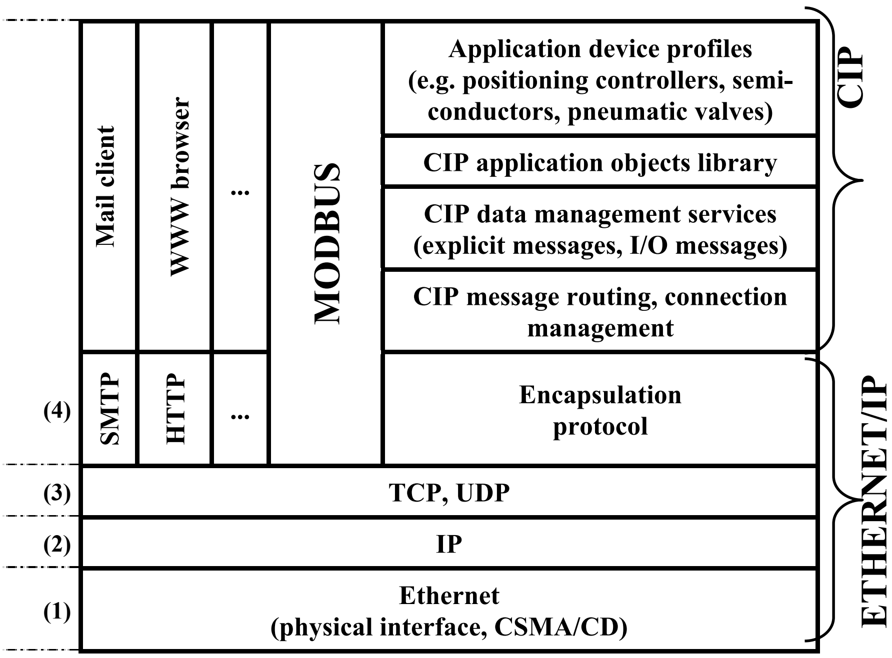
b) UDP: (User Datagram Protocol)
The UDP layer is also a transport protocol like TCP, and is arranged above the IP layer. In contrast to the TCP protocol, UDP is not connection oriented. That means there are no monitoring mechanisms for data exchange between sender and receiver.
The advantage of this protocol is in the efficiency of the transmitted

data and the resultant increase in processing speed.
Many programs use both protocols. Important status information is sent via the reliable TCP connection, while the main stream of data is sent via UDP.



(4) Management, Diagnostic and Application Protocols:

Positioned above the TCP/IP stack or UDP/IP layer are correspondingly implemented management, diagnostic and application protocols that provide services that are appropriate for the application. For the management and diagnostic, these are, for example, SMTP (Simple Mail Transport Protocol) for e-mails, HTTP (Hypertext Transport Protocol) for www browsers and some others.
In this example, the protocols MODBUS/TCP (UDP) and EtherNet/IP are implemented for use in industrial data communication.
Here the MODBUS protocol is also positioned directly above TCP (UDP)/IP; EtherNet/IP, on the other hand, basically consists of the protocol layers Ethernet, TCP and IP with an encapsulation protocol positioned above it. This serves as interface to CIP (Control and Information Protocol).
DeviceNet uses CIP in the same way as EtherNet/IP. Applications with DeviceNet device profiles can therefore be very simply transferred to EtherNet/IP.



6.1.3.2 Communication Protocols

In addition to the ETHERNET standard, the following important communication protocols are implemented in the WAGO ETHERNET based (programmable) fieldbus couplers and controllers:

- IP Version 4 (Raw-IP and IP-Multicast)
- TCP
- UDP
- ARP

The following diagram is intended to explain the data structure of these protocols and how the data packets of the communication protocols Ethernet, TCP and IP with the adapted application protocol MODBUS nested in each other for transmission. A detailed description of the tasks and addressing schemes of these protocols is contained in the following.

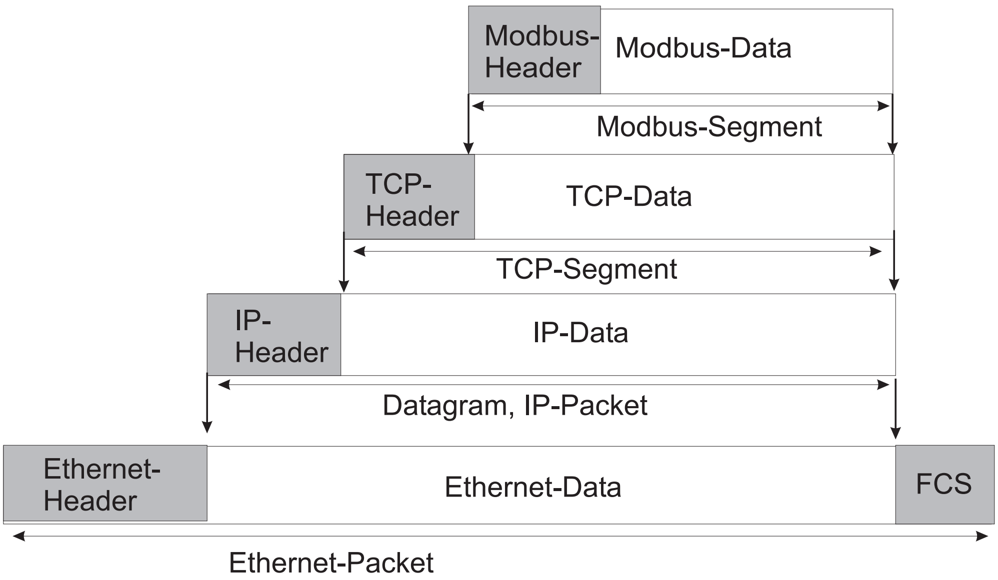


Fig. 6-8: Communication Protocols G012907e

6.1.3.2.1 ETHERNET

ETHERNET address (MAC-ID)

Each WAGO ETHERNET (programmable) fieldbus coupler or controller is provided from the factory with a unique and internationally unambiguous physical ETHERNET address, also referred to as MAC-ID (Media Access Control Identity). This can be used by the network operating system for addressing on a hardware level.
The address has a fixed length of 6 Bytes (48 Bit) and contains the address type, the manufacturer's ID, and the serial number.
Examples for the MAC-ID of a WAGO ETHERNET fieldbus coupler (hexadecimal): 00_H-30_H-DE_H-00_H-00_H-01_H.
ETHERNET does not allow addressing of different networks.
If an ETHERNET network is to be connected to other networks, higher-ranking protocols have to be used.

Note
If you wish to connect one or more data networks, routers have to be used.

ETHERNET Packet

The datagrams exchanged on the transmission medium are called "ETHERNET packets" or just "packets". Transmission is connectionless; i.e. the sender does not receive any feedback from the receiver. The data used is packed in an address information frame. The following figure shows the structure of such a packet.

Preamble	ETHERNET-Header	ETHERNET_Data	Check sum
8 Byte	14 Byte	46-1500 Byte	4 Byte

Fig. 6-9: ETHERNET-Packet

The preamble serves as a synchronization between the transmitting station and the receiving station. The ETHERNET header contains the MAC addresses of the transmitter and the receiver, and a type field.
The type field is used to identify the following protocol by way of unambiguous coding (e.g., 0800_{hex} = Internet Protocol).

6.1.3.3 Channel access method

In the ETHERNET Standard, the fieldbus node accesses the bus using CSMA/CD (Carrier Sense Multiple Access/ Collision Detection).

- Carrier Sense: The transmitter senses the bus.
- Multiple Access: Several transmitters can access the bus.
- Collision Detection: A collision is detected.

Each station can send a message once it has established that the transmission medium is free. If collisions of data packets occur due to several stations transmitting simultaneously, CSMA/CD ensures that these are detected and the data transmission is repeated.

However, this does not make data transmission reliable enough for industrial requirements. To ensure that communication and data transmission via ETHERNET is reliable, various communication protocols are required.

6.1.3.3.1 IP-Protocol

The Internet protocol divides datagrams into segments and is responsible for their transmission from one network subscriber to another. The stations involved may be connected to the same network or to different physical networks which are linked together by routers.
Routers are able to select various paths (network transmission paths) through connected networks, and bypass congestion and individual network failures. However, as individual paths may be selected which are shorter than other paths, datagrams may overtake each other, causing the sequence of the data packets to be incorrect.
Therefore, it is necessary to use a higher-level protocol, for example, TCP to guarantee correct transmission.

IP addresses

To allow communication over the network each fieldbus node requires a 32 bit Internet address (IP address).

Attention
Internet addresses have to be unique throughout the entire interconnected networks.

As shown below there are various address classes with net identification (net ID) and subscriber identification (subscriber ID) of varying lengths. The net ID defines the network in which the subscriber is located. The subscriber ID identifies a particular subscriber within this network.
Networks are divided into various network classes for addressing purposes:

•

- **Class A:** (Net-ID: Byte1, Host-ID: Byte2 - Byte4)

e.g.: 101 . 16 . 232 . 22

01100101		00010000	11101000	00010110
0	Net-ID	Host-ID		

The highest bit in Class A networks is always '0'.
Meaning the highest byte can be in a range of
'0 0000000' to '0 1111111'.

Therefore, the address range of a Class A network in the first byte is always between 0 and 127.

- **Class B: (Net-ID: Byte1 - Byte2, Host-ID: Byte3 - Byte4)**

e.g.: 181 . 16 . 232 . 22

<table>
<tr><td colspan="2">10110101</td><td>00010000</td><td>11101000</td><td>00010110</td></tr>
<tr><td>10</td><td colspan="2">Net-ID</td><td colspan="2">Host-ID</td></tr>
</table>

The highest bits in Class B networks are always ’10’.
Meaning the highest byte can be in a range of
’10 000000’ to ‘10 111111’.

Therefore, the address range of Class B networks in the first byte is always between 128 and 191.

- **Class C:** (Net-ID: Byte1 - Byte3, Host-ID: Byte4)

e.g.: 201 . 16 . 232 . 22

<table>
<tr><td colspan="2">11000101</td><td>00010000</td><td>11101000</td><td>00010110</td></tr>
<tr><td>110</td><td colspan="3">Net-ID</td><td>Host-ID</td></tr>
</table>

The highest bits in Class C networks are always ‘110’.
Meaning the highest byte can be in a range of
’110 00000’ to ‘110 11111’.

Therefore, the address range of Class C networks in the first byte is always between 192 and 223.

Additional network classes (D, E) are only used for special tasks.

Key data

	Address range of the subnetwork	Possible number of	
		networks	Subscribers per network
Class A	1.XXX.XXX.XXX - 126.XXX.XXX.XXX	127 (2^7)	Ca. 16 Million (2^{24})
Class B	128.000.XXX.XXX - 191.255.XXX.XXX	Ca. 16 thousand (2^{14})	Ca 65 thousand (2^{16})
Class C	192.000.000.XXX - 223.255.255.XXX	Ca. 2 million (2^{21})	254 (2^8)

Each WAGO ETHERNET (programmable) fieldbus coupler or controller can be easily assigned an IP address via the implemented BootP protocol. For small internal networks we recommend selecting a network address from Class C.

Attention
Never set all bits to equal 0 or 1 in one byte (byte = 0 or 255). These are reserved for special functions and may not be allocated. Therefore, the address 10.0.10.10 may not be used due to the 0 in the second byte.

If a network is to be directly connected to the Internet, only registered, internationally unique IP addresses allocated by a central registration service may be used. These are available from Inter*NIC* (International Network Information Center).

Attention
Direct connection to the Internet should only be performed by an authorized network administrator and is therefore not described in this manual.

Subnets

To allow routing within large networks a convention was introduced in the specification *RFC 950*. Part of the Internet address, the subscriber ID is divided up again into a subnetwork number and the station number of the node. With the aid of the network number it is possible to branch into internal subnetworks within the partial network, but the entire network is physically connected together. The size and position of the subnetwork ID are not defined; however, the size is dependent upon the number of subnets to be addressed and the number of subscribers per subnet.

1		8	16	24 32
1	0	Net-ID	Subnet-ID	Host-ID

Fig. 6-10: Class B address with Field for Subnet ID

Subnet mask

A subnet mask was introduced to encode the subnets in the Internet. This involves a bit mask, which is used to mask out or select specific bits of the IP address. The mask defines the subscriber ID bits used for subnet coding, which denote the ID of the subscriber. The entire IP address range theoretically lies between 0.0.0.0 and 255.255.255.255. Each 0 and 255 from the IP address range are reserved for the subnet mask.

The standard masks depending upon the respective network class are as follows:

- **Class A Subnet mask:**

255	.0	.0	.0

- **Class B Subnet mask:**

255	.255	.0	.0

- **Class C Subnet mask:**

255	.255	.255	.0

Depending on the subnet division the subnet masks may, however, contain other values beyond 0 and 255, such as 255.255.255.128 or 255.255.255.248. Your network administrator allocates the subnet mask number to you.

Together with the IP address, this number determines which network your PC and your node belongs to.

The recipient node, which is located on a subnet initially, calculates the correct network number from its own IP address and the subnet mask.
Only then does it check the node number and delivers the entire packet frame, if it corresponds.

Example of an IP address from a class B network:

IP address:	172.16.233.200	10101100 00010000 11101001 11001000
Subnet mask:	255.255.255.128	11111111 11111111 11111111 10000000
Net-ID:	172.16.00	10101100 00010000 00000000 00000000
Subnet-ID:	0.0.233.128	00000000 00000000 11101001 10000000
Host-ID:	0.0.0.72	00000000 00000000 00000000 01001000

Attention
Specify the network mask defined by the administrator in the same way as the IP address when installing the network protocol.

Gateway

The subnets of the Internet are normally connected via gateways. The function of these gateways is to forward packets to other networks or subnets.

This means that in addition to the IP address and network mask for each network card, it is necessary to specify the correct IP address of the standard gateway for a PC or fieldbus node connected to the Internet. You should also be able to obtain this IP address from your network administrator.
The IP function is limited to the local subnet if this address is not specified.

IP Packet

In addition to the data units to be transported, the IP data packets contain a range of address information and additional information in the packet header.

IP-Header	IP-Data

Fig. 6-11: IP Packet

The most important information in the IP header is the IP address of the transmitter and the receiver and the transport protocol used.

6.1.3.3.1.1 RAW IP

Raw IP manages without protocols such as PPP (point-to-point protocol). With RAW IP, the TCP/IP packets are directly exchanged without handshaking, thus enabling the connection to be established more quickly.

However, the connection must beforehand have been configured with a fixed IP address. The advantages of RAW IP are high data transfer rate and good stability.

6.1.3.3.1.2 IP Multicast

Multicast refers to a method of transmission from a point to a group, which is a point-to-multipoint transfer or multipoint connection. The advantage of multicast is that messages are simultaneously transferred to several users or closed user groups via one address.
IP multicasting at Internet level is realised with the help of the *Internet Group Message Protocol* IGMP; neighbouring routers use this protocol to inform each other on membership to the group.
For distribution of multicast packets in the sub-network, IP assumes that the datalink layer supports multicasting. In the case of Ethernet, you can provide a packet with a multicast address in order to send the packet to several recipients with a single send operation. Here, the common medium enables packets to be sent *simultaneously* to several recipients. The stations do not have to inform each other on who belongs to a specific multicast address – every station physically receives every packet. The resolution of IP address to Ethernet address is solved by the use of algorithms, IP multicast addresses are embedded in Ethernet multicast addresses.

6.1.3.3.2 TCP Protocol

As the layer above the Internet protocol, TCP (Transmission Control Protocol) guarantees the secure transport of data through the network.

TCP enables two subscribers to establish a connection for the duration of the data transmission. Communication takes place in full-duplex mode (i.e., transmission between two subscribers in both directions simultaneously).
TCP provides the transmitted message with a 16-bit checksum and each data packet with a sequence number.
The receiver checks that the packet has been correctly received on the basis of the checksum and then sets off the sequence number. The result is known as the acknowledgement number and is returned with the next self-sent packet as an acknowledgement.
This ensures that the lost TCP packets are detected and resent, if necessary, in the correct sequence.

TCP port numbers

TCP can, in addition to the IP address (network and subscriber address), respond to a specific application (service) on the addressed subscriber. For this the applications located on a subscriber, such as a web server, FTP server and others are addressed via different port numbers. Well-known applications are assigned fixed ports to which each application can refer when a connection is built up.

Examples:

Telnet	Port number: 23
HTTP	Port number: 80

A complete list of "standardized services" is contained in the *RFC 1700 (1994)* specifications.

TCP segment

The packet header of a TCP data packet is comprised of at least 20 bytes and contains, among others, the application port number of the transmitter and the receiver, the sequence number and the acknowledgement number.

The resulting TCP packet is used in the data unit area of an IP packet to create a TCP/IP packet.

6.1.3.3.3 UDP

The UDP protocol, like the TCP protocol, is responsible for the transport of data. Unlike the TCP protocol, UDP is not connection-orientated; meaning that there are no control mechanisms for the data exchange between transmitter and receiver. The advantage of this protocol is the efficiency of the transmitted data and the resulting higher processing speed.

6.1.3.3.4 ARP

ARP (Address Resolution Protocol).
This protocol combines the IP address with the physical MAC address of the respective Ethernet card. It is always used when data transfer to an IP address takes place in the same logical network in which the sender is located.

6.1.3.4 Administration and Diagnosis Protocols

In addition to the communication protocols described above, various fieldbus specific application protocols and a view protocols for system admimnistration and diagnosis can be implemented.

- BootP
- HTTP
- DHCP
- DNS
- SNTP
- FTP
- SMTP.

More information
You can find a list of the exact available implemented protocols in the chapter "Technical Data" to the fieldbus coupler and/or controller.

6.1.3.4.1 BootP (Bootstrap Protocol)

The BootP protocol defines a request/response mechanism with which the MAC-ID of a fieldbus node can be assigned a fix IP address.
For this a network node is enabled to send requests into the network and call up the required network information, such as the IP address of a BootP server.
The BootP server waits for BootP requests and generates the response from a configuration database.

The dynamic configuration of the IP address via a BootP server offers the user a flexible and simple design of his network. The WAGO BootP server allows any IP address to be easily assigned for the WAGO (programmable) fieldbus coupler or controller. You can download a free copy of the WAGO BootP server over the Internet at: http://www.wago.com.

More information
The procedure for address allocation with the WAGO BootP Server is described in detail in the Chapter “Starting up a Fieldbus Node”.

The BOOTP Client allows for dynamic configuring of the network parameters:

Parameter	Meaning
IP address of the client	Network address of the (programmable) fieldbus coupler or controller
IP address of the router	If communication is to take place outside of the local network, the IP address of the routers (gateway) is indicated in this parameter.
Subnet mask	The Subnet mask makes the (programmable) fieldbus coupler or controller able to differentiate, which parts of

	the IP address determine the network and which the network station.
IP addresses of the DNS servers	Here the IP addresses can be entered by maximally 2 DNS servers.
Host name	Name of the host

When using the bootstrap protocol for configuring the node, the network parameters (IP address, etc...) are stored in the EEPROM.

Note
The network configuration is only stored in the EEPROM when the BootP protocol is used, although not if configuration is done via DHCP.

The BootP protocol is activated in the (programmable) fieldbus coupler or controller by default.

When the BootP protocol is activated, the (programmable) fieldbus coupler or controller expects a BootP server to be permanently present.
If, however, there is no BootP server available after a power-on reset, the network remains inactive.
To operate the (programmable) fieldbus coupler or controller with the IP configuration stored in the EEPROM, you must first deactivate the BootP protocol.
This is done via the web-based management system on the appropriate HTML page saved in the (programmable) fieldbus coupler or controller, which is accessed via the “Port” link.
If the BootP protocol is deactivated, the (programmable) fieldbus coupler or controller uses the parameters stored in the EEPROM at the next boot cycle.

If there is an error in the stored parameters, a blink code is output via the IO LED and configuration via BootP is automatically switched on.

6.1.3.4.2 HTTP (HyperText Transfer Protocol)

HTTP is a protocol used by WWW (World Wide Web) servers for the forwarding of hypermedia, texts, images, audiodata, etc.

Today, HTTP forms the basis of the Internet and is also based on requests and responses in the same way as the BootP protocol.

The HTTP server implemented in the (programmable) fieldbus coupler or controller is used for viewing the HTML pages saved in the coupler/controller.
The HTML pages provide information about the coupler/controller (state, configuration), the network and the process image.
On some HTML pages, (programmable) fieldbus coupler or controller settings can also be defined and altered via the web-based management system (e.g. whether IP configuration of the coupler/controller is to be performed via the DHCP protocol, the BootP protocol or from the data stored in the EEPROM).
The HTTP server uses port **number 80**.

6.1.3.4.3 DHCP (Dynamic Host Configuration Protocol)

The coupler's/controller's built-in HTML pages provide an option for IP configuration from a DHCP server, a BootP server, or the data stored in its EEPROM by default.

Note
The network configuration via DHCP is not stored in the EEPROM, this only occurs when using the BootP protocol.

The DHCP client allows dynamic network configuration of the coupler/controller by setting the following parameters:

Parameter	Meaning
IP address of the client	Network address of the coupler/controller
IP address of the router	If communication is to take place outside of the local network, the IP address of the routers (gateway) is indicated in this parameter.
Subnet mask	The Subnet mask makes the coupler/controller able to differentiate, which parts of the IP address determine the network and which the network station.
IP addresses of the DNS servers	Here the IP addresses can be entered by maximally 2 DNS servers.
Lease time	Here the maximum duration can be defined, how long the coupler/controller keeps the assigned IP address. The maximum lease time is 24.8 days. This results from the internal resolution of timer.
Renewing time	The Renewing time indicates, starting from when the coupler/controller must worry about the renewal of the leasing time.
Rebinding time	The Rebinding time indicates, after which time the coupler/controller must have gotten its new address.

In the case of configuration of network parameters via the DHCP protocol, the coupler/controller automatically sends a request to a DHCP server after initialisation. If there is no response, the request is sent again after 4 seconds, a further one after 8 seconds and again after 16 seconds. If all requests remain unanswered, a blink code is output via the "IO" LED. Transfer of the parameters from the EEPROM is not possible.

Where a lease time is used, the values for the renewing and rebinding time must also be specified. After the renewing time expires, the coupler/controller attempts to automatically renew the lease time for its IP address . If this continually fails up to the rebinding time, the coupler/controller attempts to obtain a new IP address. The time for the renewing should be about one half of the lease time. The rebinding time should be about $^{7}/_{8}$ of the lease time.

6.1.3.4.4 DNS (Domain Name Systems)

The DNS client enables conversion of logical Internet names such as www.wago.com into the appropriate decimal IP address represented with separator stops, via a DNS server. Reverse conversion is also possible.
The addresses of the DNS server are configured via DHCP or web-based management. Up to 2 DNS servers can be specified. The host identification can be achieved with two functions, an internal host table is not supported.

6.1.3.4.5 SNTP-Client (Simple Network Time Protocol)

The SNTP client is used for synchronization of the time of day between a time server (NTP and SNTP server Version 3 and 4 are supported) and the clock module integrated in the (programmable) fieldbus coupler or controller. The protocol is executed via a UDP port. Only unicast addressing is supported.

Configuration of the SNTP client

The configuration of the SNTP client is performed via the web-based management system under the “Clock” link. The following parameters must be set:

Parameter	Meaning
Address of the Time server	The address assignment can be made either over a IP address or a host name.
Time zone	The time zone relative to GMT (Greenwich Mean time). A range of -12 to +12 hours is acceptable.
Update Time	The update time indicates the interval in seconds, in which the synchronization with the time server is to take place.
Enable Time Client	It indicates whether the SNTP Client is to be activated or deactivated.

6.1.3.4.6 FTP-Server (File Transfer Protocol)

The file transfer protocol (FTP) enables files to be exchanged between different network stations regardless of operating system.

In the case of the ETHERNET coupler/controller, FTP is used to store and read the HTML pages created by the user, the IEC61131 program and the IEC61131 source code in the (programmable) fieldbus coupler or controller.

A total memory of 1.5 MB is available for the file system. The file system is mapped to RAM disk. To permanently store the data of the RAM disk, the information is additionally copied into the flash memory. The data is stored in the flash after the file has been closed. Due to the storage process, access times during write cycles are long.

Note
Up to 1 million write cycles are possible for writing to the flash memory for the file system.

The following table shows the supported FTP commands for accesses to the file system:

Command	Function
USER	Identification of the user
PASS	User password
ACCT	Account for access to certain files
REIN	Server reset
QUIT	Terminates the connection
PORT	Addressing of the data link
PASV	Changes server in the listen mode
TYPE	Determines the kind of the representation for the transferred file
STRU	Determines the structure for the transferred file
MODE	Determines the kind of file transmission
RETR	Reads file from server
STOR	Saves file on server
APPE	Saves file on server (Append mode)
ALLO	Reservation of the necessary storage location for the file
RNFR	Renames file from (with RNTO)
RNTO	Renames file in (with RNFR)
ABOR	Stops current function
DELE	Deletes file
CWD	Changes directory
LIST	Gives the directory list
NLST	Gives the directory list
RMD	Deletes directory
PWD	Gives the actually path
MKD	Puts on a dirctory

The TFTP (Trival File Transfer Protocol) is not supported by some of the couplers/controllers.

More information
You can find a list of the exact available implemented protocols in the chapter "Technical Data" to the fieldbus coupler and/or controller.

6.1.3.4.7 SMTP (Simple Mail Transfer Protocol)

The Simple Mail Transfer Protocol (SMTP) enables sending of ASCII text messages to mail boxes on TCP/IP hosts in a network. It is therefore used for sending and receiving e-mails.

The e-mail to be sent is created with a suitable editor and placed in a mail out-basket.

A send SMTP process polls the out-basket at regular intervals and therefore finds mail waiting to be sent. It then establishes a TCP/IP connection with the target host, to which the message is transmitted. The receive SMTP process on the target host accepts the TCP connection. The message is then transmitted and finally placed in an in-basket on the target system. SMTP expects the target system to be online, otherwise no TCP connection can be established. Since many desktop computers are switched off at the end of the day, it is impractical to send SMTP mail there. For that reason, in many networks special SMTP hosts are installed in many networks, which are permanently switched on to enable distribution of received mail to the desktop computers.

6.1.3.5 Application Protocols

If fieldbus specific application protocols are implemented, then the appropriate fieldbus specific communication is possible with the respective coupler/controller. Thus the user is able to have a simple access from the respective fieldbus on the fieldbus node. There are based on ETHERNET couplers/controllers available developed by WAGO, with the following possible application protocols:

- MODBUS TCP (UDP)
- EtherNet/IP
- BACnet
- KNXnet/IP
- PROFINET
- Powerlink
- Sercos III

More information
You can find a list of the exact available implemented protocols in the chapter "Technical Data" to the fieldbus coupler and/or controller.
If fieldbus specific application protocols are implemented, then these protocols are individual described in the following chapters.

7 Glossary

B	

Baseband

Systems which operate without carrier frequencies, i.e. with unmodulated signals. Therefore, they only offer one channel which has to be logically tailored to the various requirements. Opposite: Wideband.

Bit

Smallest information unit. Its value can either be 1 or 0.

Bit rate

Number of bits transmitted within a time unit.

BNC

Bayonet Navy Connector. Socket for coaxial cable.

BootP

the bootstrap protocol is a protocol which specifies how system and network information is to be transmitted from a *server* to work stations.

Bridge

Connects two separate networks.

Broadcast

A message that is sent to all station connected to the network.

Bus

A structure used to transmit data. There are two types, serial and parallel. A serial bus transmits data bit by bit, whereas a parallel bus transmits many bits at one time.

Byte

Binary Yoked Transfer Element. A byte generally contains 8 bits.

C

Client

A system that requests the services of another. With the aid of the service request, the client can access objects (data) on the *server*. The service is provided by the server.

Coaxial cable

This cable contains a single wire and a radial shield to transmit information.

CSMA/CD

Carrier Sense Multiple Access with Collision Detection. When a collision is detected, all subscribers back off. After waiting a random delay time, the subscribers attempt to re-transmit the data.

D

Data bus

see *Bus*.

Deterministic ETHERNET

The ETHERNET data is transferred at a defined time constant. The ETHERNET network can be defined and calculated. A *Switched ETHERNET* architecture makes this possible.

Driver

Software code which communicates with a hardware device. This communication is normally performed by internal device registers.

E

ETHERNET

Specifies a Local Area Network (LAN), which was developed by Xerox, Intel and DEC in the 70's. The bus access process takes place according to the *CSMA/CD* method.

ETHERNET Standard

In 1983 ETHERNET was standardized by *IEEE 802.3* 10Base-5. ISO took over the standardization in the ISO Standard 8802/3. The essential differences between ETHERNET and the IEEE standard are to be found in the frame architecture and treatment of pad characters.

F

Fieldbus

System for serial information transmission between devices of automation technology in the process-related field area.

Firewall

Collective term for solutions which protect *LANs* connection to the *Internet* from unauthorized access. They are also able to control and regulate the traffic from the LAN into the Internet. The crucial part of firewalls are static *routers* which have an access control list used to decide which data packets can pass from which *subscriber*.

Frame

Unit of data transferred at the Data-Link layer. It contains the header and addressing information.

FTP

(File Transfer Protocol) A standard application for *TCP/IP* which allows users on one machine to transfer files to/from another.

Function

Module that always returns the same result (as a function value), prerequisite being identical input values; it has no local variables that store values beyond an invoke.

Function block

Module that delivers one or more values when being executed. They can be stored as local variables („Memory“).

G	

Gateway

Device for connecting two different networks. It converts the different protocols.

H	

Hardware

Electronic, electrical and mechanic components of a module/subassembly.

Header

A portion of the data packet, containing, among others, the address information of the receiver.

Host computer / Subscriber

Originally used to describe a central mainframe computer accessed from other systems. The services provided by the subscriber can be called up by means of local and remote request. Today, this term is also used to refer to simple computers which provide particular central *Services* (i.e. UNIX-Subscribers on the *Internet*).

HTML

Abbreviation of hypertext markup language
HTML is the description language for documents on the *World Wide Web*. It contains language elements for the design of hypertext documents.

HTTP

(Hyper Text Transfer Protocol) *client server TCP/IP* protocol which is used on the *Internet* or *Intranets* for exchanging HTML documents. It normally uses *port* 80.

Hub

A device which allows communication between several network users via *twisted pair* cable.
Similar to a *repeater*, but with many outputs, a hub is used to form a star topology.

Hypertext

Document format used by *HTTP*. Hypertext documents are text files which allow links to other text documents via particularly highlighted keywords.

I	

IAONA Europe

IAONA Europe (Industrial Automation Open Networking Alliance) is an organization for industrial network technology with the objective to establish ETHERNET in automation technology.
Further information on this subject is available on the Internet under: www.iaona-eu.com.

ICMP-Protocol

TA protocol for the transmission of status information and error messages of the *IP*, *TCP* and *UDP* protocols between IP network nodes. ICMP offers, among others, the possibility of an echo (ping) request to determine whether a destination is available and is responding.

IEC 61131-3

International standard published in 1993 for morn systems with PLC functionality. Based on a structured software model, it defines a number of high performance programming languages that can be used for various automation tasks.

IEEE

Institute of Electrical and Electronic Engineers.

IEEE 802.3

IEEE 802.3 is a IEEE standard. ETHERNET only supports the yellow cable as a medium. IEEE 802.3 also supports *S-UTP* and wideband coaxial cable. The segment lengths range from 500 m for yellow cable, 100 m for TP and 1800 m for wideband coaxial cable. A star or a bus topology is possible. ETHERNET (IEEE 802.3) uses *CSMA/CD* as a channel access method.

Intel format

Set configuration of the fieldbus coupler / controller to establish the process image. In the coupler/controller memory, the module data is aligned in different ways, depending on the set configuration (Intel/Motorola-Format, *word-alignment*,...). The format determines whether or not high and low bytes are changed over. They are not changed over with the Intel format.

Internet

A collection of networks interconnected to each other throughout the world. Its most well known area is the *World Wide Web*.

Intranet

A network concept with private network connections over which data can be exchanged within a company.

IP

Internet Protocol. The connectionless network layer, which relies on upper protocols to provide reliability.

ISA

Industry Standard Architecture. Offers a standard interface for the data exchange between CPU and periphery.

ISO/OSI- Reference Model

Reference model of the ISO/OSI for networks with the objective of creating open communication. It defines the interface standards of the respective software and hardware requirements between computer manufacturers. The model treats communication removed from specific implementations, using seven layers.

L

LAN

Local Area Network

Library

Compilation of modules available to the programmer in the programming tool **WAGO-I/O-*PRO 32*** for the creation of a control program according to IEC 61131-3.

M

Mail Server

Internet E-mails are transported and stored temporarily by so-called Mail servers. The personal post can be downloaded by such a Mail server or be sent in reverse to the far dispatch to these. With the SMTP protocol E-mails can be dispatched.

Manchester encoding

In this encoding system, a 1 is encoded as a transition from *low* to *high* and a 0 as a transition from *high* to *low*.

MIB

Short form for "Management Information Base". MIB is a sellection of infomation on all parameters, which can be handed over to the management software with a request via SNMP. Thus can be made remote maintenance, a monitoring and a control of nets by SNMP protocol.

Modules

Functions, function blocks and programs are modules.
Each module has a declaration part and a body, the latter being written in one of the IEC programming languages IL (instruction list), ST (structured text), SFC (sequential flow), FBD (function block diagram) or LD (ladder diagram).

MS-DOS

Operating system, which allows all applications direct access to the hardware.

O

Open MODBUS/TCP Specification

Specification which establishes the specific structure of a MODBUS/TCP data packet. This is dependant upon the selected function code.

Operating system

Software which links the application programs to the hardware.

P

Ping command

When a ping command (ping <IP address>) is entered, the ping program *ICMP* generates echo *request* packets. It is used to test whether a node is available.

Port number

The port number, together with the IP address, forms an unambiguous connection point between two processes (applications).

Predictable ETHERNET

The delay time of a message on an ETHERNET network can be predicted. The measures which have been taken in predictable ETHERNET make it virtually possible to realize realtime requirements.

Proxy gateway

A proxy gateway (or proxy *server,* too) allows systems which do not have direct access to the *Internet*, indirect access to the network. These can be systems which are excluded from direct access by a *firewall* for security reasons.
A proxy can filter out individual data packets between the Internet and a local network to increase security. Proxies are also used to limit access to particular servers.
In addition, proxy gateways can also have a cache function, in which case they check whether the respective *URL* address is already available locally and return it immediately, if necessary. This saves time and costs when there are multiple accesses. If the URL is not in the cache, the proxy forwards the *request* as normal.
The user should not notice the proxy *gateway* apart from the single configuration in the *web browser*. Most web browsers can be configured so that they use different or no proxy gateways per access method (*FTP*, *HTTP*).

R

Repeater

Repeaters are physical amplifiers without their own processing function. They refresh data without detecting damaged data and forward all signals. Repeaters are used for longer transmission distances or when the maximum number of nodes of 64 devices per *twisted pair* segment is exceeded. A request from a client to server is a provision to act on a serivce or funtion call.

Request

A service request from a client which requests the provision of a service from a server.

Response

The server’s reply to a client’s request.

RFC specifications

Specifications, suggestions, ideas and guidelines regarding the *Internet* are published in the form of RFCs (Request For Comments).

RJ45 connector

Also referred to as a Western connector. This connector allows the connection of two network controllers via *twisted pair* cables.

Router

Connects neighboring *subnets*, the router operating with addresses and protocols of the third *ISO/OSI* layer. As this layer is hardware independent, the routers allow transition to another transmission medium.
To transmit a message the router evaluates the logical address (source and destination address) and finds the best path if there are several possibilities. Routers can be operated as *repeaters* or *bridges*.

Routing

Method of selecting the best path over which to send data to a distant network.

S

SCADA

Abbreviation for Supervisory Control and Data Acquisition. SCADA software is a program for the control and visualization of processes.

Segment

Typically, a network is divided up into different physical network segments by way of *routers* or *repeaters*.

Server

Device providing services within a client/server system. The service is requested by the *Client*.

Service

An operation targeted at an object (read, write).

SMTP

Short form for „Simple Mail Transfer Protocol“. Standard protocol, with which E-mails are sent away in the internet.

SNMP

Short form for „Simple Network Management Protocol“. SNMP serves remote maintenance of servers. Thus leave themselves e.g. rout directly from the office of the network carrier out to configure, without someone must drive for this to the customer.

SOAP

Short form for “Simple Object Access Protocol“. XML is a standard for Meta data, the access on the XML objects takes place via SOAP.The standard defines, how transactions via internet and XML can be done and how dynamic Web services over distributed networks can be used.

Socket

Is a software interface introduced with BSD-UNIX for inter-process communication. Sockets are also possible in the network via TCP/IP. As from Windows 3.11, they are also available in Microsoft operating systems.

STP

With the STP cable (Shielded twisted pair) it acts around a symmetrical cable with in pairs stranded and protected veins. The classical STP cable is a multi-core cable, whose stranded conductors are isolated. The conductors of the STP cable are individually protected. It has no total screen.

S-STP

Beside the STP cables there is cable, which has total shielding from foil or network shielding additionally to the single shielding of the conductors still another. These cables are called S/STP cables: Screened/Shielded twisted pair.

Structured cabling

This specifies the maximum permissible cable lengths (EIA/TIA 568, IS 11801) and gives recommendations for the different types topology for ground area, building and floor cabling.

Subnet

A portion of a network that shares the same network address as the other portions. These subnets are distinguished through the subnet mask.

Subnet mask

The subnet mask can be used to manipulate the address areas in the IP address room with reference to the number of *subnets* and *subscribers*. A standard subnet mask is, for example, 255.255.255.0.

S-UTP

Screened unshielded *twisted pair* cable which only has one external shield. However, the twisted pair cables are not shielded from each other.

Switch

Switches are comparable to *bridges*, but with several outputs. Each output uses the full ETHERNET bandwidth. A switch switches a virtual connection between an input port and an output port for data transmission. Switches learn which nodes are connected and filter the information transmitted over the network accordingly. Switches are inteligent devices that learn the node connections and can transfer data at the switch and not have to send it back to the main server.

Switched ETHERNET

The segments of this type of ETHERNET are connected by *switches*. There are many applications for switching technologies. ETHERNET switching is becoming increasingly popular in local networks as it allows the realization of a *deterministic ETHERNET*.

T

TCP

Transport Control Protocol.

TCP/IP Protocol Stack

Network protocols which allow communication between different networks and technologies.

Telnet

The Telnet protocol fulfils the function of a virtual terminal. It allows remote access from the user's computer to other computer systems on the network.

Traps

Traps are unsolicited messages, which are sent by an agent to a management system, as soon as somewhat unexpected and for the management system interesting happens. Traps is with from the hardware admitted interrupts comparably. A well-known example of a Trap message is the „Blue screen“ with Win95/98.

Twisted Pair

Twisted pair cables (abbreviated to TP).

U

UDP protocol

The user datagram protocol is a transport protocol (layer 4) of the *ISO/OSI-reference model* which supports data exchange between computers without a connection. UDP runs directly on top of the underlying *IP* protocol.

URL

Abbreviation for uniform resource locator.
Address form for *Internet* files which are mostly applied within the World Wide Web (*WWW*). The URL format makes the unambiguous designation of all documents on the Internet possible by describing the address of a document or object which can be read by a *web browser*. URL includes the transmission type (http, ftp, news etc.), the computer which contains the information and the path on the computer. URL has the following format:
Document type//Computer name/List of contents/File name.

UTP

The UTP cable is a symmetrical, not-protected cable with twisted colored wires in pairs. This type of cable, which there is in execution two-in pairs and four-in pairs, is the dominating type of cable in the floor wiring and the terminal wiring.

W

WAGO-I/O-*PRO* CAA

Uniform programming environment, programming tool from WAGO Kontakttechnik GmbH for the creation of a control program according to IEC 61131-3 for all programmable fieldbus controllers. Allows testing, debugging and the start-up of a program.

Web browser

Program for reading *hypertext*. The browser allows the various documents to be viewed in hypertext and navigation between documents.

Wide band

Transmission technology which operates with a high bandwidth, thereby permitting high transmission rates. This allows several devices to transmit simultaneously.
Opposite: Baseband.

Word-alignment

Set configuration of the fieldbus coupler/controller for the creation of a process image. Word-alignment is used to establish the process image word-by-word (2 bytes).

World Wide Web

HTTP server on the Internet.

8 Literature List

Manual TCP/IP-ETHERNET for Beginners
Wiesemann & Theis GmbH
1st edition, 11/1999
http://www.WuT.de/us_printmed.html

Local Area Networks - An introduction to the technology
John E. McNamara,
Digital Press, 1985
ISBN 0-932376-79-7 Digital Press Teil Nummer EY-00051-DP

Network Troubleshooting Guide von Digital Equipment Corporation,
August 1990,
Digital Press Teil Nummer EK-339AB-GD-002

Zu RFC:
Request for Comments
http://members.xoom.com/spielchen2k/archiv/public/exploits/rfcs/rfcs/

9 Index

R

S

T

U

W

WAGO Kontakttechnik GmbH
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Phone: 05 71/8 87 – 0
Fax: 05 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: http://www.wago.com