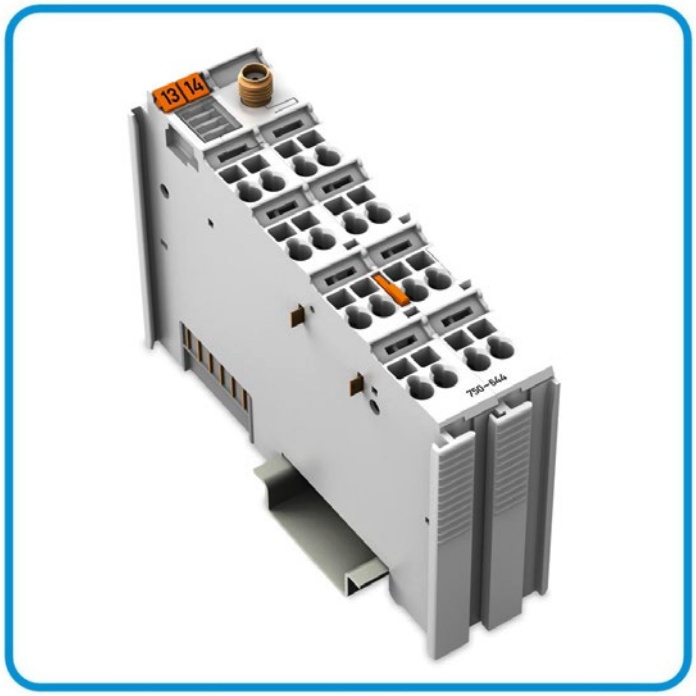
Manual



WAGO-I/O-SYSTEM 750

Bluetooth® RF-Transceiver

750-644

Version 2.0.0, valid from FW/HW Version 03/03

© 2014 by WAGO Kontakttechnik GmbH & Co. KG
All rights reserved.

WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

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

E-Mail: info@wago.com

Web: http://www.wago.com

Technical Support

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

E-Mail: support@wago.com

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

E-Mail: documentation@wago.com

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

Table of Contents

1 Notes about this Documentation

Note

Keep this documentation!
The operating instructions are part of the product and shall be kept for the entire lifetime of the device. They shall be transferred to each subsequent owner or user of the device. Care must also be taken to ensure that any supplement to these instructions are included, if applicable.

1.1 Validity of this Documentation

This documentation is only applicable to the I/O module 750-644 (*Bluetooth*® RF-Transceiver).

The I/O module 750-644 shall only be installed and operated according to the instructions in this manual and in the manual for the used fieldbus coupler/controller.

NOTICE

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

1.2 Revision History

Table 1: Revision history

Document version	Equipment version		Change
	Hardware	Software	
1.0.0	01	01	Initial draft
2.0.0	03	03	Reorganization after functional expansion of the “Ad-hoc communication ” mode

1.3 Copyright

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

1.4 Symbols

⚠ DANGER

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

⚠ DANGER

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

⚠ WARNING

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

⚠ CAUTION

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

NOTICE

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

NOTICE

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

Note

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

Information

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

1.5 Number Notation

Table 2: Number notation

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

1.6 Font Conventions

Table 3: Font conventions

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

2 Important Notes

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

2.1 Legal Bases

2.1.1 Subject to Changes

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

2.1.2 Personnel Qualifications

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

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

2.1.3 Use of the 750 Series in Compliance with Underlying Provisions

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

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

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” in the manual for the used fieldbus coupler/controller.

Appropriate housing (per 94/9/EG) is required when operating the WAGO-I/O-SYSTEM 750 in hazardous environments. Please note that a prototype test certificate must be obtained that confirms the correct installation of the system in a housing or switch cabinet.

2.1.4 Technical Condition of Specified Devices

The devices to be supplied ex works are equipped with hardware and software configurations, which meet the individual application requirements. WAGO Kontakttechnik GmbH & Co. KG will be exempted from any liability in case of changes in hardware or software as well as to non-compliant usage of devices.

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

2.2 Safety Advice (Precautions)

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

DANGER

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

DANGER

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

NOTICE

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

NOTICE

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

NOTICE

Cleaning only with permitted materials!
Clean soiled contacts using oil-free compressed air or with ethyl alcohol and leather cloths.

NOTICE

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

NOTICE

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

NOTICE

Avoid electrostatic discharge!
The devices are equipped with electronic components that you may destroy by electrostatic discharge when you touch. Pay attention while handling the devices to good grounding of the environment (persons, job and packing).

3 *Bluetooth*® Technology

Bluetooth® technology is wireless communication method based on IEEE 802.15.1. It allows license-free wireless communication in the ISM-band frequency range between 2.402 and 2.480 GHz worldwide.

This technology is primarily used for data exchange between wireless input devices or smartphones, but has also established itself in industrial applications.

Compared to other wireless technologies, only relatively low data rates are achieved, but connections are very robust. Industrial implementations also allow connections with ranges well beyond the ranges of devices for private use.

3.1 Piconet

Devices that use *Bluetooth*® technology to communicate form a so-called piconet. One device assumes the master role and all others are slaves.

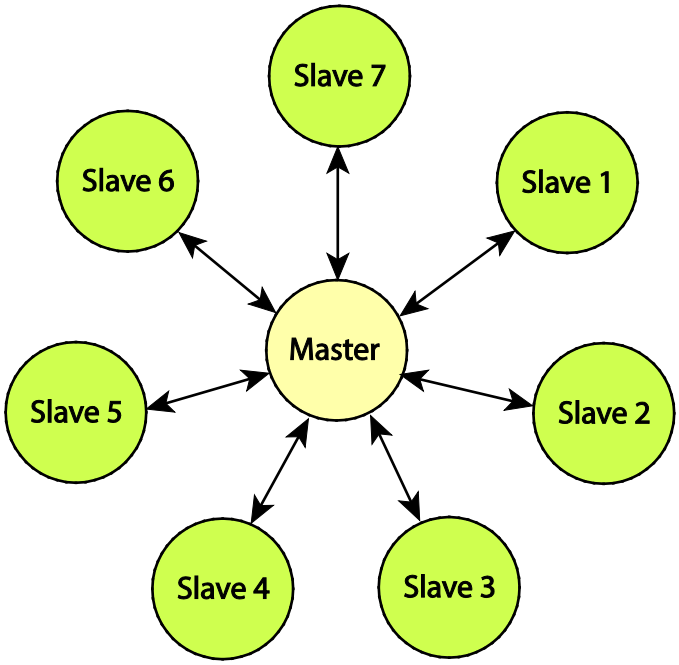


Figure 1: Piconet

Data is exchanged between the master and slave directly, but not between the slaves themselves.

3.2 Profile and Identification

Bluetooth® technology can be used in many ways. Supported applications can have very different requirements. For example, transmission of a business card or audio signal requires different protocols. *Bluetooth*® devices normally support only certain protocols. To make identification of suitable partners easier for devices, profiles have been defined. Communication between devices is then possible when both devices support the same profile.

A standardized profile is the “Serial Port Profile” (SPP). It is used for wireless transmission of serial interface data such as RS-232 or RS-485.

The “Service Discovery Profile” (SDP) supported by all *Bluetooth*® devices allows a device to discover which devices are in range and to identify supported profiles. Devices can broadcast which functions are supported by encoding such functions as a so-called “Class of Device” (CoD) and by sharing that with other devices. The CoD then makes it possible to look for devices with specific functionality.

Table 4: Classes (CoD) of WAGO devices

WAGO Device	CoD (hexadezimal)
Bluetooth® RF Transceiver 750-644 (real-time profile)	0x0020f8
Bluetooth® Module 757-801 with RS-232	0x1f00
Bluetooth® Adapter 750-921	0x1f00

In addition to the CoD, all devices have a MAC ID that is used to uniquely identify the device. Many devices also allow you to choose and save a specific name for the device. This name is then also be queried wirelessly.

In addition to standardized profiles, industrial devices sometimes also use proprietary profiles to meet specific requirements, e.g. in terms of reliability.

3.3 Security

Compatibility, connectivity and the security of radio communication are high priorities for *Bluetooth*® technology. Due to the wide variety of devices and applications that the technology uses, configurations can arise where devices are unable to communicate with each other because of the different security mechanisms implemented. This can affect devices, for example, that have no configuration options for reasons of cost or devices that do not allow direct user interaction based on their design.

3.4 Definition of Terms

The following table contains explanations about the terms used in this documentation.

Table 5: Terms used

Term	Explanation
Local	A component or device to which there is a wired connection. Example: A 750-644 I/O module represented in WAGO-I/O-*CHECK* on the operator PC.
Remote	A component or device to which there is no wired connection.
Communication partner	A remote device (from the perspective of the local device) or the local device (from the perspective of a remote device).
In range	Devices or components are in range when transmitter and receiver are mutually so strong or sensitive that communication is technically possible. Whether communication actually occurs depends on several other factors, e.g. if radio connection is even wanted or if security settings are compatible.
Allowed	For secure connections, communication is only possible with devices defined as “authorized devices”. *Bluetooth*® technology designates devices that view each other as “authorized devices”, as well as “paired” or “coupled”.
Visible	Describes a *Bluetooth*® device that responds to scans from any other *Bluetooth*® device. A device that is not visible only responds to scans or connection attempts when such attempts come from authorized devices.
Connectable	Describes a *Bluetooth*® device to which connections can be established. Sometimes it may not be possible for authorized devices to connect to a *Bluetooth*® device. This is the case, for example, when the device is designed or configured to allow only a certain number of simultaneous connections and that number has already been reached.
Connected	Describes a state in which devices are able to exchange data with each other. To do so, the devices must be in range. If the devices have a secure connection, the connection must be between authorized devices.
Authentication	Refers to the process of mutual verification of two *Bluetooth*® devices, where each device determines if the other is an authorized device. Part of the process can be user interaction, e.g. display of a dialog box on a smartphone or access to a saved configuration that indicates which devices are and are not allowed.

4 Device Description

The *Bluetooth*® RF Transceiver 750-644, hereafter referred to as the 750-644 I/O module, is used to integrate a *Bluetooth*® network (Piconet) in the WAGO-I/O-SYSTEM 750. 750-644 modules are installed and used jointly with other I/O modules of the WAGO-I/O-SYSTEMs 750 in different fieldbus systems.

The 750-644 I/O module permits wireless exchange of data within the *Bluetooth*® piconet. It can function as the coordinator (referred to in the following as the "master") or as the terminal (referred to in the following as the "slave") depending on the configuration. A maximum of seven slaves may communicate with one master.

In piconets consisting of 750-644 I/O modules exclusively, the I/O modules can use a specific proprietary profile for communication. If using the WAGO 758-912 antenna, ranges of up to 1000 meters can be achieved. In addition to particularly robust, deterministic radio communication, expanded diagnostic functions are also available.

The 750-644 I/O module can communication with devices from different manufacturers that support the protocols of the SPP profile, i.e. the wireless transmission of serial interface data. Apps are available that allow smartphones to exchange data with the 750-644 I/O module.

The WAGO-I/O-*CHECK* software is used to configure the 750-644 I/O module locally (network configuration, process image mapping). Eight 3-color LEDs indicate the current status of the 750-644 I/O module.

The 750-644 module can be used with the fieldbus couplers and controllers of the WAGO-I/O-SYSTEM 750 of the specified version or higher listed in the "Compatibility list" table.

Table 6: Compatibility list

Bus system	Fieldbus couplers/controllers	Item no.	Hardware status	Software status
750 Series				
BACnet/IP	Programmable fieldbus controller	750-830	03	03
	Programmable fieldbus controller	750-831	01	01
CAL	Fieldbus coupler	750-305		
CANopen	Fieldbus coupler	750-307		
	Programmable fieldbus controller	750-807		
	Fieldbus coupler	750-337	13	10
	Fieldbus coupler, D-Sub	750-338	01	10
	ECO fieldbus coupler	750-347	01	04
	ECO fieldbus coupler, D-Sub	750-348	01	04
	Programmable fieldbus controller, MCS	750-837	06	11
	Programmable fieldbus controller, D-Sub	750-838	01	11
DeviceNet	Fieldbus coupler	750-306	15	4I
	ECO fieldbus coupler	750-346	02	07
	Programmable fieldbus controller	750-806	07	07
EtherCAT	Fieldbus coupler	750-354	01	01
ETHERNET	Programmable fieldbus controller	750-880	04	02
	Programmable fieldbus controller, TeleControl	750-880/ 025-001		02
	Programmable fieldbus controller, TeleControl ECO	750-880/ 025-002		02
	Programmable fieldbus controller	750-881	02	03
	Programmable media-redundant fieldbus coupler	750-882	05	02
	Programmable fieldbus controller, Application Controller BA	750-884		02
	Programmable media-redundant fieldbus coupler	750-885	06	04
ETHERNET TCP/IP	TCP/IP fieldbus coupler	750-341	03	03
	TCP/IP fieldbus coupler	750-342	04	14
	Fieldbus coupler	750-352	02	02
	Programmable fieldbus controller	750-841	11	07
	Programmable fieldbus controller	750-842	13	12
	Programmable fieldbus controller	750-843		
	Programmable fieldbus controller, 2 ports	750-871	05	07
	Programmable fieldbus controller, RS-232	750-873		03
IEC60870-5	Programmable fieldbus controller for telecontrol applications	750-872	04	03
	Programmable fieldbus controller for telecontrol applications, TeleControl	750-872/ 020-000		
INTERBUS	Fieldbus coupler	750-304		
	Fieldbus coupler	750-324		
	Programmable fieldbus controller	750-804		
	Fieldbus coupler with fiber-optic connection	750-334		
	ECO fieldbus coupler	750-344		
	ECO fieldbus coupler	750-345		

Table 6: Compatibility list

Bus system	Fieldbus couplers/controllers	Item no.	Hardware status	Software status
KNX IP	Programmable fieldbus controller	750-849	03	03
KNX IP	KNX IP controller	750-889		
Linux®	Programmable fieldbus controller	750-860		
	Programmable fieldbus controller, RS-232	750-863		
LonWorks®	Fieldbus coupler	750-309		
	Programmable fieldbus controller	750-819	08	07
	Fieldbus coupler	750-319	07	05
	Peer-to-peer fieldbus coupler	750-319/ 004-000		
MODBUS	Fieldbus coupler, RS-485	750-312		
	Fieldbus coupler, RS-232	750-314		
	Fieldbus coupler, RS-485	750-315		
	Fieldbus coupler, RS-232	750-316		
	Programmable fieldbus controller, RS-485	750-812		
	Programmable fieldbus controller, RS-232	750-814		
	Programmable fieldbus controller, RS-485	750-815		
	Programmable fieldbus controller, RS-232	750-816		
PROFIBUS	DP/FMS fieldbus coupler	750-303	01	07
	Fieldbus coupler with fiber-optic connection	750-331		
	DP/V1 fieldbus coupler	750-333	16	07
	DPECO fieldbus coupler	750-343	03	06
	DP fieldbus coupler	750-323		
	Programmable fieldbus controller DP/V1	750-833	20	07
PROFINET IO	Fieldbus coupler	750-340		
	Fieldbus coupler, 2 ports	750-370	04	05
	Fieldbus coupler, 2 ports	750-375	01	01
sercos III	Fieldbus coupler, 2 ports	750-351	02	02
758 Series				
	WAGO-I/O-IPC	758-870		06

4.1 View

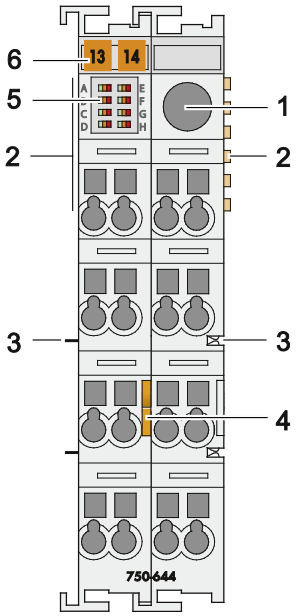


Figure 2: View

Table 7: Legend for the “View” figure

Pos.	Description	Details see section
1	Antenna socket SMA	“Device Description” > “Connectors”
2	Data contacts	“Device Description” > “Connectors”
3	Power jumper contacts	“Device Description” > “Connectors”
4	Release tab	“Mounting” > “Inserting and Removing Devices”
5	Status LEDs	“Device Description” > “Display Elements”
6	Marking possibility with Mini-WSB	---

4.2 Connectors

4.2.1 Data Contacts/Internal Bus

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

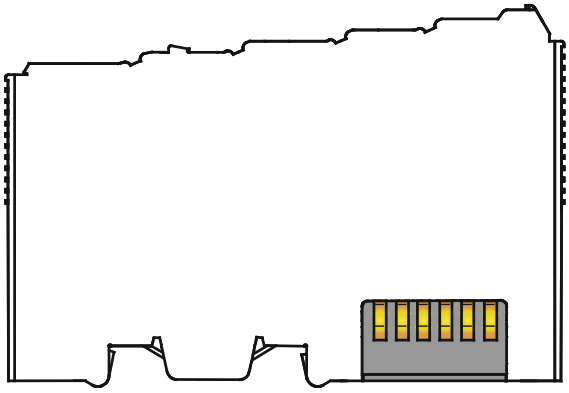


Figure 3: Data contacts

NOTICE

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

NOTICE

Ensure that the environment is well grounded!
The 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.

4.2.2 Power Jumper Contacts/Field Supply

⚠ CAUTION

Risk of injury due to sharp-edged blade contacts!
The blade contacts are sharp-edged. Handle the I/O module carefully to prevent injury.

The I/O module 750-644 has 2 self-cleaning power jumper contacts that supply and transmit power for the field side. The contacts on the left side of the I/O module are designed as blade contacts and those on the right side as spring contacts.

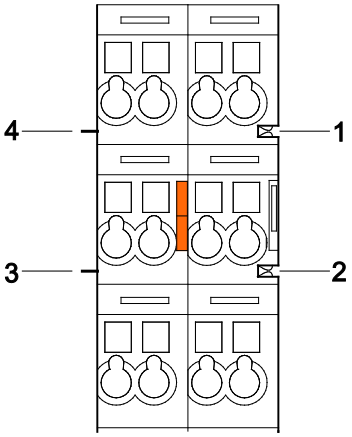


Figure 4: Power jumper contacts

Table 8: Legend for the “Power jumper contacts” figure

Contact	Type	Function
1	Spring contact	Potential transmission (U_v) for field supply
2	Spring contact	Potential transmission (0 V) for field supply
3	Blade contact	Potential feed-in (0 V) for field supply
4	Blade contact	Potential feed-in (U_v) for field supply

NOTICE

Do not exceed maximum current via power contacts!
The maximum current to flow through the power contacts is 10 A.
Greater currents can damage the power contacts.
When configuring the system, ensure that this current is not exceeded. If exceeded, an additional potential feed module must be used.

Note

Use potential feed module for Ground (earth)!
The I/O module has no power contacts for earth intake and transfer. Use a potential feed module when an earth feed is needed for the subsequent I/O modules.

4.3 Antenna

The 750-644 I/O module has an SMA socket for attaching an external antenna.

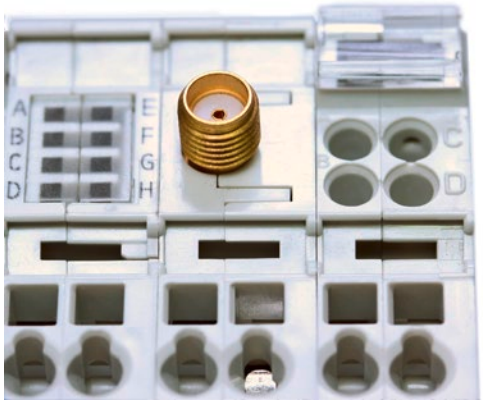


Figure 5: SMA socket

The 750-644 I/O module can also be used without an external antenna, but the achievable range of the radio connection is then significantly smaller.

In countries where use of radio products are regulated by the R&TTE directive, any antenna with radio gain of 0 dBi or less can be used. For use in all other countries where the I/O module can be operated, only the WAGO 758-912 magnetic base antenna can be used.

Figure 6: Accessory – WAGO 758-912 antenna

Note

No operating license when antenna configuration is not approved!
Use of an unapproved antenna configuration can void your operating license. This applies in particular when the equivalent isotropic radiated power (EIRP) of the structure is higher when using the antenna configuration than is permitted according to the regional or national specifications applicable to the location.

The composite function of the 750-644 I/O module and WAGO 758-912 antenna is not intended for use outside of buildings.

4.4 Display Elements

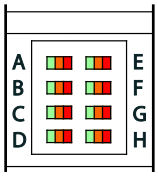


Figure 7: Display elements

8 LEDs indicate status. Depending on the operating mode (configuration/communication) and role (master/slave), the LEDs display the colors green, yellow or red as continuous illumination or flashing.

4.4.1 “Configuration” Mode

Table 9: Legend for the “Display elements” figure – “Configuration” mode

Designation	LED	State	Function
Status indicator	A	green	I/O module ready for operation
		red	I/O module note ready for operation
	E	yellow	System is configured (in “Configuration” mode only) or no connection to the first slave could be established (in “Communication” mode only).
		off	I/O module note ready for operation
-	B, C, D, F, G, H	off	-

4.4.2 “Real-Time Communication” Mode, Master

Table 10: Legend for the “Display elements” figure – “Real-Time Communication” mode, master

Designation	LED	State	Function
Status indicator	A	green	I/O module ready for operation
		red	I/O module note ready for operation
Status of the slave in slot 0x20, 0x21, 0x22, 0x23, 0x24, 0x25, 0x26	E, B, F, C, G, D, H	off	Slave not connected
		yellow	Wait for connection
		yellow flashing	Connection is being established
		green	Connection established
		green flashing	Data transfer
		red	Connection error

4.4.3 "Real-Time Communication" Mode, Slave

Table 11: Legend for the "Display elements" figure – "Real-Time Communication" mode, slave

Designation	LED	State	Function
Status indicator	A	green	I/O module ready for operation
		red	I/O module note ready for operation
Interference	D, H	green	> 53 channels free (no or negligible external activity in the frequency range)
		yellow	39 … 53 uninterrupted channels
		red	< 39 only marked as free (massive external activity in the frequency range)
		off	There is no connection or only for a few seconds
Signal quality	C, G	green	Signal quality optimal
		yellow	Signal quality moderate
		red	Signal quality low
		off	There is no connection or only for a few seconds
Signal strength	B, F	green	Signal strength optimal
		yellow	Signal strength high
		red	Signal strength low
		off	There is no connection or only for a few seconds
Status of the master in slow 0x20	E	off	Master not connected
		yellow	Wait for connection
		yellow flashing	Connection is being established
		green	Connection established
		green flashing	Data transfer
		red	Connection error

4.4.4 "Ad-hoc Communication" Mode

Table 12: Legend for the "Display elements" figure – "Ad-hoc Communication" mode

Designation	LED	State	Function
Status indicator	A	green	I/O module ready for operation
		red	I/O module note ready for operation
Status for slot 0x10, 0x11, 0x12, 0x13, 0x14, 0x15	E, B, F, C, G, D	off	No connection configured
		yellow	Connection terminated
		yellow flashing	Connection is being established
		green	Connection established
		green flashing	Data transfer
		red	Connection failed
		red flashing	Connection terminated
		flashing alternately yellow and green	Input buffer is full
-	H	off	-

4.5 Schematic Diagram

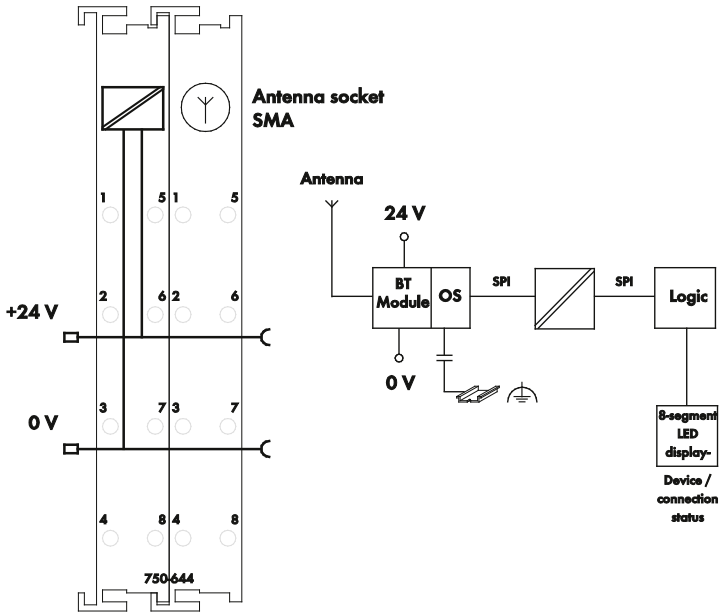


Figure 8: Schematic diagram

4.6 Technical Data

4.6.1 Device

Table 13: Technical Data – Device

Width	24 mm
Height (from upper edge of DIN 35 rail)	64 mm + excess length of the SMA socket approx. 6.5 mm
Length	100 mm
Weight	85 g

4.6.2 Supply

Table 14: Technical Data – Supply

Bluetooth® power supply	via 24 V DC field supply
Current consumption, field supply	approx. 8 mA, max. 35 mA
Power supply, system	via system voltage (DC/DC)
Current consumption, system	approx. 20 mA
Isolation	500 V antenna/system

4.6.3 Communication

Table 15: Technical Data – Communication

Data width (internal)	Configurable to 12, 24, 48 bytes, including 1 control/status byte
Wireless technology	*Bluetooth*® 2.0 + EDR
Frequency band	License-free ISM band, 2402-2480 MHz
Transmitter power	up to 20 dBm (*Bluetooth*® Class 1)
Receiver sensitivity	-94 dBm
Transmission range	Max. 1000 m open area, 100 m in building (using the magnetic base antenna from WAGO, item no. 758-912)
Topology	Piconet (1 master, max. 7 slaves)
Coexistence	AFH and adaptive transmission power
Profile, standard	Serial Port Profile (SPP)
Profile, proprietary	Real-time

4.6.4 Configuration and Diagnostics

Table 16: Technical Data – Configuration and Diagnostics

Diagnostics, statuses	Device status, connection status, signal strength, signal quality, interference
Diagnostics, interfaces	LED indication, process image, WAGO-I/O-*CHECK*
Configuration	WAGO-I/O-*CHECK*, WAGO-I/O-*PRO* CAA

4.6.5 Climatic Environmental Conditions

Table 17: Technical data – Climatic environmental conditions

Operating temperature range	0 °C … 55 °C
Storage temperature range	-25 °C … +85 °C
Relative humidity without condensation	max. 95 %
Resistance to harmful substances	Acc. to IEC 60068-2-42 and IEC 60068-2-43
Maximum pollutant concentration at relative humidity < 75 %	$SO_2 \leq 25$ ppm $H_2S \leq 10$ ppm
Special conditions	Ensure that additional measures for components are taken, which are used in an environment involving: – dust, caustic vapors or gases – ionizing radiation

4.6.6 Connection Type

Table 18: Technical Data – Field Wiring

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

Table 19: Technical Data – Power Jumper Contacts

Power jumper contacts	blade/spring contact, self-cleaning
Voltage drop at $I_{max.}$	< 1 V/64 modules

Table 20: Technical Data – Internal Bus

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

4.7 Approvals

The following approvals have been granted to 750-644 I/O modules:

cULus UL508

Conformity Marking

FCC "Federal Communications Commission" FCC ID: PV7-WTB08

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Bluetooth®

Federal Maritime and Hydrographic Agency

NCC CCAB12LP1280T4

本產品符合低功率電波輻射性電機管理辦法 第十二條、第十四條等條文規定

1. 經型式認證合格之低功率射頻電機，非經許可，公司、商號或使用者均不得擅自變更頻率、加大功率或變更原設計之特性及功能。
2. 低功率射頻電機之使用不得影響飛航安全及干擾合法通信；經發現有干擾現象時，應立即停用，並改善至無干擾時方得繼續使用。
   前項合法通信，指依電信法規定作業之無線電通信。
   低功率射頻電機須忍受合法通信或工業、科學及醫療用電波輻射性電機設備之干擾。

The following Ex approvals have been granted to 750-644 I/O modules:

TÜV 07 ATEX 554086 X

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

Ambient temperature range: 0 °C ≤ T_a ≤ +60 °C

IECEx TUN 09.0001 X

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

Ambient temperature range: 0 °C ≤ T_a ≤ +60 °C

$_{C}UL_{US}$ ANSI/ISA 12.12.01

Class I, Div2 ABCD T4

4.8 Standards and Guidelines

750-644 I/O modules meet the following requirements on emission and immunity of interference:

EMC CE-Immunity to interference	acc. to EN 61000-6-2: 2005
EMC CE-Immunity to interference	acc. to EN 61131-2: 2007
EMC CE-Emission of interference	acc. to EN 61000-6-3: 2007
EMC CE-Emission of interference	acc. to EN 61131-2: 2003

5 Function Description

The 750-644 I/O module allows wireless communication with other *Bluetooth*® devices.

Depending on the operating mode, data can be exchanged with up to 7 other 750-644 I/O modules or 6 *Bluetooth*® SPP devices.

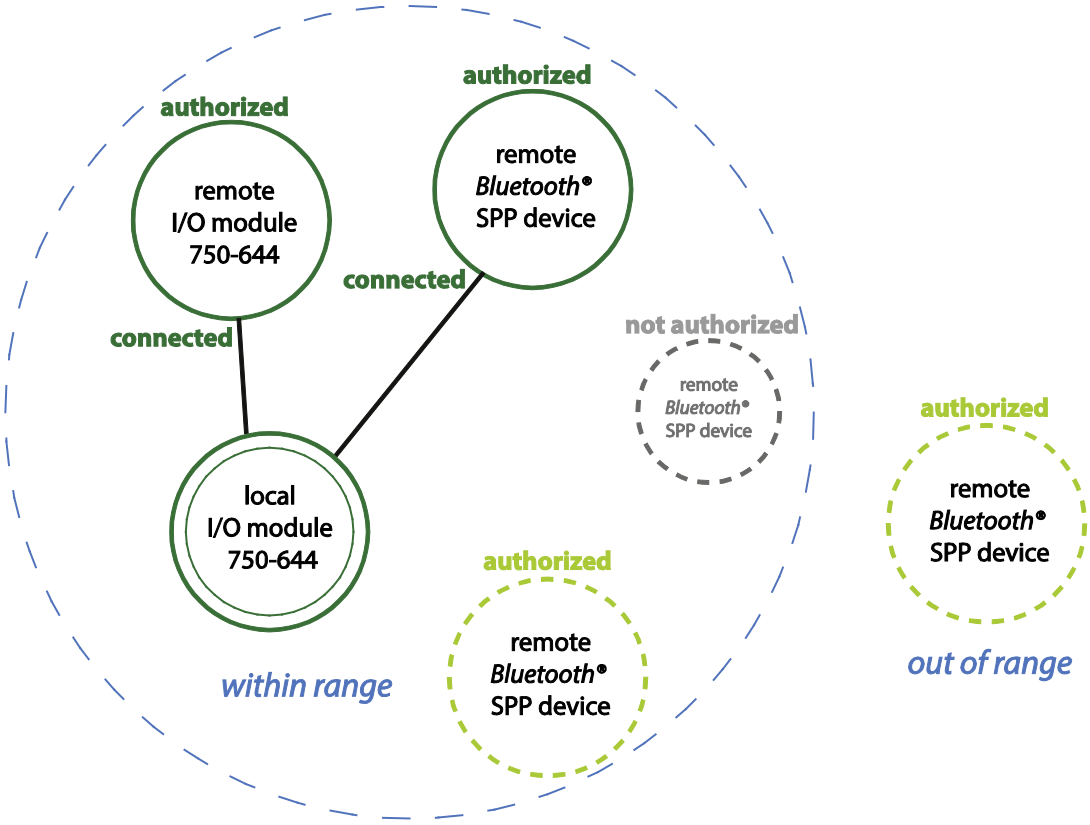


Figure 9: Device classification: Local/remote, authorized/not authorized, connected/not connected, in range/out of range

5.1 Operating Modes

The 750-644 I/O module has 3 different operating modes.

The operating mode is changed using WAGO-I/O-*CHECK* or function blocks in the WAGO-I/O-*PRO* CAA software and is controlled by mailbox commands. The *Bluetooth*® subsystem is automatically reset after changing the operating mode.

When used for the first time, the 750-644 I/O module is automatically operated in “Configuration” mode. If the “Real-Time Communication” or “Ad-hoc Communication” mode has already been selected in WAGO-I/O-*CHECK*, the 750-644 I/O module is transferred to the respective profile. During startup of the 750-644 I/O module, the last configuration made is loaded.

5.1.1 “Configuration” Mode

In “Configuration” mode, all parameters of the 750-644 I/O module are configured, e.g. via WAGO-I/O-*CHECK* or function blocks of WAGO-I/O-*PRO* CAA. Alternatively, the I/O module can also be configured via the process image directly. The Mailbox protocol is used for this purpose. A description of all Mailbox commands is available in the appendix under “Mailbox Commands”.

Of particular importance is configuration of the settings that define with which remote 750-644 I/O modules or remote *Bluetooth*® SPP devices a radio connection should be set up. These settings are summarized under the term “Net Forming” (see section “Function Description” > … > “Net Forming”).

In “Configuration” mode, no data connection is established with remote 750-644 I/O modules or remote *Bluetooth*® SPP devices. All process data assigned to the respective slots have the content “0”. However, a *Bluetooth*® inquiry can be used to scan for remote 750-644 I/O modules or remote *Bluetooth*® SPP devices and a query to identify the MAC ID and device name.

5.1.2 “Real-Time Communication” Mode

In “Real-Time Communication” mode, the 750-644 I/O module uses a proprietary real-time profile for radio communication. Operation in this mode has the following characteristics:

- Automatic connection to and communication with up to 7 other 750-644 I/O modules
- Cyclical transmission of assigned information of the output process image (PIO)
- Long ranges are possible
- Deterministic latency and cycle time
- Monitoring the timing using software watchdog

5.1.3 “Ad-hoc Communication” Mode

In “Ad-hoc Communication” mode, the 750-644 I/O module uses the standardized “Serial Port Profile” (SPP) for radio communication with remote *Bluetooth*® SPP devices. Operation in this mode has the following characteristics:

- Manual connection to and communication with up to 6 remote *Bluetooth*® SPP devices
- Serial transmission of information
- Flow control by handshake bytes (HS bytes)
- Range is determined in each case by the weakest device

5.2 Radio Interface

The 750-644 I/O module permits wireless exchange of data with remote devices using version 2.0 of the *Bluetooth*® standard.

The following profiles are supported for communication:

- **"Serial Port Profile" (SPP)** for communication with devices of other manufacturers. This profile is used in "Ad-hoc Communication" mode.
- **Real-time profile** for communication with other 750-644 I/O modules. This profile is used in "Real-Time Communication" mode.

5.2.1 Net Forming

5.2.1.1 Slots, Slot Addresses

To establish a *Bluetooth*® connection to another device, its MAC ID must be known. The MAC ID is used for unique identification of the respective connection partner.

The respective MAC ID and other connection settings can be saved in the 750-644 I/O module for up to 13 possible connection partners. The connection settings are referred to as "slots" below. The possible connection partners identified by MAC IDs are referred to as "authorized devices" below. For each slot, one so-called UserFriendlyName can also be assigned that, for example, identifies the function of the device.

Depending on the operating mode and role of the master or slave, connections can be maintained for up to seven slots. The slots for the "Real-Time Communication" mode are managed using slot addresses 0x20 … 0x26 and the slots for "Ad-hoc Communication" using slot addresses 0x10 … 0x15.

Use is described in the following table:

Table 21: Net Forming, assignment of the slot addresses

Mode	Role	Description
Real-Time Communication	Master	Connection with up to 7 other 750-644 I/O modules whose MAC IDs are saved as slots 0x20 … 0x26 and are each configured as a slave
	Slave	Connection with 1 other 750-644 I/O module whose MAC ID is saved as slot 0x20 and is configured as a master
Ad-hoc Communication	Master	Connection with up to 6 other *Bluetooth*® SPP devices whose MAC IDs are saved as slots 0x10 … 0x15 and each taking the role of a slave
	Slave	Connection with up to 6 other *Bluetooth*® SPP devices whose MAC IDs are saved as slots 0x10 … 0x15 and their role is arbitrary

5.2.1.2 Scan for Remote Devices

An inquiry can be used to determine the MAC ID of a *Bluetooth*® device if unknown. This requires that the target device is within range and that it responds to scans, i.e. it is "visible".

The result of a device scan is usually a list with the MAC IDs with the *Bluetooth*® found. In addition to the MAC IDs, the 750-644 I/O module also makes it possible to determine the names of the *Bluetooth*® devices found.

The *Bluetooth*® device names can help to identify specific devices.

Bluetooth® devices can take very different functions and a scan may find numerous devices or even devices with which communication is not possible. To prevent this, the “Class of Device” (CoD) can be used to limit the scan.

The 750-644 I/O module makes it possible to scan for *Bluetooth*® devices of a specific CoD explicitly. In this way, for example, the scan is limited to other 750-644 I/O modules.

5.2.2 Operation as a Master in “Real-Time Communication” Mode

When operated as a master in “Real-Time Communication” mode, the 750-644 I/O module plays the role of a master in the *Bluetooth*® piconet. In this operating mode, the 750-644 I/O module is not visible, i.e. it does not respond to scans. After power-on or restart in this operating mode, the 750-644 I/O module automatically attempts to establish a connection to all slots for which the following applies:

- Slot address falls in the range 0x20 … 0x26 (real-time slots)
- The MAC ID configured for the slot does not equal 0
- The slot has been enabled for connection

The 750-644 I/O module configured as a master establishes connections in order of the slot addresses. If the other 750-644 I/O modules to be connected as slaves are ready to connect, the following approximate times apply for establishing a connection:

- Approx. 5 s to establish a connection to the first slot activated
- Approx. 3 s to establish a connection to any other slot activated

If a remote 750-644 I/O module configured for a slot does not connect after approx. 5 seconds, e.g. because it is out of range, the local 750-644 I/O module continues with the next slot to be connected until the slots activated to be connected have been scanned once to establish the connection.

If the connection to one or more other 750-644 I/O modules could not be established or was interrupted later, the 750-644 I/O module attempts to reestablish a connection at regular intervals.

The interval between two attempts to establish all activated, but not yet existing connections can be configured.

Note

No data exchange when establishing a connection!
While the 750-644 I/O module establishes radio connections, any existing radio connections are paused. Cyclic data exchange is interrupted for all slots already connected.

Note

Connections to failed devices can be temporarily deactivated!
Repeated attempts to reestablish failed connections may affect data exchange for intact connections. To avoid this, individual slots can be temporarily excluded when making a connection. For this purpose, the respective connection can be marked as “not activated for connection” in “Real-Time Communication” mode. This setting is stored in volatile memory and remains valid until the next restart. Until then, the 750-644 I/O module excludes the connection concerned from automatic connection establishment.

If all slots activated for the connection are connected, or if the wait time for another connection attempt has not elapsed, the 750-644 I/O module exchanges data cyclically with all connected remote 750-644 I/O modules. The timing is monitored. If a connected slot receives no data, a warning and then error is indicated. If a connection is completely interrupted by a remote I/O module, a general error is indicated. The process data of the respective slot then remains unchanged until reconnected or new data has been received. More information is available in the sections “Process Image” and “Diagnostics”.

5.2.3 Operation as a Slave in “Real-Time Communication” Mode

When operated as a slave in “Real-Time Communication” mode, the 750-644 I/O module plays the role of a slave in the *Bluetooth*® piconet. In this operating mode, the 750-644 I/O module is not visible, i.e. it does not respond to scans. After power-on, restart in this mode or the connection is interrupted, the I/O module waits continuously until establishment or reestablishment of the connection by the assigned master. The assigned master is the device assigned to slot address 0x20.

Once the 750-644 I/O module configured as the master has established the connection, both I/O modules exchange data cyclically. Monitoring the connection status and timing is basically identical for master and slave in “Real-Time Communication” mode where only the connection to the master is monitored when operated as a slave.

5.2.4 Operation as a Master in “Ad-hoc Communication” Mode

When operated as a master in “Ad-hoc Communication” mode, the 750-644 I/O module can only be connected to *Bluetooth*® SPP devices that in turn take the role of a slave.

In this operating mode, the I/O module can maintain up to 6 connections. The following conditions apply:

- The MAC ID of the device to be connected is entered for one of the ad-hoc slots (0x10 … 0x15).
- The slot has been enabled for the connection.

The connection is not established automatically in this operating mode, i.e. if the local 750-644 I/O module should establish a connection, the process must be initiated by a corresponding parameter in the process image (see section “Process Image”). In this operating mode, connections can also be established externally. If the above conditions are met, the 750-644 I/O module automatically accepts a connection established externally.

5.2.5 Operation as a Slave in “Ad-hoc Communication” Mode

When operated as a slave in “Ad-hoc Communication” mode, the 750-644 I/O module can be connected to other *Bluetooth*® SPP devices irrespective of whether they assume the role of a master or slave.

The number of supported connections, the requirements for the configuration and the procedures for establishing a connection are identical when operated as a master.

Note

Lower power when operating in a scatternet!
If the 750-644 I/O module is operated as a slave in “Ad-hoc Communication” mode and it maintains more than one connection, the resulting *Bluetooth*® network may not have a piconet topology, but a scatternet topology. If so, significantly worse timing must be expected.

5.2.6 Visibility and Connectivity

Whether the 750-644 I/O module responds to a device scan of remote *Bluetooth*® devices or not depends on the set operating mode. If yes, it is “visible”.
The 750-644 I/O module is visible under the following conditions:

- “Ad-hoc Communication” mode or
- “Configuration” mode

A connection between the 750-644 I/O module and a remote *Bluetooth*® device is generally only possible if the configuration of the I/O module defines it as an authorized device.

- **“Real-Time Communication” mode:**
  The MAC ID of the remote device is entered for one of the slot addresses 0x20 … 0x26. The related slot is activated for the connection.

- **“Ad-hoc Communication” mode:**
  The MAC ID of the remote device is entered for one of the slot addresses 0x10 … 0x15. The related slot is activated for the connection.

Regardless of the operating mode, a connection is only possible when the security settings of the I/O module and those of the remote *Bluetooth*® device are compatible.

Note

Use authentication and encryption for compatibility!
Many *Bluetooth*® devices are set up to only allow secure connections. To achieve maximum compatibility, the I/O module should in turn be set to force authentication and encryption.

5.2.7 Encryption

The 750-644 I/O module supports both encrypted and unencrypted connections.

5.2.8 Authentication

The 750-644 I/O module supports connections with informal or no authentication or with secure authentication.

5.2.9 Coexistence

The 750-644 I/O module allows coexistence of other wireless networks through adaptive frequency hopping (AFH) according to the *Bluetooth*® standard. AFH automatically detects occupied frequency ranges of other wireless networks. The detected frequency ranges are automatically avoided by the 750-644 I/O module. The 750-644 I/O module reads the avoided frequency ranges in the form of a channel blacklist, i.e. a list that includes the 79 possible channels currently excluded from its own transmissions. Based on the channel blacklist, it is also possible to draw conclusions about other wireless networks or interference in the frequency band.

The 750-644 I/O module supports automatic adjustment of the transmission power according to the *Bluetooth*® standard. If a remote devices tells the I/O module that the received signal strength is higher than needed, the I/O module automatically reduces its transmission power. Likewise, the 750-644 I/O module automatically increases its transmission power if a remote device tells the I/O module that the received signal strength is too low.

The configuration of the 750-644 I/O module allows the user to set upper limits for automatic adjustment of the transmission power. The lower the transmission power, the smaller the distance at which other *Bluetooth*® devices can receive data. Adjusting the transmission power can be used to increase security or to improve transmission quality in other wireless networks.

Two upper limits can be defined:

- **Default transmission power:**
  This value determines which transmission power the I/O module uses for a device scan or when establishing a connection or shortly after establishing a

connection. The default transmission power must be less than or equal to the maximum transmission power.

- **Maximum transmission power:**
  This value determines the upper limit for automatic adjustment of the transmission power.

5.3 Startup Behavior

After applying the supply voltage or after a restart, the 750-644 I/O module performs an initialization that takes approx. 5 seconds. During the initialization, bit 6 is set in the status byte and there are no valid process data. LED A lights up red during the initialization.

After initialization, the 750-644 I/O module starts the last configured operating mode and LED A turns green. When used for the first time (factory setting), the 750-644 I/O module is in “Configuration” mode after the initialization.

5.4 Factory Settings

The following table lists the factory settings of the 750-644 I/O module. The factory settings can be restored and saved using the “SetFactorySettings” Mailbox command when the configuration has changed. (Exception: *Bluetooth*® device name. A device name is overwritten by the assigned Mailbox command.)

Table 22: Factory settings of the 750-644 I/O module

Parameters	Settings
Bluetooth® device name	WAGO-750-644
Default transmission power	20 dBm
Maximum transmission power	20 dBm
Device role	Slave
Mode	Configuration
Encryption	enabled
Authentication	with password
Default password	0
Quality-of-Service (QoS)	disabled
Class of Device	0x0020f8
Time to reestablish connection	30 seconds
Authorized devices	None (all lists are initialized with “0”)
Linked devices	None (all lists are initialized with “0”)
Process image sizes of the up to 7 slaves in the master	10,0,0,0,0,0,0 bytes

5.5 Timing

In “Real-Time Communication” mode, the 750-644 I/O module achieves low transmission and cycle times. The timing of data transmission is continuously monitored. If no data is received by the opposite side for a longer period of time, a warning and then error is indicated.

The timing during normal operation primarily depends on the role in the piconet and the number of slaves connected to the master. For this reason, the time limits, which when exceeded indicate a warning or error, depend on these factors.

Table 23: Timing of the 750-644 I/O module – time limits

Name	Value	
BTCOM_ WARNTIME	Master, 1 slave linked:	40 ms
	Master, 2 … 5 slaves linked:	+20 ms per slave
	Master, 6 slaves linked:	240 ms
	Master, 7 slaves linked:	280 ms
	Slave:	280 ms
BTCOM_ ERRORTIME	Master, 1 slave linked:	80 ms
	Master, 2 … 5 slaves linked:	+40 ms per slave
	Master, 6 slaves linked:	480 ms
	Master, 7 slaves linked:	560 ms
	Slave:	560 ms

For optimal timing, there must be a valid piconet configuration. If the master is unable to establish a link to all slaves, the attempts at reintegration of these slaves lead to interruptions in data traffic (see section “Function Description” > … > “Operation as a Slave in 'Real-Time Communication' Mode”).

During normal operation, the timing falls well below the time limits specified in the table “Timing of the 750-644 I/O module – time limits”.

6 Process Image

The process image of the 750-644 I/O module is used to exchange data with, configure, parameterize and diagnose devices connected wirelessly.

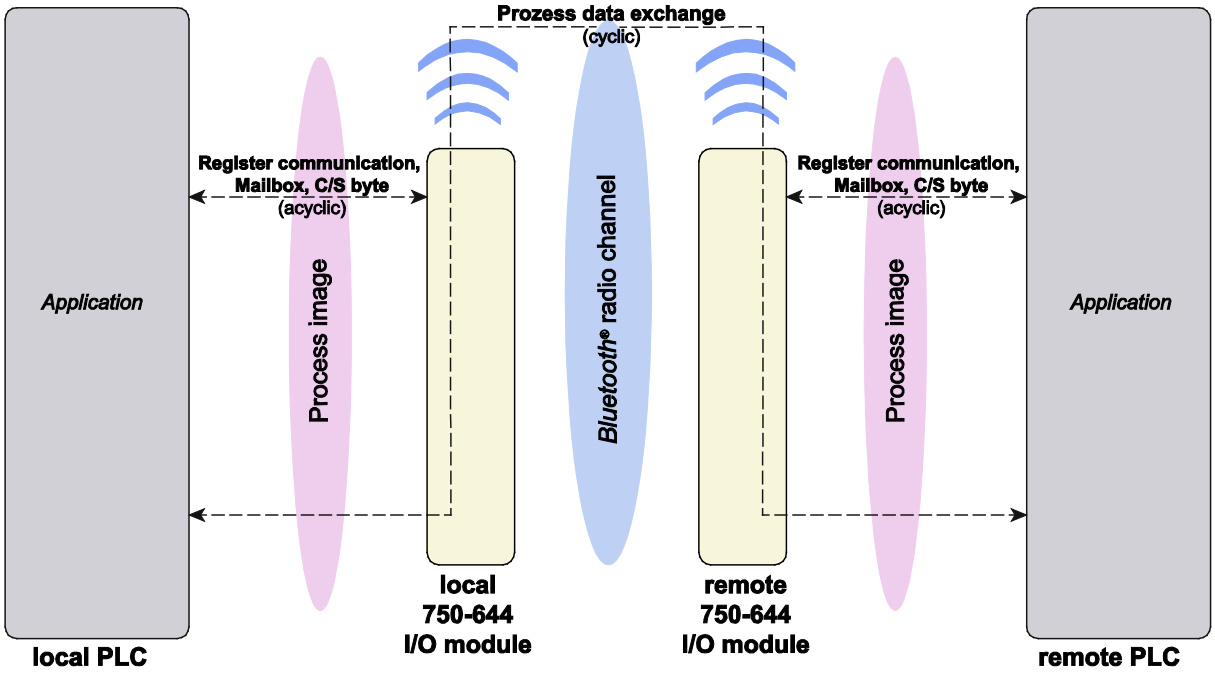


Figure 10: Cyclic and acyclic communication

The size of the process image of the 750-644 I/O module is always identical for input and output data. Possible settings are 12, 24 and 48 bytes.

Depending on the current configuration of the 750-644 I/O module and how it is used, the process image is divided into various segments: Fixed components are a control/status byte (C/S byte) and a null byte. The rest of the process image can be used for register communication, Mailbox configuration or process data exchange (see also figure “Segments of the Process Image”).

- If the process image is used for **register communication**, it is not available for data exchange with remote *Bluetooth*® SPP devices or for configuring the 750-644 I/O module using the Mailbox protocol. Regardless of the operating mode, all bytes of the process image are invalid that are not occupied by data for register communication.

- In “Configuration” mode, the **Mailbox protocol** can be used for configuration. For this purpose, the Mailbox is activated using a bit in control byte C0. If the Mailbox is activated, it occupies 6, 12 or 18 bytes in the process image. The 750-644 I/O module populates any remaining bytes with 0.

- In “Real-Time Communication” and “Ad-hoc Communication” modes, up to 7 *segments of configurable size* (hereafter also referred to as “slots”) are available for exchanging **process data** with remote *Bluetooth*® SPP devices. The Mailbox can also be activated in both operating modes and then over-

lays a certain part of the process data according to its size. The data of the slots that are completely or partially overlaid by the Mailbox are only available again for data exchange with remote *Bluetooth*® SPP devices when the Mailbox is hidden. The 750-644 I/O module populates bytes in the process image not assigned to any slot with 0.

The following figure “Segments of the process image” shows the structure of the process image at a glance:

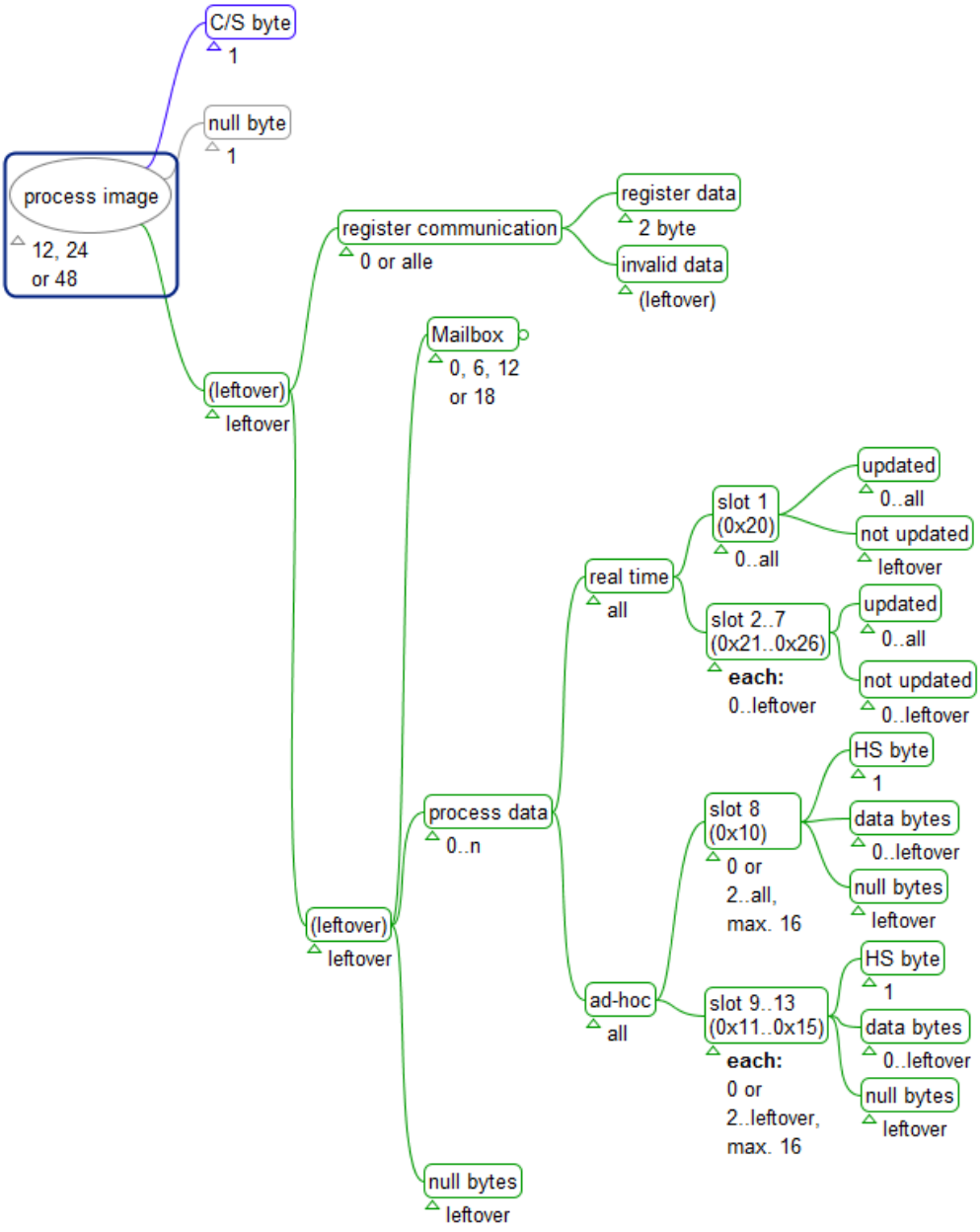


Figure 11: Segments of the process image

6.1 Control/Status Byte (C/S Byte)

Table 24: Control byte C

Control byte C							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
REG=0	0	MBX	0	0	0	0	0
REG=1	REGCOM						
REG=0	Register communication is disabled						
REG=1	Register communication is enabled						
MBX	0: Mailbox is disabled 1: Mailbox is enabled						

Table 25: Status byte S

Status byte S							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
REG=0	DIRTY	MBX	0	WARN	MBXR	WATCH	0
REG=1	REGCOM						
REG=0	Register communication is disabled						
REG=1	Register communication is enabled						
DIRTY	Process data invalid, e.g. due to interrupted radio connection or because "Configuration" mode is set						
MBX	0: Mailbox is disabled						
	1: Mailbox is enabled						
WARN	0: No warning						
	1: Warning; e.g. if the defined time limit for warning messages has elapsed, no packet has been received on the other side						
MBXR	0: The Mailbox is not enabled on any of the 750-644 I/O modules connected by radio						
	1: The Mailbox is enabled on at least one of the 750-644 I/O modules connected by radio, process data can be obsolete.						
WATCH	0: Process data is regularly received by all 750-644 I/O module connected by radio.						
	1: At least one of the 750-644 I/O modules connected by radio has not received any process data for a period longer than defined as the error limit.						

More information about the behavior of the bits in the status byte is available in section “Diagnostics”.

6.2 Prozess Data

The process data range in the process image can be divided into as many as 7 *segments of configurable size* (slots). Each slot can be assigned to a remote *Bluetooth*® SPP device. The process data in the range of the respective slot is then used for data exchange with the remote *Bluetooth*® SPP device.

Data written to the output process image (PIO) of a segment is transmitted from the 750-644 I/O module to the remote *Bluetooth*® SPP device. Inversely, the 750-644 I/O module makes data received by the remote *Bluetooth*® SPP device available in the input process image (PII).

The size of a segment is referred to as “cut-off”. This cut-off affects the band-width available for communication with the assigned *Bluetooth*® SPP device. Therefore, larger segments allow higher data rates than small segments.

Remote *Bluetooth*® SPP devices are assigned to segments in two steps:

1. Link the *Bluetooth*® SPP device to one of the 13 slots:
   - Slot addresses 0x20 … 0x26 are for “Real-Time Communication” mode

- Slot addresses 0x10 … 0x15 are for “Ad-hoc Communication” mode

2. Select operating mode:
   - in “Real-Time Communication” mode, slots 0x20 … 0x26 are mapped to segments 3 … 9
   - in “Ad-hoc Communication” mode, slots 0x10 … 0x15 are mapped to segments 3 … 8, segment 9 remains unused

The following table clarifies this assignment:

Table 26: Overview of segments and slots

Segment	Real-time slot				Ad-hoc slot			
	No.	Addr.	Offset	Cut-off	No.	Addr.	Offset	Cut-off
1	---	---	0	1	---	---	0	1
2	---	---	1	1	---	---	1	1
3	1	0x20	2	r0	8	0x10	2	a0
4	2	0x21	2 + r0	r1	9	0x11	2 + a0	a1
5	3	0x22	2 + r0 + r1	r2	10	0x12	2 + a0 + a1	a2
6	4	0x23	2 + r0 + r2 + r2	r3	11	0x13	2 + a0 + a1 + a2	a3
7	5	0x24	2 + r0 + … + r3	r4	12	0x14	2 + a0 + … + a3	a4
8	6	0x25	2 + r0 + … + r4	r5	13	0x15	2 + a0 + … + a4	a5
9	7	0x26	2 + r0 + … + r5	r6	---	---	2 + a0 + … + a5	n - (2 + a1 + … + a5)
10	---	---	2 + r0 + … + r6	n - (2 + r0 + … + r6)	---	---	---	---

No.: Consecutive number of the slot
Addr.: Address of the slot
Offset: Offset of the segment in the respective operating mode
Cut-off: Size of the segment in the respective operating mode
r0 … r6, a0 … a5: Cut-off configured for the respective slot
n: Total size of the process image (12, 24 or 48 bytes)

6.2.1 “Real-Time Communication” Mode

In “Real-Time Communication” mode, the local 750-644 I/O module transmits the process data of each segment of the output process image (PIO) by radio to the assigned remote 750-644 I/O module. The local 750-644 I/O module maps the data received from the remote I/O module to the assigned segment of its input process image (PII).

If the cut-off setting matches on both sides, the data in the respective segment of the PII is overwritten completely by the data located on the other side in the associated segment of the PIO each time data is exchanged.

If the cut-off setting does not match between two communicating 750-644 I/O modules, only the data that corresponds to the smaller cut-off can be transmitted cyclically. The 750-644 I/O module with the larger cut-off then overwrites the extra bytes of its respective segment in the PII with the value 0. Similarly, data written to extra bytes of this segment in the PIO are not transmitted to the other side, but discarded.

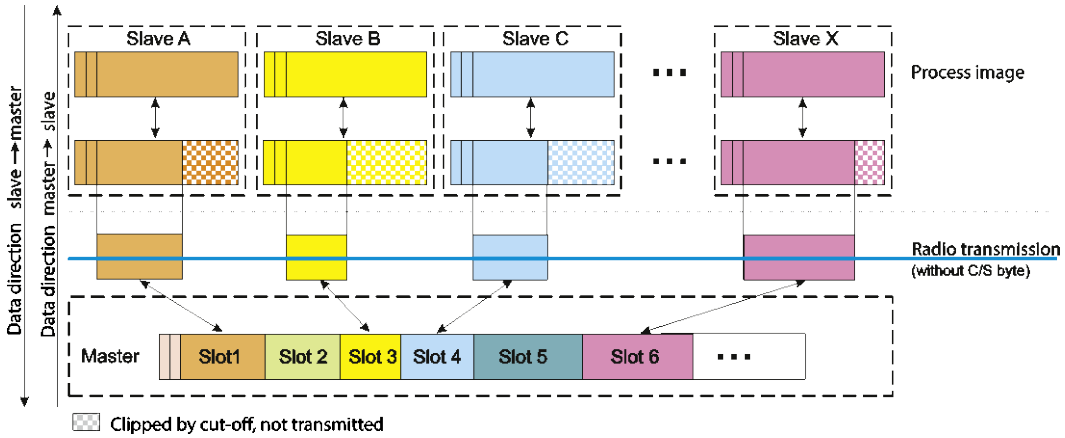


Figure 12: Process image mapping between master and slaves

A 750-644 I/O module configured as the master can exchange data in “Real-Time Communication” mode with the 750-644 I/O modules configured for slot addresses 0x20 … 0x26. A 750-644 I/O module configured as a slave can only exchange data in “Real-Time Communication” mode with the other 750-644 I/O module configured for slot 0x20.

If a 750-644 I/O module is configured as a slave, the range available for process data is reserved entirely for slot 0x20. This reservation is independent of the value set as the cut-off at this 750-644 I/O module for this slot. In addition, a 750-644 I/O module configured as a slave can only connect in “Real-Time Communication” mode to the remote 750-644 I/O module configured for slot address 0x20.

The data width available for data exchange thus depends exclusively on the cut-off set for the 750-644 I/O module configured as the master. For data to be exchanged, a value of at least 1 must be set on the master for the cut-off of the respective segment.

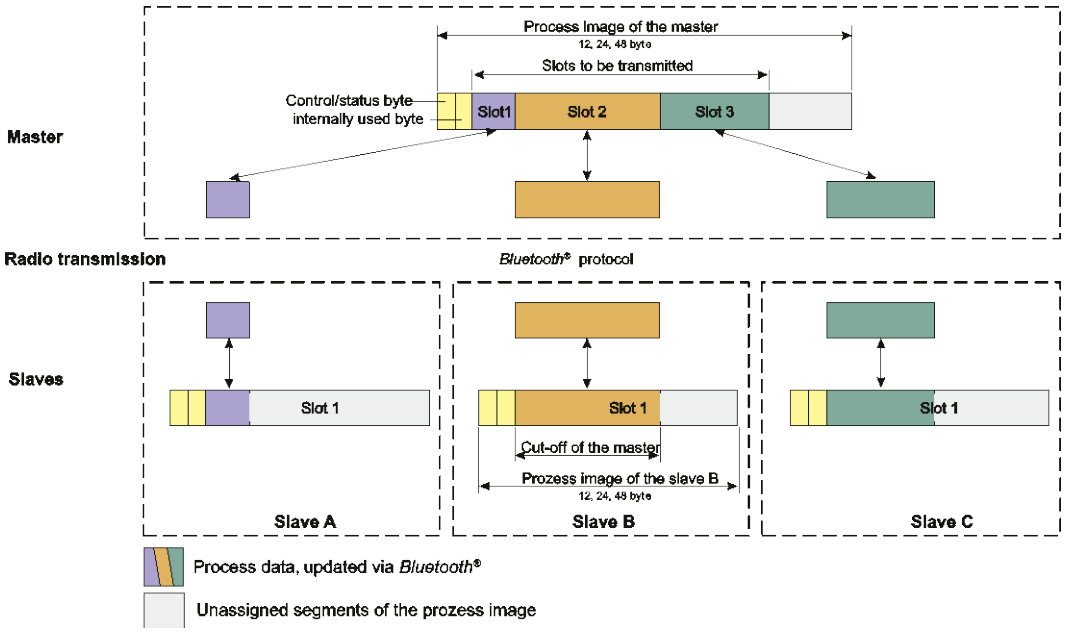


Figure 13: Process image mapping of the slave

The maximum value of the cut-off is the total size of the process image minus 2. If just two 750-644 I/O modules are exchanging data with each other, this maximum value can be used. In this case, the maximum bandwidth is solely available to the connection.

However, if a 750-644 I/O module configured as the master exchanges data with more than one other 750-644 I/O module, the available bandwidth must be divided between the segments and thus between the slots and configured *Bluetooth*® SPP devices.

Data is exchanged cyclically, i.e. with low latency and cycle time, but no flow control. The cycle time, i.e. the time interval to the next transmission of data in the segment of the PIO, depends on the current configuration. Whether the data has changed since the previous transmission cycle plays no role. In the PII, segment data always remains current until updated by a new wireless transmission from the other 750-644 I/O module.

Example configurations for segments in the process image from master and slaves:

- "**1:1**" – Master with 48-byte process image, a cut-off of 0 bytes is set for slot address 0x20, a cut-off of 46 bytes is set for slot address 0x21. Slave with 48-byte process image, the cut-off setting is not evaluated. All 46 bytes are available for data transmission between the master and the slave linked to slot address 0x21.

- "**1:2**" – Master with 48-byte process image, 23 bytes for slot addresses 0x20 and 0x22, 0 bytes for all other slot addresses. The slave linked to 0x20 has a process image of 24 bytes, the slave linked to 0x22 has a process image of 48 bytes. 22 bytes can be exchanged cyclically between master and first-mentioned slave – the process image of the slave is limiting here. 23 bytes can be exchanged cyclically between master and second-mentioned slave – the cut-off set on the master is limiting here.

- "**1:7**" – Master with 48-byte process image, cut-offs 6, 6, 6, 6, 6, 8, 8 are set for slot addresses 0x20, 0x21, 0x22, 0x23, 0x24, 0x25, 0x26. In this case, the limiting factor is always the cut-off set on the master independent of the size of the process image of the slave.

6.2.2 "Ad-hoc Communication" Mode

In "Ad-hoc Communication" mode, a 750-644 I/O module transmits the process data of each of its segments of the output process image (PIO) via the RFCOMM protocol to another assigned *Bluetooth*® SPP device. The 750-644 I/O module then maps the data received to the associated segment of its input process image (PII).

Data is exchanged serially, the first byte of each segment is used to control data flow to and from the 750-644 I/O module and other bytes of the segment contain serial data. To allow data to be exchanged with the *Bluetooth*® SPP device configured for the respective slot, the cut-off for this segment must be set to at

least 2 bytes. The maximum value for the cut-off of a segment of an ad-hoc slot is 16 bytes.

Table 27: Process data of the slot in “Ad-hoc Communication” mode – basic structure

Byte	Description
0	Control/status byte of the slot
1 … n	Data byte D0 … Dn

n: (size of the slot) - 1

Data is transmitted as needed to the respective remote *Bluetooth*® SPP device exclusively. If new data to send is provided by the PLC to the 750-644 I/O module in the respective segment of the PIO, the 750-644 I/O module acknowledges receipt and saves the data in the **transmit buffer**. If there is data in the transmit buffer, the 750-644 I/O module attempts to transmit the data to the other side until the transmit buffer is empty again. If the other side is not accepting data, the transmit buffer fills up. If the transmit buffer is full, the 750-644 I/O module no longer accepts any data from the segment in the PII.
If the 750-644 I/O module receives data from the other side, the 750-644 I/O module saves the data in an **input buffer**. If there is data in the input buffer, the 750-644 I/O module makes data in the segment of the PII available.
Data exchange between a PLC application and the 750-644 I/O module is controlled by the control/status byte (C/S byte) of the respective slot.
The structure of the control byte and status byte is shown in the two tables below:

Table 28: Process data of the slots in “Ad-hoc Communication” mode – control byte C

Control byte C							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
LEN				0	CR	RA	TR
TR	TRANSMIT_REQUEST If the value changes: Indication that the PIO contains new data for the respective device.						
RA	RECEIVE_ACKNOWLEDGE If consistent with bit RR of the status byte: Confirmation that new data of the PII has been processed.						
CR	CONNECT_REQUEST If the value changes: Indication that a radio link to the device corresponding to this slot should be established or disabled.						
LEN	LENGTH Indicates how many of data bytes D0 … Dn of the PIO contain new data.						

Table 29: Process data of the slots in “Ad-hoc Communication” mode – status byte S

Status byte S							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
LEN				BF	CS	RR	TA
TA	TRANSMIT_ACKNOWLEDGE If consistent with bit TR of the control byte: Confirmation that new data of the PIO has been processed.						
RR	RECEIVE_REQUEST If the value changes: Indication that the PII contains new data for the respective device.						
CS	CONNECTION_STATUS 0: No connection 1: Connection available						
BF	BUFFER_FULL 0: Input buffer not full 1: Input buffer full						
LEN	LENGTH Indicates how many of data bytes D0 … Dn of the PII contain new data.						

Note

Connection with other 750-644 I/O modules in "Ad-hoc Communication" mode only makes sense in mixed mode!
"Ad-hoc Communication" mode can also be used for communication between several 750-644 I/O modules. Just like all other *Bluetooth*® SPP devices, the I/O modules must be configured to slot addresses 0x10 … 0x15. Because the timing compared to "Real-Time Communication" mode is much worse, the procedure should only be selected when communication is required with other types of *Bluetooth*® SPP devices at the same time (mixed mode).

6.3 Higher-Level Configuration Protocols

NOTICE

Limited description of configuration and parameterization process!
Use register communication, parameter channel and Mailbox for the purposes described in these instructions only. Improper use of these protocols can cause the 750-644 I/O module to not work as intended.

Note

Use WAGO-I/O-*CHECK* for configuration and parameterization.
If possible, use the WAGO-I/O-*CHECK* software to configure and parameterize the 750-644 I/O module. The software displays all relevant settings for the I/O module in a graphical user interface and ensures convenient, proper use.

For configuration, parameterization or diagnostics, process data can partially and temporarily be overlaid with data from one of two configuration protocols, so-called register communication or Mailbox:

- For register communication, 2 bytes are overlaid as needed. All subsequent bytes are then invalid.
- The size of the Mailbox can be configured. Depending on the setting, 6, 12 or 18 bytes are overlaid as needed. Subsequent bytes up to the next segment boundary are invalid.

The possible settings related to the overall process image and Mailbox size, as well as the impact of mapping the overlaid communication protocol are illustrated in the following table:

Table 30: Process data and register communication

Process data communication		Register communication
Mailbox activated	**Mailbox deactivated**	
Control/status (1 byte length, from byte 0)		
Used internally (1 byte length, from byte 1)		
Mailbox (Acyclic data, 6 byte … 18 byte length, from byte 2 to byte n)	Process data (Cyclic data, 0 byte … 46 byte length, from byte 2 to m)	Register data (2 byte length, from byte 2 to byte 3)
Process data (Cyclic or serial data, 0 byte … 40 byte length, from byte n+1 to byte m) (Note validity of the data!)		Invalid data (from byte 4 to byte m)

The figure below graphically shows data overlaid by data from register communication or Mailbox:

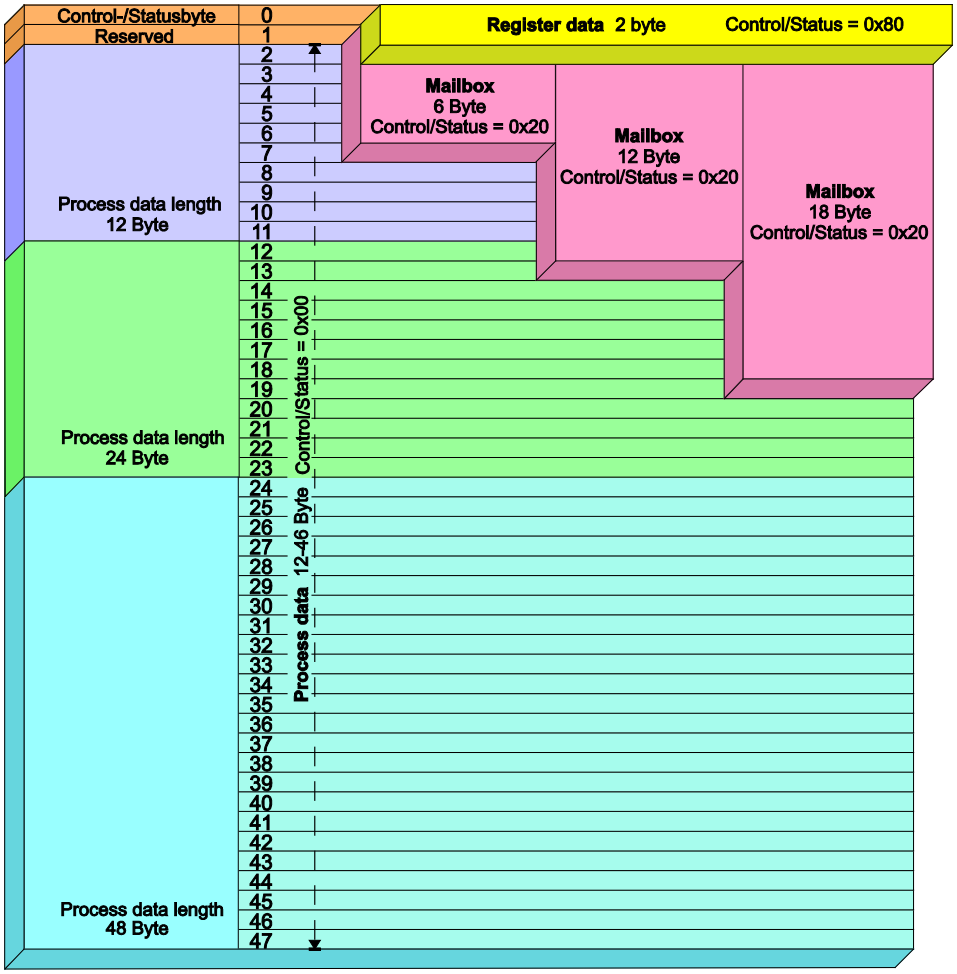


Figure 14: Process data overlaid with Mailbox or register data

Whether process data is overlaid by data from register or Mailbox data or process data is made invalid is controlled by specific bits in the control byte (see section “Control/Status Byte (C/S Byte)”).

6.3.1 Mailbox

The 750-644 I/O module provides a protocol referred to as “Mailbox” below for parameterization and diagnostics. The protocol allows data exchange between the 750-644 I/O module and a PLC application.

The Mailbox of the 750-644 I/O module is not a permanent part of the process image, but may be mapped as required. If the Mailbox is mapped, part of the process image is no longer available for exchanging process data with remote devices.

If bit 5 is set in the control byte in the output process image (PIO), it is viewed as a request to the 750-644 I/O module to map the Mailbox. The 750-644 I/O module responds to the request by setting bit 5 in the status byte. The Mailbox is thus considered mapped. If the Mailbox is mapped, it can be used to parameterize the I/O module or to query status or diagnostic information.

The Mailbox begins after the control/status byte and the internal byte with the 3rd byte of the process image and, depending on the size set, covers **6**, **12** or **18 bytes** of the process image (see section “Process Image” > ... > “Process Data Overlaid by Mailbox”).

Note

The Mailbox size determines which mailbox commands can be executed!

To configure with WAGO-I/O-*CHECK* or function blocks of the WAGO-I/O-*PRO* CAA, you can choose each available Mailbox size independent of fieldbus limitations.

- For successful configuration, a Mailbox size of at least 12 bytes is necessary.
- If all diagnostic commands are to be fully available, a Mailbox size of 18 bytes is required.

If you are using a fieldbus over which less than 20 bytes per data element can be transmitted (e.g. CANopen), you should reduce the Mailbox size again to an appropriate size after successfully completing the device configuration.

In the area of the process image overlaid by the Mailbox, the data is interpreted as Mailbox data, so commands (Opcodes) can be sent here. The structure of the data in the Mailbox is always identical:

Table 31: Structure of the process image when the Mailbox is visible

Byte(s)	Assignment	Description
0	Control/status byte	See section “Process Image” > ... > “Control/Status Byte”
1	-	-
2	Mailbox: Opcode	See description below.
3	Mailbox: Toggle byte	
4 … 7, 4 … 13 or 4 … 19	Mailbox: Parameters and/or null bytes	
Rest	Process data	See section “Process Image” > ... > “Process Data”

The so-called opcode defines the operating to be processed by the I/O module. The opcode determines how many bytes are occupied by parameter data in the request and response or how many bytes are populated with the value 0 in the back part of the area overlaid by the Mailbox (null bytes).

The toggle byte consists of two parts:

- **Bit 7** serves as a toggle bit
  - The toggle bit in the output process image (PIO) is used to make repeated requests with the same opcode distinguishable.
  - The toggle bit in the input process image (PII) is used to confirm execution of requests in general.
- **Bits 6 … 0** contain the so-called return code.
  The return code in the process image of the inputs provides more detailed information about whether execution of the request was successful or whether errors have occurred.

All processes within the Mailbox follow the request/response principle:

1. A request is made in the PIO. This can be generated by a PLC application, for example.

2. The I/O module generates a response in the PII.

There is a request when the value of the opcode or toggle byte changes in the PIO compared to the previous status. The 750-644 I/O module then evaluates the entire request including any parameters and starts to process the requested operation.

Table 32: Mailbox request

Mailbox request								
Byte	**2^7**	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**
0	0	0	1	0	0	0	0	0
1	-							
2	Opcode*							
3	T**	-						
4	Request Parameter byte 0							
…	…							
max. 19	Request Parameter byte 15							

*Command code of the Mailbox job
**Toggle bit: The Mailbox job is started when there is a change.

Note

Interpretation only when changing the opcode or toggle bit!
The content of the Mailbox is only interpreted by the 750-644 I/O module when the opcode changes or the toggle bit is inverted. A change in the parameters does not result in processing of the mailbox content!

Note

Ensure consistency of the parameters!
If a request includes parameters, they must be written to the output process image with the associated opcode and toggle bit at the same time. If the parameters only occupy part of the area of the process image overlaid by the Mailbox, all unoccupied bytes must be overwritten with the value 0.

There is a response when the value of the opcode and bit 7 of the toggle byte in the PII matches those in the PII. A PLC application, for example, can evaluate the entire response. The return code plays a specific role here, for example, because the parameter data of the response can be invalid when certain errors arise.

Table 33: Mailbox response

Mailbox response								
Byte	**2^7**	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**
0	0	0/1	1	0	0	0	0	0
1	-							
2	Opcode* (mirrored)							
3	T**	Return value***						
4	Response Parameter byte 0							
…	…							
max. 19	Response Parameter byte 15							

*Command code of the Mailbox job
**Toggle bit: The Mailbox job is started when there is a change.
***Status/error of the Mailbox job

The processing time in the 750-644 I/O module may require several bus cycles. Some specific commands trigger a longer process (e.g. scan for devices within range). For such commands, the response of the 750-644 I/O module confirms that the process has begun. The results of longer lasting processes can be queried after completion of the respective process by other commands.

The figure “Example of Mailbox communication” describes the request and processing of a Mailbox command. The process data are shown as follows:

[Parameter 0-x | Toggle | Opcode | int. Byte | Control/Status Byte]

First, any process data may be present in the output and input process image. The PLC application then maps the Mailbox by setting bit 5 in the control byte and the 750-644 I/O module responds with bit 5 in the status byte. Initially, no specific process is started, rather the entire request is written with value 0, i.e. IDLE. By the 750-644 I/O module responding to IDLE, it indicates its readiness to process other operations. Next, the PLC application starts a request in which a value is requested by the 750-644 I/O module. The PLC application writes the modified opcode and associated parameter data to the PIO. The 750-644 I/O module responds with a mirrored opcode and requested value as parameter data of the response. By the PLC application then inverting the toggle bit, it requests the same value again from the 750-644 I/O module. The 750-644 I/O module again responds with a mirrored opcode and toggle bit, as well as the current value in the parameter data of the response. The PLC application then hides the Mailbox again.

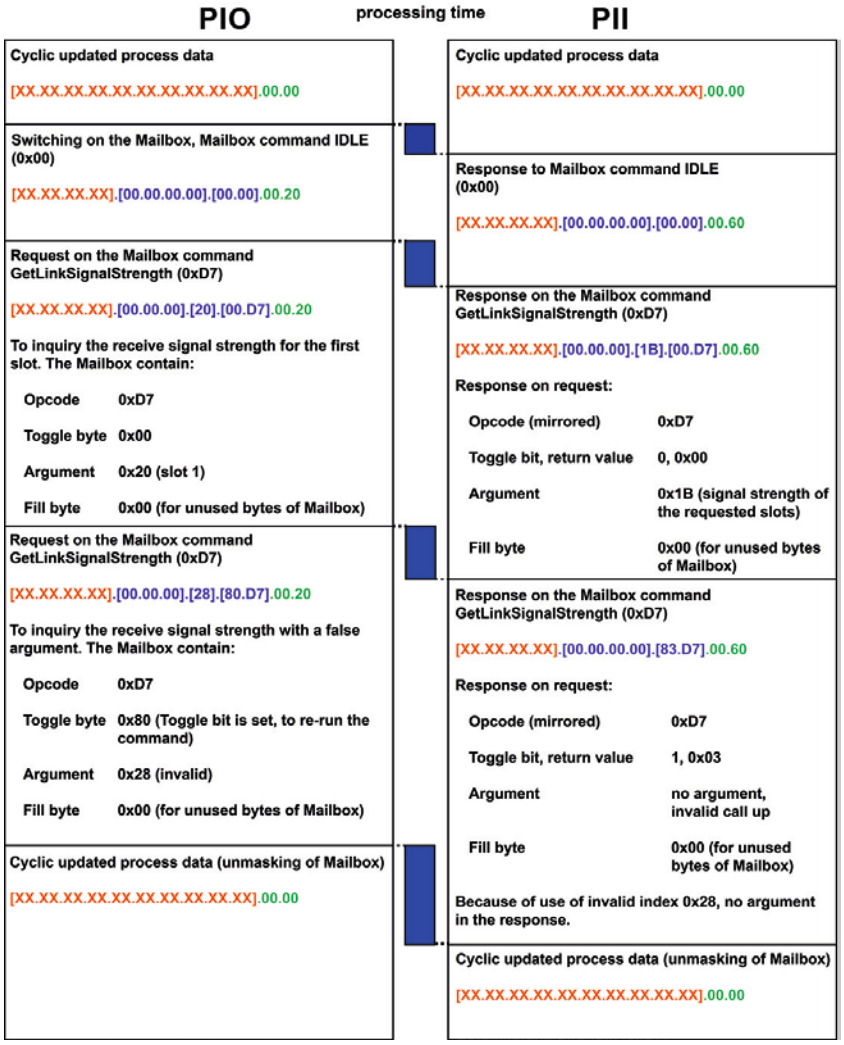


Figure 15: Example of Mailbox communication

A detailed description of all Mailbox commands is available in the appendix.

6.3.2 Process Data Overlaid by Mailbox

If the Mailbox is mapped, it overlays part of the process data based on its configured size. In “Ad-hoc Communication” and “Real-Time Communication” modes, other process data may not be valid. Such is the case when the process data of a slot is partially covered by Mailbox data.

While the Mailbox is mapped, the data of a partially overlaid slot is not updated in the PII even if the remote device assigned to this slot sends new data.

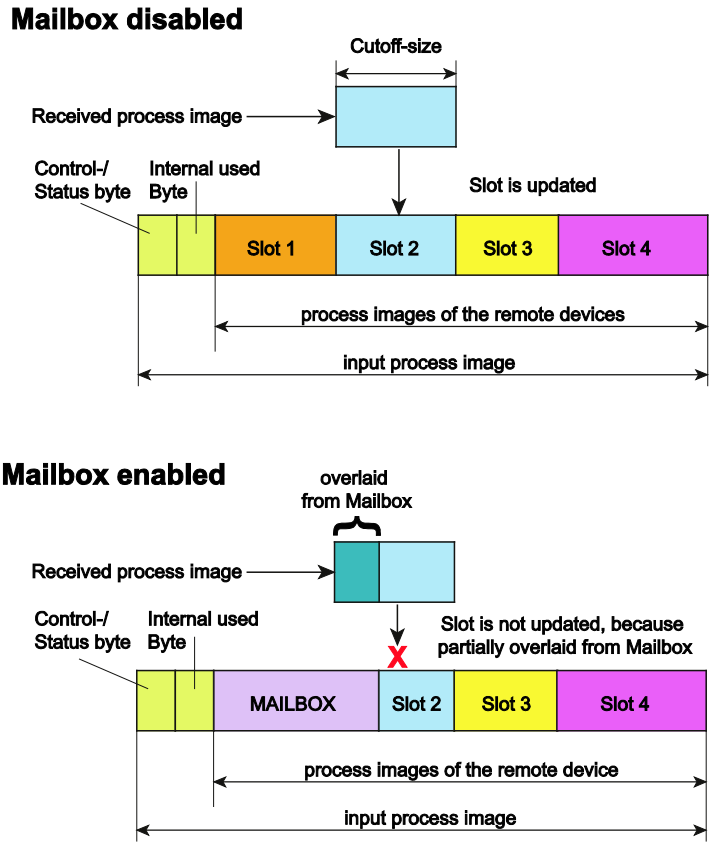


Figure 16: Partial overlay of process data by the Mailbox – PII

Likewise, the local 750-644 I/O module does not transmit the remaining data in the PIO of an entirely or partially overlaid slot to the remote 750-644 I/O module.

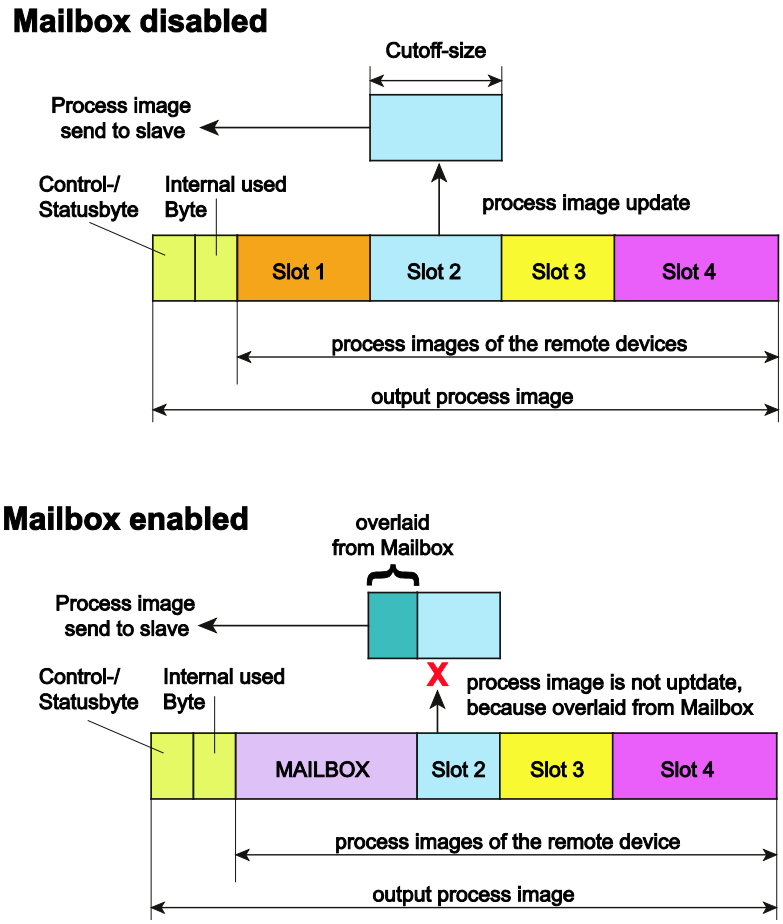


Figure 17: Partial overlay of process data by the Mailbox – PIO

If process data exchange is blocked by the Mailbox for a slot, the other side is notified. If a 750-644 I/O module receives a corresponding notification from the otherwise, it indicates such in the status byte of the local PII. The I/O modules must be in “Real-Time Communication” mode.

If you plan to use the Mailbox during ongoing communication, e.g. for diagnostic purposes, take note when mapping the Mailbox, process data may be temporarily overlaid. In such case, you can also carry out the configuration with the 750-644 I/O module configured as the master so that the first slot has no device allocated to it and the size of the first slow corresponds to the Mailbox size. This means, however, that one less slave can be linked, but the up-to-dateness of the process data is not dependent on the masking or unmasking of the Mailbox.

Note

Alternative configuration possible for master only!
This alternative configuration is only possible for the 750-644 I/O module that is configured as the master. Slave process images always consist of a single slot that immediately starts with the third byte.

6.3.3 Register Communication

Some I/O modules of the 750 series offer a protocol subsequently referred to as “register communication” for the configuration. Register communication provides read and/or write access to up to 64 data words designated as “registers” of the respective I/O module. This protocol allows, for example, data exchange between the 750-644 I/O module and a PLC application.

Register communication of the 750-644 I/O module is not a permanent part of the process image, but may be mapped as required. If mapped, the process image is only available for register communication, i.e. no other process data or Mailbox data can be transmitted.

If bit 7 is set in the control byte of the process image of the outputs, it is viewed as a request to the 750-644 I/O module to map register communication. The 750-644 I/O module responds to the request by setting bit 7 in the status byte. Register communication is thus considered mapped.

Note

Revoking Mailbox and process data validity
During register communication, the Mailbox and process data is no longer valid!

The register communication protocol uses bytes 0, 2 and 3 of the process image. All other bytes are not valid in this state.

Table 34: Setup of the process image for register communication

Byte	Word	Input process image	Output process image
0	0	Control byte C	Status byte S
1		Internal byte	Internal byte
2	1	D0	D0
3		D1	D1
4 … 47	2 … 23	Invalid	Invalid

If register communication is enabled, the control/status byte must be interpreted otherwise.

For register communication, the control byte is occupied as follows:

Table 35: Control byte C

Control byte C							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
1	RW	REG_NO					
REG_NO	Register number (e.g. 56 or 57)						
RW	0: Read access 1: Write access						

For register communication, the status byte is occupied as follows:

Table 36: Status byte S

Status byte S							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
1	0	REG_NO					
REG_NO	Register number (e.g. 56 or 57)						

All register communication processes follow the request/response principle:

1. A request is made in the output process image (PIO). This can be generated by a PLC application, for example.
2. The I/O module generates a response in the input process image (PII).

There is a request when the value of REG_NO or RW changes in the PIO compared to the previous status or register communication is remapped. If write access is involved, the I/O module also processes bytes D0 and D1 of the PIO as new values for register REG_NO.

There is a response when the value of REG_NO or D0 or D1 changes in the PII compared to the previous status. If it involves read access in the previous request, the D0 and D1 bytes then receive the current value of the REG_NO register.

6.3.4 Parameter Channel

Some I/O modules of the 750 series offer a protocol subsequently referred to as “parameter channel” for the configuration. Parameter channel provides read and/or write access to up to 256 parameters, i.e. data words of the respective I/O module. The protocol allows, for example, data exchange between the 750-644 I/O module and a PLC application.

The parameter channel uses register communication to transport data:

- Register 56: Parameter data is stored here word by word.
- Register 57: Communication is controlled for the data.

The basic sequence for reading a parameter is:

1. Write register 57 with the read request command.
2. Read register 57 to check the execution status.
3. If the read request was successful: Read register 56 to obtain the requested data.

The basic sequence for writing a parameter is:

1. Write register 56 to store the data of the write request.
2. Write register 57 with the write request command.
3. Read register 57 to check the execution status.

A more detailed description is available in section “Process Image” > “Process of Parameter Transmission”.

6.3.4.1 Parameter Data (Register 56)

Register 56 contains the parameter data to be read or written. Depending on the access type, either the 750-644 I/O module (read parameters) or the fieldbus coupler/controller (write parameters) writes data to register 56.

Table 37: Register 56

Register 56								
Bit	**2^7**	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**
Parameters	PRM7	PRM6	PRM5	PRM4	PRM3	PRM2	PRM1	PRM0
Bit	**2^{15}**	**2^{14}**	**2^{13}**	**2^{12}**	**2^{11}**	**2^{10}**	**2^9**	**2^8**
Parameters	PRM15	PRM14	PRM13	PRM12	PRM11	PRM10	PRM9	PRM8

PRM0 … PRM15: Parameter data bit 2^0 … bit 2^{15}

6.3.4.2 Communication Control (Register 57)

Parameter channel control and diagnostics are done via register 57.

Table 38: Register 57

Register 57								
Bit	**2^7**	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**
Request parameter	A7	A6	A5	A4	A3	A2	A1	A0
Response parameter	A7	A6	A5	A4	A3	A2	A1	A0
Bit	**2^{15}**	**2^{14}**	**2^{13}**	**2^{12}**	**2^{11}**	**2^{10}**	**2^9**	**2^8**
Request parameter	TGL_MS	PRM_RW	MORE_PRM	RES	RES	RES	RES	RES
Response parameter	TGL_SM	TIMEOUT	BUF_OVF	PRM_ERR	RES	RES	RES	RES

Request parameter: Information is written by the application and read by the 750-644 I/O module.

Response parameter: Information is written by the 750-644 I/O module and read by the application.

Parameters	Value range	Description
A0 … A7	0 … 255	Word address of the parameter to be read / to be written.
TGL_MS	FALSE, TRUE	Toggle bit to release new instructions from the application to the 750-644 I/O module. If TGL_SM and TGL_MS have the same status, no new instruction has been released yet. If the flags have different statuses, a new instruction has been released and is currently being processed.
PRM_RW	FALSE	Parameter data of A7 ... A0 is read.
	TRUE	Parameter data is written to A7 ... A0.
MORE_SM	FALSE	End of parameter transmission.
	TRUE	More parameter data to follow.
TGL_SM	FALSE, TRUE	Toggle bit indicating that a parameter sent by the 750-644 I/O module has been transferred. If TGL_SM and TGL_MS have different statuses, the corresponding instruction is processed by the 750-644 I/O module. If both flags have the same status, the instruction for the parameter that was sent or requested is completed.
TIMEOUT	FALSE	The transmission of the parameters has been completed within the stipulated time (parameter address 0).
	TRUE	The maximum time for the transmission of the parameters between 750-644 I/O module and application was exceeded.
BUF_OVF	FALSE	Access to the write and read buffer of the 750-644 I/O module were permitted.
	TRUE	Parameters outside of the write or read buffer were accessed.
PRM_ERR	FALSE	The parameter/all parameters previously transmitted are valid.
	TRUE	At least one transmitted parameter was defective. The flag can either be set after each parameter that is received or after the transmission of the parameters is completed.
RES	FALSE	Reserved for expansions.

6.3.4.3 Parameter Sets

For use of the parameter channel, parameter sets are defined and indexed using parameter addresses (A7 ... A0). Module-specific parameters (parameters 0 through 249) and general system parameters (parameters 250 through 255) are differentiated.

6.3.4.4 Process of Parameter Transmission

Parameter data is exchanged between the application and 750-644 I/O module via request/response process. The application initiates an instruction using the toggle bit (TGL_MS != TGL_SM). The application then queries the communication control register (R57) of the 750-644 I/O module until the I/O module confirms execution of the instruction (TGL_SM == TGL_MS).
The possible instructions to the parameterization interface of the 750-644 I/O module are listed in below:

Calculate the Maximum Parameter Data of the Bus Module (System Parameters)

Request (Application)

Parameters	Value	Description
TGL_MS	!= TGL_SM	Enter instruction
PRM_RW	= FALSE	Read access
A0 … A7	255	Address parameter data length

Response (I/O Module)

Parameters	Value	Description
TGL_MS	= = TGL_SM	Enter instruction
A0 … A7	255	Read access
PRM0 … PRM15	N	Number of parameter data in the address range 0 … (n-1), n ∈ {N < 250}

Set Factory Settings (System Parameters)

Request (Application)

Parameters	Value	Description
TGL_MS	!= TGL_SM	Enter instruction
PRM_RW	= FALSE	Write access
A0 … A7	255	Factory setting

Response (I/O Module)

Parameters	Value	Description
TGL_MS	= = TGL_SM	Instruction completed
A0 … A7	255	Set address factory setting, mirrored

By writing 255 to the parameter address, the factory setting of the I/O module subsystem for the 750-644 I/O module is restored. This includes the process image and Mailbox size. The settings of the *Bluetooth*® subsystem can only be accessed through the mailbox interface and can be separately reset to standard values via mailbox command.

Read/Write Parameters (Module-Specific)

Request (Application)

Parameters	Value	Description
TGL_MS	!= TGL_SM	Enter instruction
PRM_RW	= FALSE	Read access
	= TRUE	Write access
MORE_PRM	= FALSE	Parameter data transmission is completed.
	= TRUE	More parameter data follows
A0 … A7	0 … (n-1)	Parameter address
PRM0 … PRM15	0 … 65535	Parameter data write access

Response (I/O Module)

Parameters	Value	Description
TGL_MS	= = TGL_SM	Instruction completed
A0 … A7	0 … (n-1)	Address parameter data mirrored
TIMEOUT	FALSE, TRUE	Monitoring time expired
BUF_OFL	FALSE, TRUE	Access outside the 750-644 I/O module parameter range
PRM_ERR	FALSE, TRUE	Parameter / parameter set error
PRM0 … PRM15	0 … 65535	Parameter data read access

The 750-644 I/O module uses the error flags TIMEOUT, BUF_OV and PRM_ERR to report errors during the parameter data exchange.

After the last parameter data has been sent to the 750-644 I/O module (MORE_PRM = FALSE), the 750-644 I/O module checks the entire parameter set and accepts it if everything is correct. Otherwise, the 750-644 I/O modules returns parameterization errors (PRM_ERR = TRUE).

Example: Configuring *Bluetooth*® Process Data and Mailbox

The user can only change parameter 0 of the 750-644 I/O module. This includes configuration of the process image and Mailbox size.

Request (Application)

Parameters	Value	Description
TGL_MS	!= TGL_SM	Enter instruction
PRM_RW	= TRUE	Write access
MORE_PRM	= FALSE	Parameter data transmission is completed.
A0 … A7	0	Parameter address
PRM0 … PRM7	DATA_LEN	12, 24 or 48 byte data length
PRM8 … PRM14	MBX_LEN	6, 12 or 18 byte Mailbox size
PRM15	MBX_MODE	1 (not adjustable)

Response (I/O Module)

Parameters	Value	Description
TGL_MS	= = TGL_SM	Instruction completed
A0 … A7	0	Address parameter data mirrored
TIMEOUT	FALSE, TRUE	Monitoring time expired
BUF_OFL	FALSE, TRUE	Access outside the 750-644 I/O module parameter range
PRM_ERR	FALSE, TRUE	Parameter / parameter set error

7 Mounting

7.1 Mounting 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 installation.

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

⚠ CAUTION

Risk of injury due to sharp-edged blade contacts!
The blade contacts are sharp-edged. Handle the I/O module carefully to prevent injury.

NOTICE

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

Note

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

7.2 Inserting and Removing Devices

NOTICE

Perform work on devices only if the system is de-energized!
Working on devices when the system is energized can damage the devices. Therefore, turn off the power supply before working on the devices.

7.2.1 Inserting I/O Module

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

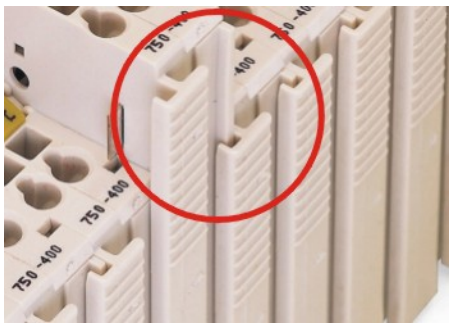


Figure 18: Insert I/O module (sample)

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

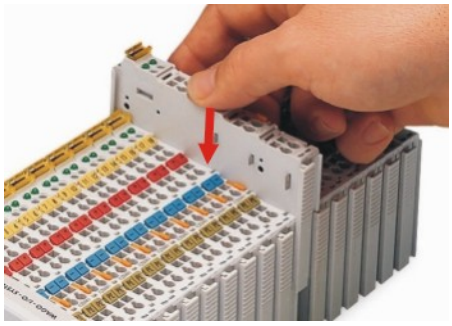


Figure 19: Snap the I/O module into place (sample)

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

7.2.2 Removing the I/O Module

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

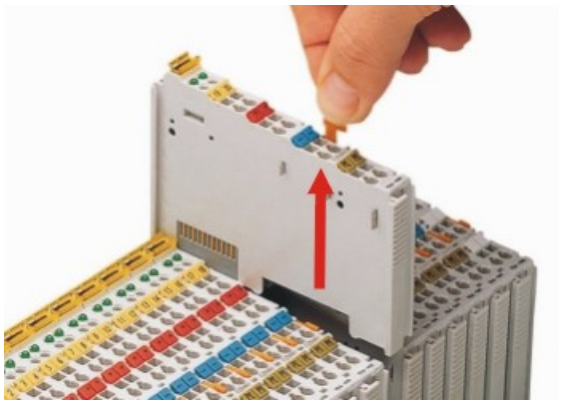


Figure 20: Removing the I/O module (sample)

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

7.3 Mount Antenna

The 750-644 I/O module has an SMA socket for attaching an external antenna.

1. Mount the respective antenna based on your method for mounting.

NOTICE

Set up lightning protection when using an external antenna!
When using an external antenna located outside the building, appropriate measures must be taken to protect against lightning.

Information

Installation instructions when using antenna 758-912
Antenna 758-912 may be used in an environment that meets the IP20 degree of protection. The integrated magnetic base can be used to secure it to a suitable metallic surface. The surface selected should be level in a radius of min. 12 cm around the place of installation. The absence of such a reference surface can negatively affect transmission and reception.

2. Then lay the connecting cable of the antenna and note the bend radius. If external forces can act on the connecting cable: Make sure there is adequate strain relief.

3. Then attach the connecting cable to the SMA socket of the 750-644 I/O module.

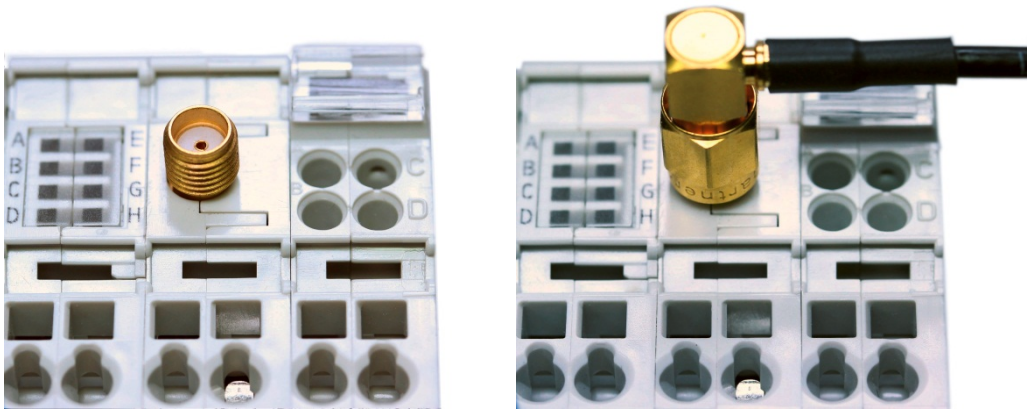


Figure 21: Connection to the SMA socket

When selecting and mounting the antenna, please note the instructions in section “Device Description” > ... > “Antenna”.

7.3.1 Scope of Application

Note

Please note the scope of application!

- When used in the scope of application of the FCC, the magnetic base antenna from WAGO may be used exclusively (item No.: 758-912).
- When used in the scope of application of the R&TTE, any antenna with antenna gain of max. 0 dBi may be used.

7.3.2 Range in Open Area

The maximum distance that can be overcome by a radio link is defined by the following factors:

1 Input Sensitivity

This denotes the capability of the device hardware to detect the radio signal transmitted by the remote device. The greater the sensitivity, the more weaker signals that can be received.

→ This is a permanent device property.

2 Transmitting Power

This denotes the signal strength which the device hardware outputs/can output for transmitting.

→The maximum transmitting power for the 750-644 I/O module can be set as a parameter. Transmitting power should be set to the highest level for the maximum range.

3 Antenna Gain

This factor denotes the focusing or bundling properties of the antenna. An antenna with high antenna gain exhibits a strong alignment characteristic, i.e. depending on the antenna alignment, only highly amplified or extremely attenuated signals can be received.

→That means that the reception properties can be greatly influenced by correct alignment. The best results are achieved when the antennas are aligned in parallel and on the same level (see also section “Device Description” > ... > “Antenna”).

Note

Antenna gain of the antenna 758-921

The magnetic base antenna from WAGO (item No. 758-912) itself exhibits antenna gain of 2 dBi. Due to its cable length, the effective antenna gain is only 0 dBi, thus also meeting the scope of application of the R&TTE.

4 Ambient Conditions

This factor deals with the physical environment/area around the wireless system. To achieve the maximum range, a line-of-sight link should exist between the antennas and there should be no objects present along the direct

line-of-sight link within a specified radius (the so-called 1st Fresnel zone – see figure “Fresnel zone”). If this zone is even only partially blocked by any objects the achievable range can quickly be cut in half.

To achieve maximum range when using WAGO antenna 758-912, the following conditions must be met:

- “Real-Time Communication” mode is used.
- The transmission power is set to the maximum value allowed at the place of installation (observe local and national codes).
- The antennas are parallel to each other.
- The antennas are on the same level, i.e. a straight line connecting the centers of the antennas is a right angle to the longitudinal axis of each antenna.
- There is a direct line-of-sight link between the antennas in which a radius according to the 1st Fresnel zone is free of objects along the line-of-sight.

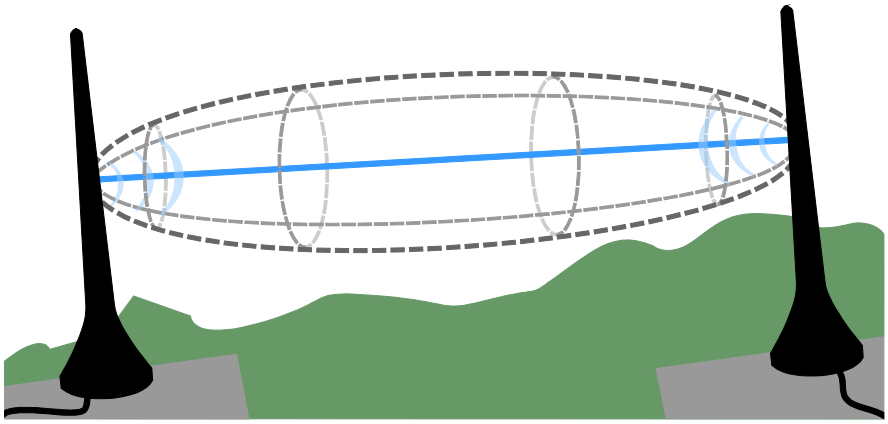


Figure 22: Fresnel zone

The shape of the 1st Fresnel zone is roughly an ellipse whose largest diameter (twice the radius of the 1st Fresnel zone) is at precisely half the distance.

The table below lists the radii that are to be kept clear:

Table 39: Radii to be kept clear

Distance	Radius 1st Fresnel zone
100 m	1.7 m
200 m	2.5 m
300 m	3.0 m
400 m	3.5 m
500 m	3.9 m
750 m	4.8 m
1.000 m	5.5 m

Note

Range can be affected by other wireless systems!
Other, more difficult to detect factors to be excluded influencing the range are caused by interference from other radio systems or if the radio changes over time. Such influences, for example, can be station movements (swaying of the mast assembly in the wind) or other environmental changes (vehicles, movement of stored goods, movement of people). These factors can make a precise prediction of the maximum range extremely difficult.

Note

Effective path of radio waves is more significant than the actual distance!
Overcoming of maximum distances for a line-of-sight link is a scenario completely different from use inside rooms or in the direct vicinity to competing wireless technologies. In these types of scenarios the decisive factor is frequently not the actual distance between the devices, but, rather, the effective path taken by the radio waves for multipath propagation, along with the actual interference present at the exact installation location. Under some circumstances, signal quality may even be enhanced by increasing the distance between the devices.

8 Commissioning

8.1 Configuration and Parameterization with WAGO-I/O-*CHECK*

The 750-644 I/O module can be configured using the WAGO-I/O-*CHECK* software (version 3 or higher). The software's basic functionality is described separately in the WAGO-I/O-*CHECK* documentation.

Information

Additional information

You receive the WAGO-I/O-*CHECK* on a CD-ROM under order ID 750-302. The CD-ROM includes all program files for the application. The documentation for the WAGO-I/O-*CHECK* software is available on the Internet at **http://www.wago.com** under *Downloads > Additional Documentation and Information for Automation Products > WAGO Software > WAGO-I/O-CHECK*.

Open the specific configuration dialog for the 750-644 I/O module by right-clicking on the figure for a 750-644 I/O module and selecting **Settings** in the context menu.

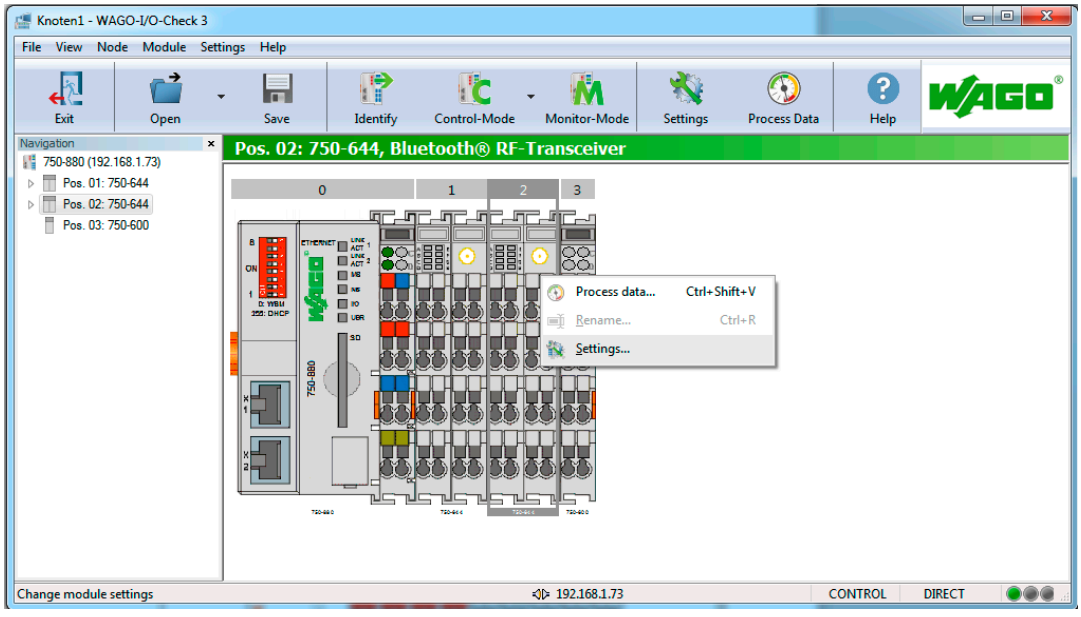


Figure 23: WAGO-I/O-*CHECK* user interface

The content of the configuration dialog forms the basis for the following description.

The user interface of the *Bluetooth*® parameterization dialog is divided into the title bar, toolbar, navigation, operating mode / role assignment, parameterization and status display:

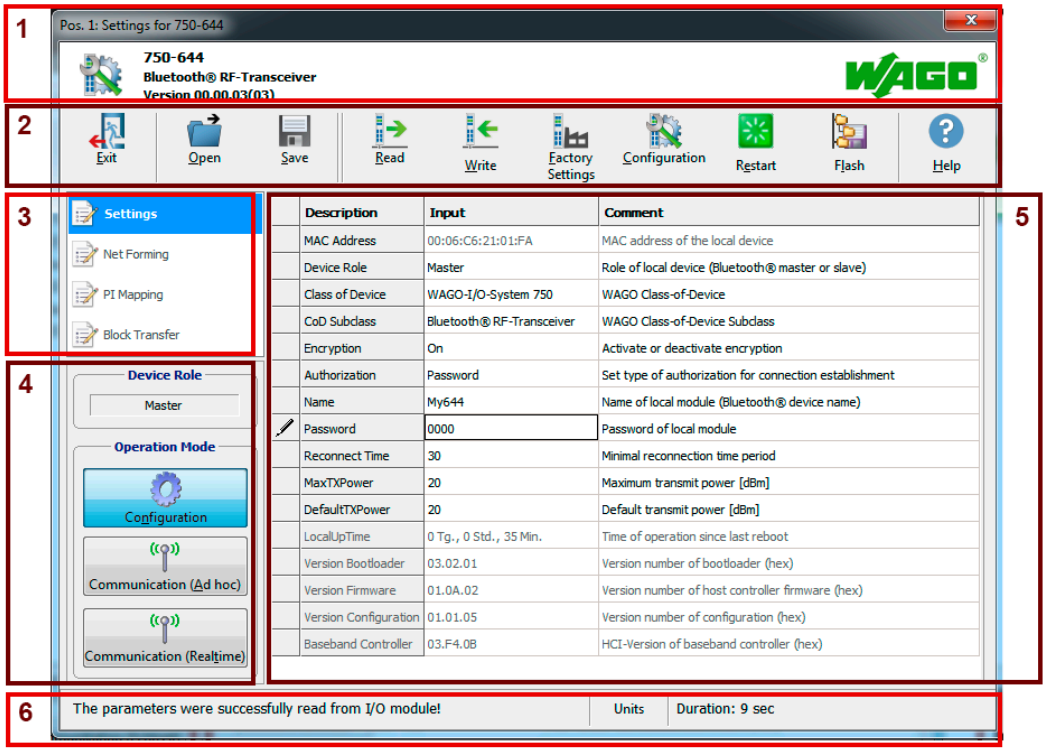


Figure 24: User interface of the *Bluetooth*® parameterization dialog

Table 40: Legend for the user interface of the *Bluetooth*® parameterization dialog

Range	Description
1	Title bar (see section “Title Bar”)
2	Toolbar (see section “Toolbar”)
3	Navigation (see section “Navigation”)
4	Operating mode and role assignment (see section “Operating Mode and Role Assignment”)
5	Parameterization area (see section “Parameterization Mode”)
6	Status display area (see section “Status Display”)

These areas will be explained in more detail in the following sections.

8.1.1 Title Bar

The position of the 750-644 I/O module within the node as well as its name, item and version number is displayed in the title bar of the parameterization dialog.

8.1.2 Toolbar

The toolbar *Bluetooth*® parameterization dialog contains the following buttons:

Figure 25: Buttons in the *Bluetooth*® parameterization dialog

Table 41: Buttons in the *Bluetooth*® parameterization dialog

Button	Description
	**[Exit]** closes the active window. If you have changed settings, you are asked to save these values in the 750-644 I/O module.
	**[Open]** opens a window to select a parameter file. Device settings are read from the parameter file and transferred to the connected 750-644 I/O module.
	**[Save]** opens a window to select a parameter file. The device settings are saved to the parameter file.
	**[Read]** Reads the current settings from the 750-644 I/O module and displays them in this window.
	**[Write]** transfers the settings displayed in this window to the connected 750-644 I/O module.
	**[Factory Settings]** overwrites the locally-saved configuration with factory settings.
	**[Configuration]** opens the data frame dialog. There you can set the process image size and Mailbox size.
	**[Restart]** starts the host controller again. **NOTICE:** All radio connections are broken off.
	**[Flash]** writes the current configuration of the host controller to the flash memory and restarts it. **NOTICE**: All radio connections are broken off.
	**[Help]** displays help for this window.

8.1.3 Navigation

You can toggle between the different configuration areas of the 750-644 I/O module by using the navigation on the left side of the screen:

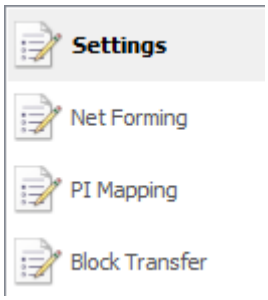


Figure 26: Navigation between configuration areas

You can choose from the following menu items that display their own pages in the parameterization area:

Table 42: Navigation between configuration areas

Menu item	Description
Settings	Opens a page with general I/O module parameters such as device name, MAC address, device role, etc. These parameters can be modified here and loaded to the 750-644 I/O module (see section “Settings”).
Net Forming	Opens a page with device lists. Configured and linked devices within range are displayed with MAC address and name and configured (see section “Net Forming”).
PI Mapping	Opens a page for the allocation of slave process data to slots in the master (see section “PI Mapping”).
Block Transfer	Opens a page for viewing the process data while uploading and downloading. The “Block Transfer” menu item is only displayed in “Configuration” mode (see section “Block Transfer”).
Diagnostics	Opens a page with comprehensive diagnostic information on the status of the 750-644 I/O module and the network, as well as the connection quality (see section “Diagnostics”). **NOTICE:** The **Diagnostics** menu item only appears in “Communication” mode.

8.1.4 Operating Mode and Role Assignment

The Device Role area at the top displays of the configured 750-644 I/O module takes the role of the master or the role of a slave.

The Operating Mode area at the bottom is used to assign an operating mode to the local 750-644 I/O module. Use the buttons to choose whether the 750-644 I/O module should be operated in “Configuration” mode or in one of the two “Communication” modes (“Real-Time” or “Ad-hoc”):

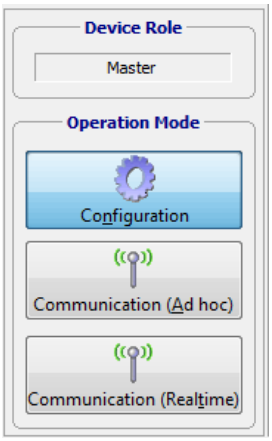


Figure 27: Operating mode and role assignment area

You can choose between the following buttons:

Table 43: Navigation between configuration pages

Status field / button	Description
Device Role	
"Slave" or "Master"	Displays the device role of the 750-644 I/O module currently assigned.
Operation Mode	
[Configuration]	Switches the local 750-644 I/O module to "Configuration" mode.
[Communication (Ad hoc)]	Switches the local 750-644 I/O module to "Ad-hoc Communication" mode.
[Communication (Realtime)]	Switches the local 750-644 I/O module to "Real-Time Communication" mode.

8.1.5 Parameterization Mode

The 750-644 I/O module is configured and prepared for communication in the parameterization area. This is described in further detail in the following chapters.

Changing and Saving Data

To change the settings of the 750-644 I/O module, adjust the values on the **Settings** page in the **Parameterization area**. Modified settings are labeled with a change icon  , which indicates that the displayed values are no longer the same as the originally queried values of the 750-644 I/O module. To transfer the new values to the 750-644 I/O module, click the **[Write]** button in the toolbar. The change icons go away. During the write operation, the values from the 750-644 I/O module are first saved in volatile memory, so that the changes can be discarded again by clicking the **[Restart]** button in the toolbar. When discarding changes in this way, you have to refresh the graphical display of the values after the 750-644 I/O module restarts by clicking **[Read]** in the toolbar.

To save value changes to non-volatile memory (flash process), you can either click the **[Flash]** button or switch the 750-644 I/O module into a different

operating mode. When switching the operating mode, transmitted changes are automatically stored in non-volatile memory.

For example, you can switch the 750-644 I/O module to “Communication (Real-Time)” mode after completing the configuration under the **Net Forming** menu item. The modified configuration is saved and the 750-644 I/O module immediately attempts to exchange data with the configured partner devices.

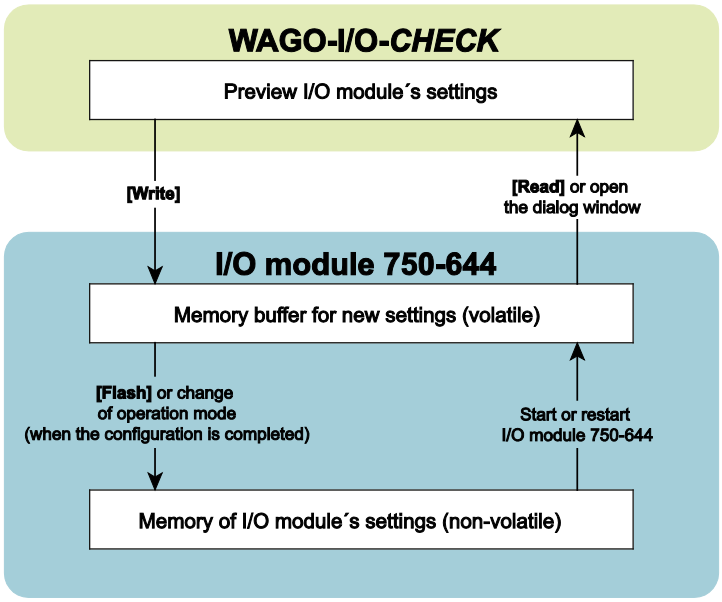


Figure 28: Saving the configuration

8.1.5.1 Settings

The *Settings* page displays general I/O module parameters:

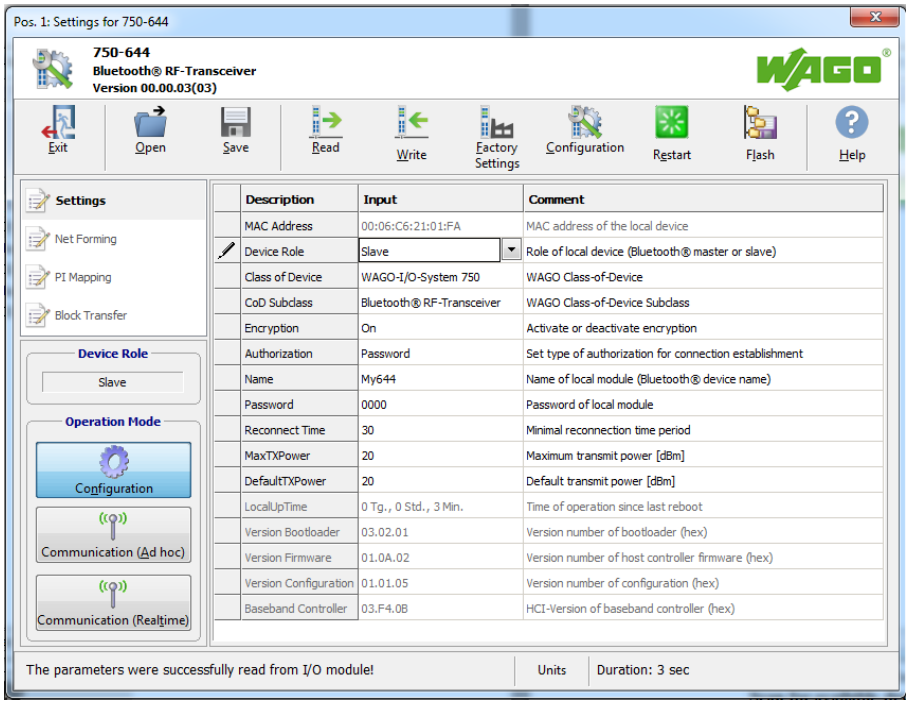


Figure 29: Screenshot of the *Settings* page

The following parameters can be changed and loaded onto the 750-644 I/O module:

Table 44: Navigation between configuration pages

Description	Input/selection	Description
MAC Address	_ _ : _ _ : _ _ : _ _ : _ _ : _ _	MAC address of the device
Device Role	Master	Assignment of the master role
	Slave	Assignment of the slave role
Class of Device	WAGO *SPEEDWAY* 767	WAGO-Class-of-Device (for *Bluetooth*®, "WAGO-I/O-SYSTEM 750" is set)
	WAGO-System 763	
	WAGO-I/O-SYSTEM 750	
	WAGO-I/O-SYSTEM 753	
	WAGO-I/O-SYSTEM 755	
	WAGO-I/O-SYSTEM 757	
CoD Subclass	*Bluetooth*® RF Transceiver	WAGO CoD subclass
Encryption	On	Switch on encryption
	Off	Turn off encryption
Authorization	Off	No authorization required
	Password	For *Bluetooth*® SPP devices, password entry is required. The "Link Key" for authorization must be recalculated for each connection established.
	Link Key	The "Link Key" for authorization does not have to be recalculated. After connecting for the first time, password entry is no longer required for *Bluetooth*® SPP devices.
Name	Input as ASCII characters, length depends on Mailbox size (max. 16 characters)	Name of the local 750-644 I/O module (*Bluetooth*® device name)
Password	Input as ASCII characters, length depends on Mailbox size	Password of the local 750-644 I/O module
Reconnect Time		Minimum time between two attempts to connect in seconds
MaxTXPower		Maximum transmission power in dBm
DefaultTXPower		Default transmission power in dBm
LocalUpTime	_ _Day,_ _Hr.,_ _Min.	Operating time of the 750-644 I/O module since the last reboot
Version Bootloader	_ _ . _ _ . _ _	Version number of bootloader
Version Firmware	_ _ . _ _ . _ _	Version number of host controller firmware
Version Configuration	_ _ . _ _ . _ _	Version number of configuration
Baseband Controller	_ _ . _ _ . _ _	HCI version of the baseband controller

8.1.5.2 Net Forming

"Net Forming" refers to the structure of the *Bluetooth*® network. On the **Net Forming** page, the devices are filled in manually or searched automatically. They are bind for communication at a later stage.

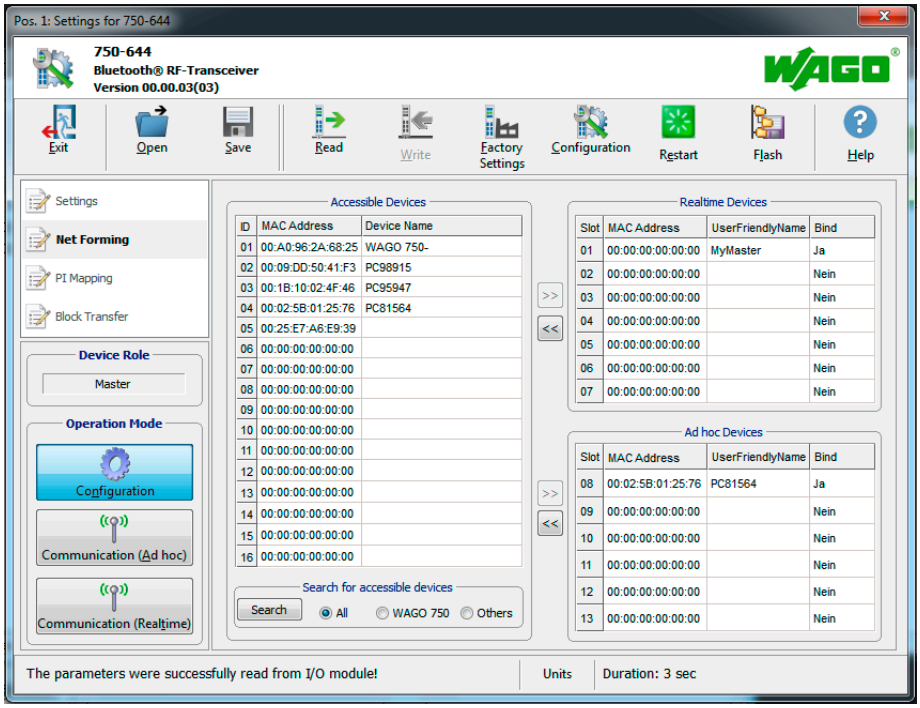


Figure 30: Screenshot of the *Net Forming* page

All available devices are listed on the **Net Forming** page on the lift side. You can refresh the list by clicking the **[Search]** button. Depending on the option selected, you can limit device discovery. The Class-of-Device (CoD) is used as a criterion for filtering scan results.

- Select **All** to search for all available *Bluetooth*® devices in the area.
- Select **WAGO 750** to search for all available WAGO devices of the 750 series.
- Select **Other** to manually define by which CoD the search results should be filtered. The “Class of Device” dialog appears:

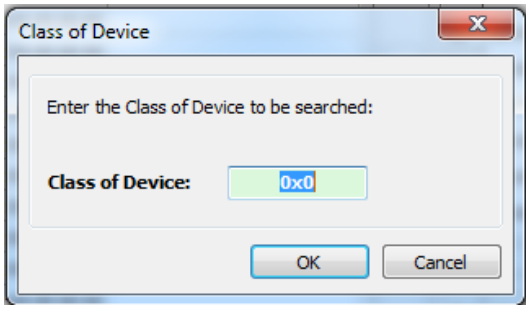


Figure 31: The *Class of Device* dialog for filtering by device class

- The configured devices are displayed on the right side of the **Net Forming** page in two lists:
  - The top list contains 750-644 I/O modules in “Real-Time Communication” mode.
  - The bottom list contains both *Bluetooth*® SPP devices and 750-644 I/O modules in “Ad-hoc Communication” mode in range.

- Use the **[>>]** button to move the selected devices to the real-time or ad-hoc list. MAC IDs or devices names can be moved from the list of available devices to the ad-hoc or real-time list by drag & drop.
- Use the **[<<]** button to remove the configured devices from the real-time or ad-hoc list. You can also double-click the respective MAC ID to delete the devices.

The tables on the **Net Forming** page are populated as follows:

Table 45: Table identifiers in *Net Forming*

Designation	Input/selection	Description
ID	_ _	ID of the device in available devices
MAC Address	_ _ : _ _ : _ _ : _ _ : _ _ : _ _	MAC address of the device
Device Name	ASCII characters	Device name (cannot be changed)
Slot	_	Slot number of the assigned devices
UserFriendlyName	ASCII characters	Name assigned to a slot (can be changed)
Bind	Yes	Bind device (“Yes”)
	No	Do not bind device (“No”)

Note

UserFriendlyName
Please note when assigning a “UserFriendlyName” that to display the entire length of the name, a Mailbox size of 18 bytes is necessary. With a smaller Mailbox setting, the full name is actually displayed within WAGO-I/O-*CHECK*, but not completely saved, so when the name is read back from the 750-644 I/O module, not all the characters are displayed.

8.1.5.3 PI Mapping

To make settings on the **PI Mapping** page, the process image size of the master must first be set.

Use the **[Configuration]** button in the toolbar to open the “Data Frame” dialog for entering the process image and Mailbox sizes:

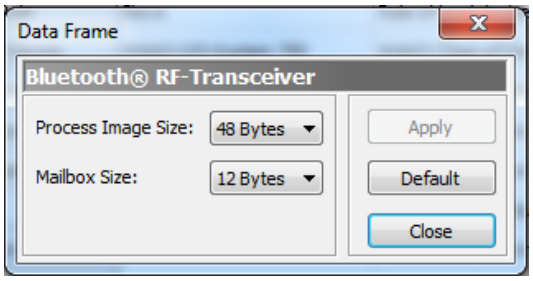


Figure 32: *Data Frame* dialog

You can make the following settings:

Table 46: Setting options in the *Data Frame* dialog

Selection box	Settings
Process Image Size	12 Bytes, 24 Bytes, 48 Bytes*
Mailbox Size	6 Bytes, 12 Bytes*, 18 Bytes

*Default setting

Button	Description
[Apply]	Transfers the modified parameters to the non-volatile memory of the 750-644 I/O module. A software reset is carried out, so that the changes take effect. The dialog remains open.
[Default]	Selects the default setting for this 750-644 I/O module. The **[Apply]** button then transfers the parameters to the non-volatile memory of the 750-644 I/O module.
[Close]	Closes the parameterization dialog without transferring any modified parameters to the non-volatile memory of the 750-644 I/O module.

Note

Changing the process image size
Please note that the structure of the process image changes when the process image size or Mailbox size is changed. Therefore, changes in the configuration of the superordinate controller may be necessary.

On the **PI Mapping** page, the slave process data is allocated to the slots in the master (see figure “Screenshot of the *PI Mapping* page”). Up to 46 bytes of the process image are available for this purpose (depending on which process image size was set in the “Data Frame” dialog). The control/status byte and internal byte are not taken into consideration here.

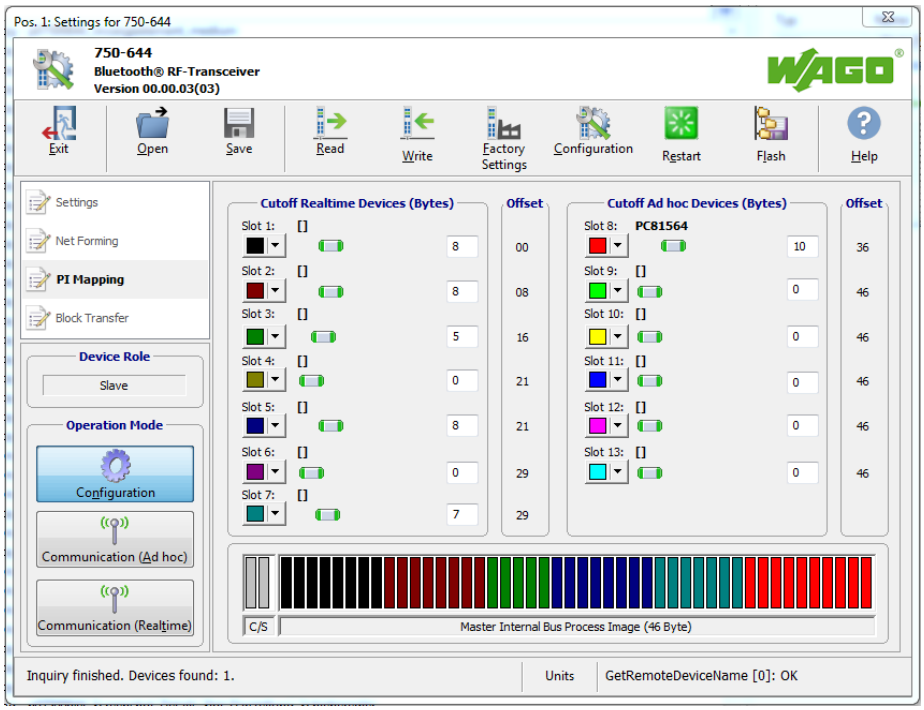


Figure 33: Screenshot of the *PI Mapping* page

Slots 1 to 7 for “Real-Time Communication” mode are displayed on the left side (for 750-644 I/O modules only). Slots 8 to 13 for “Ad-hoc Communication” mode are displayed on the right side (for 750-644 I/O modules and other *Bluetooth*® SPP devices). Each row designates one slot:

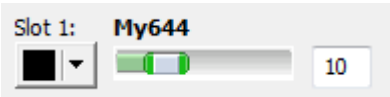


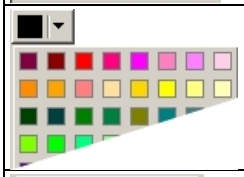
Figure 34: Representation of a slot

Note

WAGO-I/O-*CHECK* not adapted to current firmware!

The display of offset values and graphical display of the slot assignment for “Ad-hoc Communication” mode has not yet been adapted to the real values in the current software version of WAGO-I/O-*CHECK*!

Table 47: Representation of a slot

Setting	Description
Slot 1:	Slot identification (1 … 7 real-time, 8 … 13 ad-hoc)
Bluetooth_015	Display of the “UserFriendlyName”, if given
	Selection of the slot color for the graphical representation at the bottom (see figure “Screenshot of the *PI Mapping* page”)
	Slider for the size of the process data byte (cut-off size) assigned to a slot
8	Input field for the size of the process data byte (cut-off size) assigned to a slot
00	Offset in bytes to the start of the slot (no control/status byte or internal byte) **NOTICE:** Values for the ad-hoc devices are not currently displayed correctly.

The master process image is displayed graphically below the slot configuration with a breakdown of the slots:

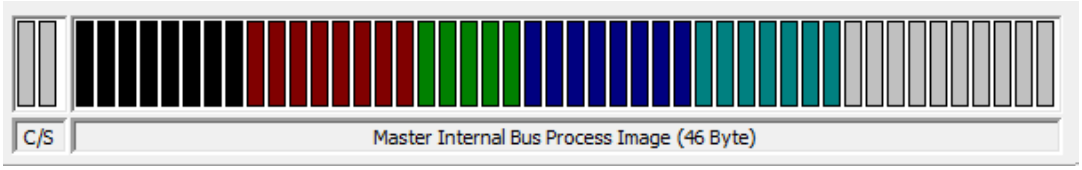


Figure 35: Slot assignment

8.1.5.4 Block Transfer

The **Block Transfer** page displays the configuration block when uploading and downloading of the process data. The **Block Transfer** menu item is only displayed in “Configuration” mode.

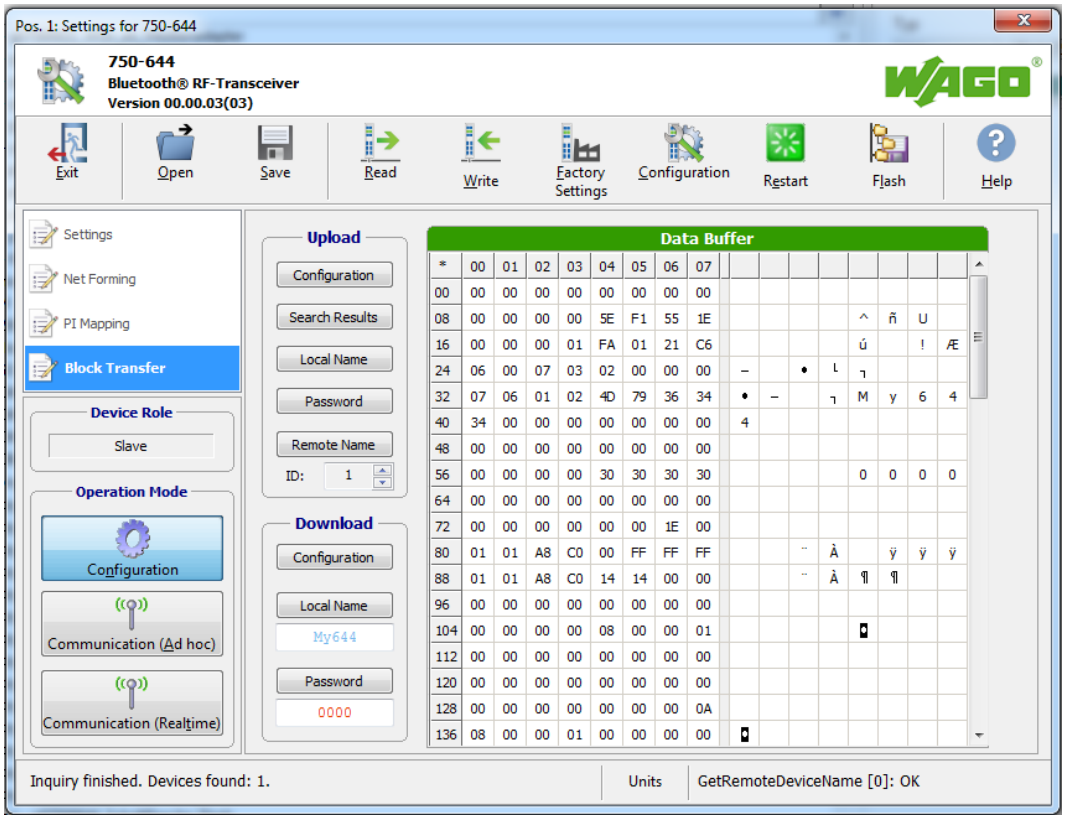


Figure 36: Screenshot of the *Block Transfer* page

You can select from among the following menu items:

Table 48: Block transfer

Menu item	Description
Upload	
[Configuration]	Displays the configuration transferred from the 750-644 I/O module to the application.
[Search Results]	Displays the list of MAC IDs found during a scan.
[Local Name]	Displays the complete local name of the 750-644 I/O module (in **Settings** menu item, the name may be incomplete due to insufficient Mailbox size).
[Password]	Loads the password set.
[Remote Name]	Displays the devices name of the linked 750-644 I/O module. Entering an ID displays the device name of a specific *Bluetooth*® SPP device (see section “Net Forming”).
Download	
[Configuration]	Writes the configuration to the 750-644 I/O module.
[Local Name]	Writes the local name to the 750-644 I/O module. The name can be entered in the input field.
[Password]	Writes the password to the locally linked 750-644 I/O module. The password can be entered in the input field.

8.1.5.5 Diagnostics

The **Diagnostics** page displays diagnostic information about the status of the 750-644 I/O module, the network and the connection quality. The **Diagnostics** menu item only appears in “Communication” mode.

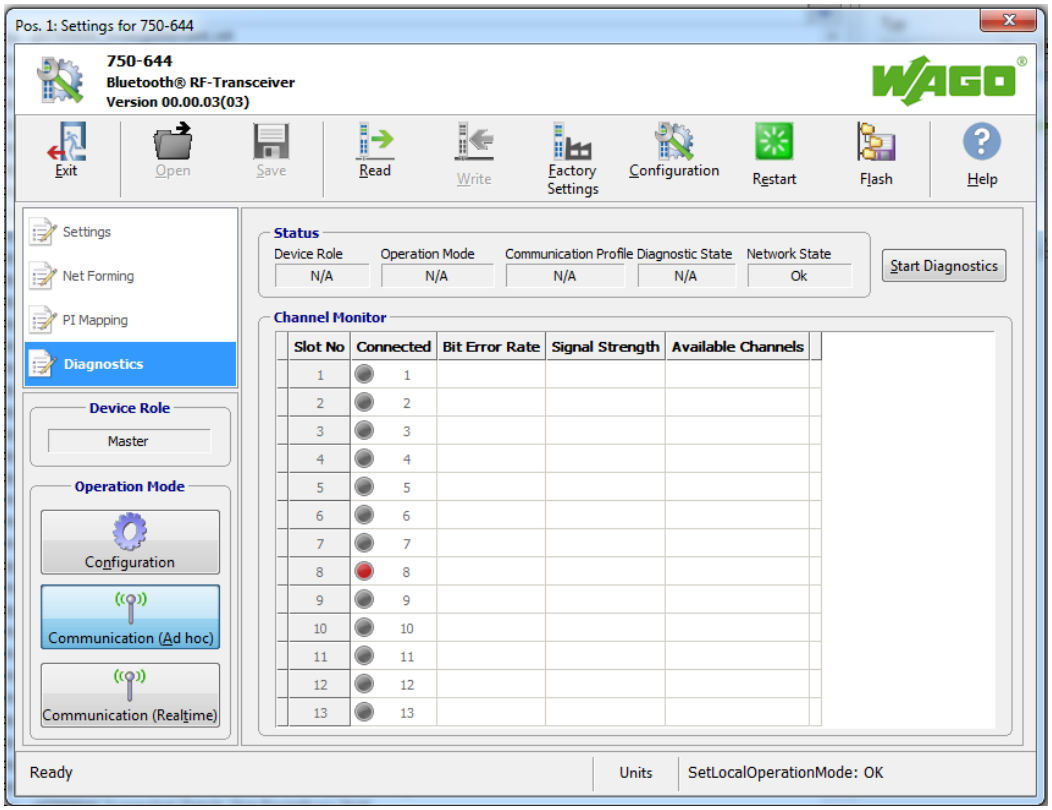


Figure 37: Screenshot of the *Diagnostics* page

The following status displays are summarized below the “Status” header:

Table 49: General status displays

Status	Value	Description
Device Role	Slave	Device take the “slave” role
	Master	Device take the “master” role (see also appendix “Read Role of the Local Device”)
Operation Mode	Communication	Device is in “Communication” mode (see also appendix “Read Local Operating Mode”)
Communication Profile	Real-time profile	Device is in “Real-Time Communication” mode
	Ad hoc profile	Device is in “Ad-hoc Communication” mode
Diagnostic State	Ok	No warnings / errors
	Warning	Warning
	Error	General error
	Critical defect	Critical defect (for details, see appendix “Read Status of the Local I/O Module”)
Network State	Ok	Configured network is established.
	Inconsistent	Not all configured connections are established.
	Faulty	Faulty, configured network is not (yet) established (for details, see appendix “Read Status of the Wireless Network”)
[Start Diagnostics]	Starting value monitoring	
[Stop Diagnostics]	Stopping value monitoring	

The link quality for each slot is displayed under “Channel Monitor”:

Table 50: Transmission channel status

Status	Value	Description
Slot No.	Slot _	Slot number
Connect	Yes	Connect
	No	Not connected
	No	No device configured for this slot
Bit error rate	0 %	No bit errors have occurred
	0.1 … 10 %	Some bit errors have occurred
	>10 %	High bit error rate
Signal strength	-127 … 0	RSSI value/signal strength too weak
	0	Signal strength very good
	0 … +127	Signal strength too strong (see appendix “Read Signal Strength for a Connection”)
Available channels	<39	Too many busy/faulty channels
	39 … 53	Some busy/faulty channels
	>53	Open/uninterrupted channels (low interference) (see appendix “Read Available Hopping Channels”)

If you click a field in the last table column, a dialog appears with the detailed status of the selected slot:

1. Mark the checkbox **for all connections** to query the status of all slots.

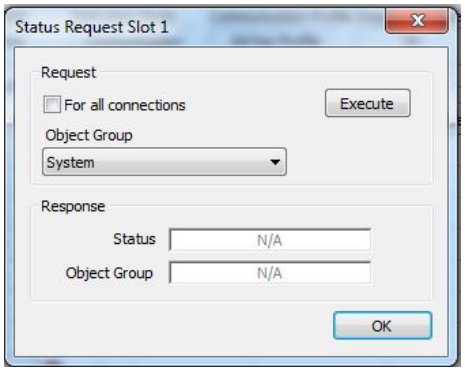


Figure 38: *Status query for slot 1* dialog

2. You can limit the status messages by selecting a specific **Object Group** (see table “Object groups and possible status messages”).

Table 51: Object groups and possible status messages

Object Group	Status*
System	Ok
Remote connections	Ok
	Not specified
	BER is moderate
	BER is high
	Less than 39 channels are available
	Less than 54 channels are available
	Connection errors
	Radio connections
	Connection interrupted
Timing	Ok
	Watchdog
Process Image	Ok
	PI is faulty
	A remote mailbox is enabled
Inter System Communication	Ok
	Error in Mailbox communication
Configuration	Ok
	Configuration changed
	Error in the network configuration

*The meanings of the individual status messages are available in the appendix.

3. Click the **[Execute]** button to query the status of the selected object group.

8.1.6 Status Display

The status messages are output in the **Status display area**. The display varies depending on which page is called up, e.g. *Settings*, *Net Forming*, *PI Mapping*, *Block Transfer* and *Diagnostics*.

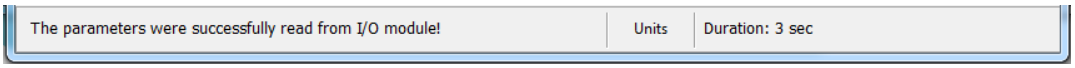


Figure 39: *Status Display* area

8.2 Configuration and Parameterization via Process Image

As an alternative to using WAGO I/O-*CHECK*, you can also use register communication, parameter channel and Mailbox to configure and parameterize the 750-644 I/O module. A more detailed description is available in section “Process Image”.

9 Example Configuration

9.1 Example Configurations via WAGO-I/O-*CHECK*

9.1.1 Startup with the *Bluetooth*® Parameterization Dialog

This section describes startup and configuration of 750-644 I/O modules using the WAGO-I/O-*CHECK* software.

The following example demonstrates startup for the minimum configuration and does not describe the entire functionality. The purpose of these instructions is to configure simple peer-to-peer communication between two 750-644 I/O modules. One 750-644 I/O module functions as the master and the other as a slave.

9.1.1.1 Network Configuration

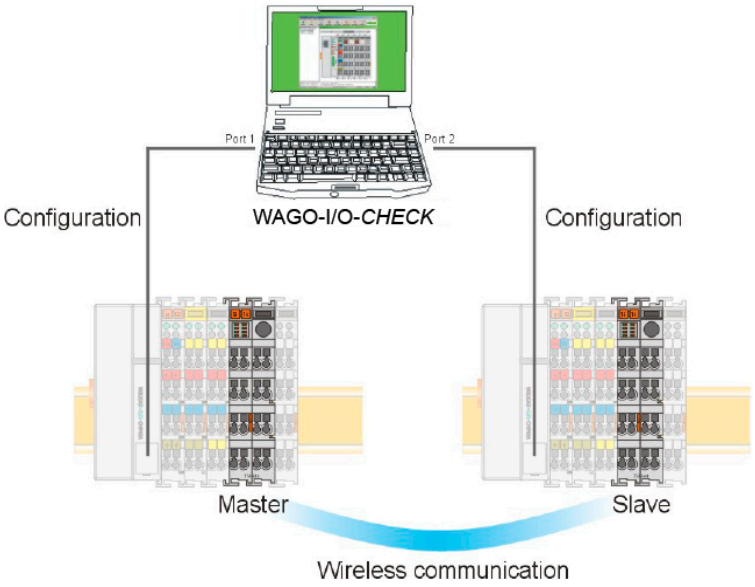


Figure 40: Hardware configuration

1. Similar to the figure “Hardware configuration”, set up two identical bus nodes:

   - 750-841 ETHERNET fieldbus controllers
   - 750-644 *Bluetooth*® RF Transceivers (750-644 I/O modules)
   - 750-600 end modules

2. Connect one of the fieldbus controllers to an open serial port on your PC using a WAGO communication cable (750-920).

3. Connect the second fieldbus controller in the same way to another PC serial port.

Note

Do not set up a fieldbus connection!
Do not set up a fieldbus connection (e.g. by using an ETHERNET cable).
Otherwise, access to the process data within WAGO-I/O-*CHECK* is not possible.

4. Connect both nodes on the field and system side to a 24 V power supply.

5. Switch on the power supply.

Information

Port availability
Each PC serial port is operated by its own WAGO-I/O-*CHECK* software. Depending on port availability, use one or two PCs to configure the 750-644 I/O modules. If you use one PC with 2 ports, you can launch the WAGO-I/O-*CHECK* software several times. You can select the COM ports concerned using the “F8” key on your keyboard. If only one port is used meaning that the software is launched just once, configuring the master and slaves is somewhat more time-consuming.

9.1.1.2 Startup of the 750-644 I/O Modules

1. Specify which of your 750-644 I/O modules should take the role of master and which the role of slave.

2. Note the MAC ID of the master: 0 0 : 0 6 : C 6 : _ _ : _ _ : _ _.
Note the MAC ID of the slave: 0 0 : 0 6 : C 6 : _ _ : _ _ : _ _ .

9.1.1.2.1 Configuring Slaves via “Net Forming”

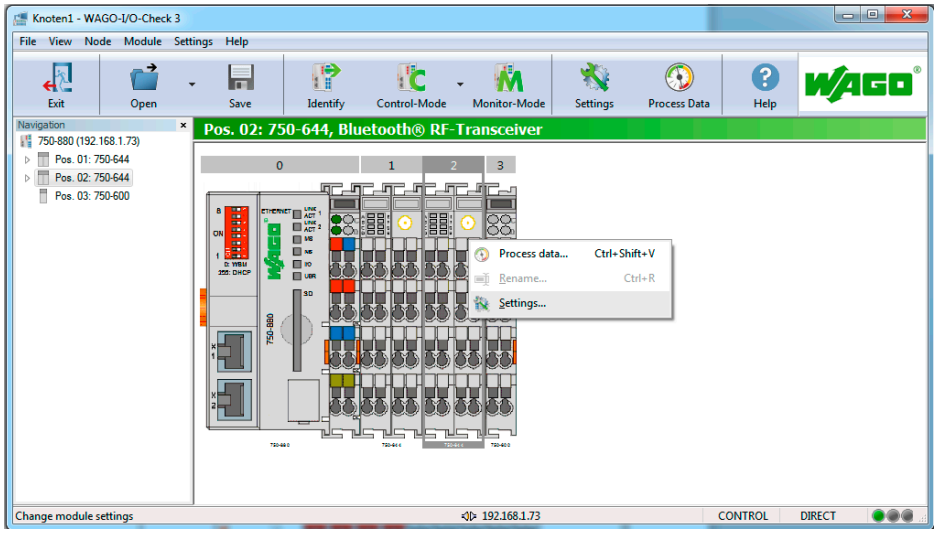


Figure 41: Identifying the node configuration (example)

1. Launch the WAGO-I/O-*CHECK* software (Version 3 or higher).

2. Click the **[Identify]** button on the toolbar of the window. Your node configuration is displayed graphically (see figure “Identifying the Node Configuration”).

3. Right-click on the 750-644 I/O module that you want to configure as the slave.

4. Select **Settings** in the context menu of the 750-644 I/O module to open the *Bluetooth*® specific parameterization dialog of the 750-644 I/O module (see figure “*Bluetooth*® specific parameterization area”).

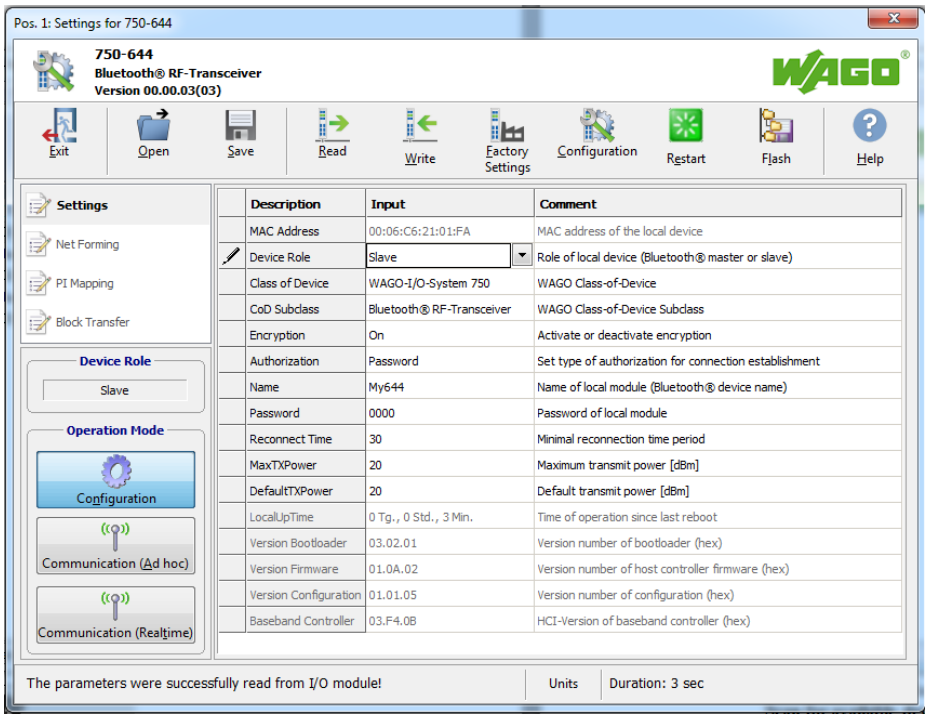


Figure 42: *Bluetooth*® specific parameterization area (example)

Note

Restore to factory settings!
The following steps require that the 750-644 I/O module remain intact, i.e. no configuration has been attempted. If not the case, click the **[Factory Settings]** button in the toolbar to restore the 750-644 I/O module to its factory settings.

5. Click the **[Configuration]** button in the toolbar. A dialog appears:

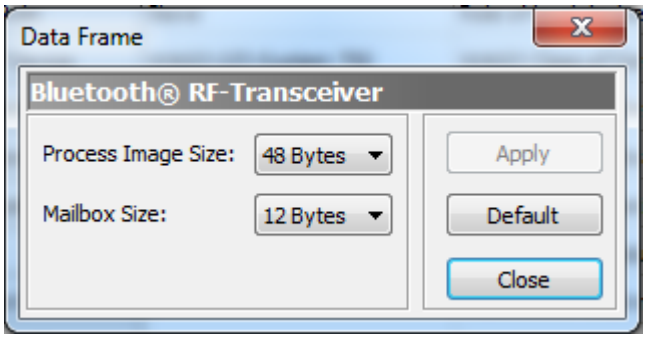


Figure 43: *Data Frame* dialog

6. If necessary, enter a process image size of 48 bytes and a Mailbox size of 12 bytes.

7. Click **[Apply]** to confirm your entries. Then close the "Data Frame" dialog.

8. To refresh the view of the configuration saved in the 750-644 I/O module click the **[Read]** button in the toolbar (see figure "*Bluetooth*® specific parameterization area").

9. Select the **Net Forming** menu item in the navigation on the side:

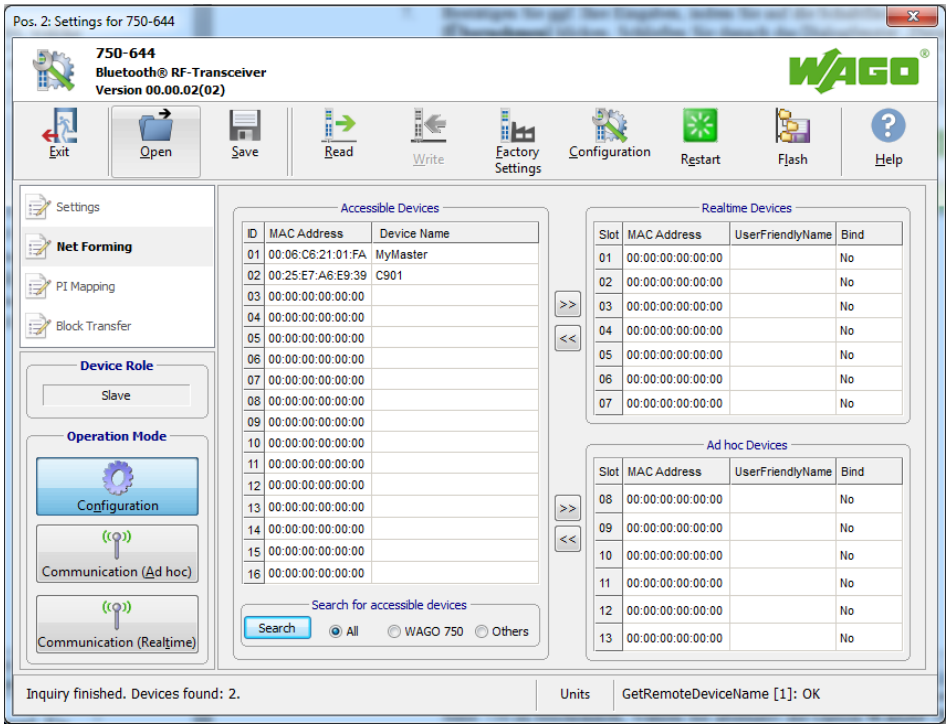


Figure 44: Screenshot of *Net Forming* (example)

10. To search the network for *Bluetooth*® SPP devices in the area, select **All** in the **Search for available devices** area. To limit the search results to WAGO I/O modules of the 750 series, select the **WAGO 750** option.

11. Click the **[Search]** button.

The MAC ID of all *Bluetooth*® SPP devices found are displayed in the list of available devices (see figure "Screenshot of *Net Forming*").

12. Search in the list of MAD IDs of the master to connect this master to your slave.

Note

Factor settings required
The master also has to be in “Configuration” mode (factory setting).

13. Select the MAC ID of the required master.

14. Click the **[>>]** button to transfer the selected MAC ID to the list of real-time devices (or ad-hoc devices) for the current slave. The MAC ID of the master is entered in the first row (slot 1) of the respective table.

15. Give the device a name (UserFriendlyName), e.g. “MyMaster”.

16. Select the MAC ID and select the value “**Yes**” in the **Bind** drop-down menu.

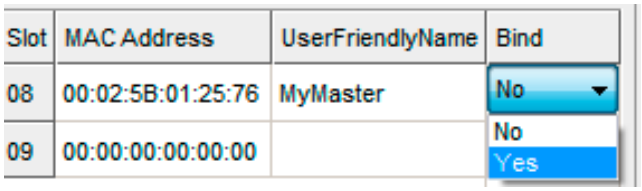


Figure 45: Binding a device (example)

17. Click the **[Write]** button in the toolbar to write the modified configuration to the 750-644 I/O module.

You have assigned a master to the processed slave (slave → master).

18. Click the **[Communication (Realtime)]** button in the **Operating Mode** area to select it.

19. To establish a link from the master side (master → slave), go to the section “Configuring the Master via 'Net Forming'”).

9.1.1.2.2 Configuring the Master via “Net Forming”

1. Launch the WAGO-I/O-*CHECK* software (Version 3 or higher).

2. Click the **[Identify]** button. Your node configuration is displayed graphically (see figure “Identifying the Node Configuration”).

3. Right-click on the 750-644 I/O module that you want to configure as the master.

4. Select **Settings** in the context menu of the 750-644 I/O module to open the *Bluetooth*® specific parameterization dialog of the 750-644 I/O module (see figure “*Bluetooth*® specific parameterization area”).

Note

Restore to factory settings!
The following steps require that the 750-644 I/O module remain intact, i.e. no configuration has been attempted. If not the case, click the **[Factory Settings]** button in the toolbar to restore the 750-644 I/O module to its factory settings.

5. Click the **[Configuration]** button in the toolbar. A dialog appears (see figure "*Data Frame* dialog").

6. If necessary, enter a process image size of 48 bytes and a Mailbox size of 12 bytes.

7. Click **[Apply]** to confirm your entries. Then close the "Data Frame" dialog.

8. To refresh the view of the configuration saved in the 750-644 I/O module click the **[Read]** button in the toolbar (see figure "*Bluetooth*® specific parameterization area").

9. Assign the role of master to the 750-644 I/O module by selecting "**Master**" in the **Device Role** drop-down menu.

10. Select the **Net Forming** menu item in the navigation.

The following section describes how to select the devices required to establish a connection to the master. Devices that are visible for scan requests can first be scanned for in a similar way to the slave configuration (see section "Example Configuration" > … > "Configuring Slaves via 'Net Forming'" steps 10 to 14) and then be stored by drag & drop for example, from the scan results in the list of allowed devices (slots 1 to 13). For security reasons, 750-644 I/O modules are hidden from scan request in "Real-Time Communication" mode, but can be entered like other hidden devices or devices out of reach:

11. Enter the listed MAC address of the slave, which is already in "Real-Time Communication" mode, in the intended field manually.

The following steps assume that you are using slot 1.

12. Give the device a name (UserFriendlyName); e.g. "Slave_01". This makes the overview easier for you.

13. Select the MAC ID and select the value "**Yes**" in the **Bind** drop-down menu (see figure "Binding a device").

14. Click the **[Write]** button in the toolbar to write the modified configuration to the 750-644 I/O module.

The master and slave are now assigned to each other. The master is still in "Configuration" mode.

9.1.1.2.3 Assigning Process Data

Note

Skipping action steps
If the *Bluetooth*® parameterization window is still open, begin with step 3 (see figure "Screenshot of *PI Mapping*").

1. Right-click on the 750-644 I/O module (master).
2. Select **Settings** in the context menu. A new window opens for configuring the 750-644 I/O module concerned.
3. In the navigation, select the **PI Mapping** menu item.

The process data assignment is loaded from the 750-644 I/O module and graphically displayed in WAGO-I/O-*CHECK*.

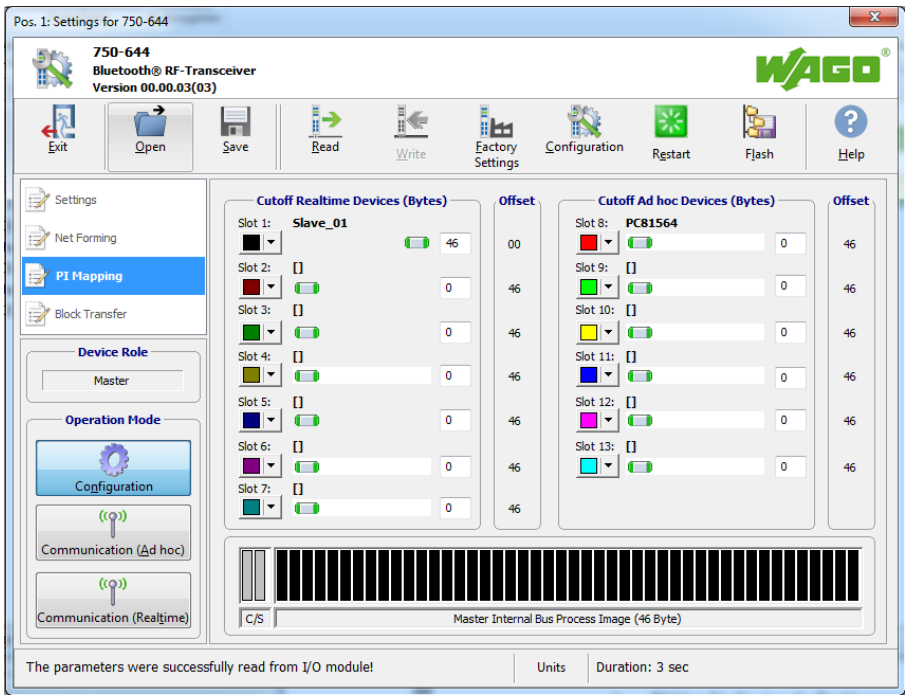


Figure 46: Screenshot *PI Mapping*

4. Move the slider for the first slave to the right so that the first slave is assigned the maximum possible number of bytes in the master's process image.
5. Click the **[Write]** button in the toolbar to write the modified configuration to the 750-644 I/O module.
6. Click the **[Communication (Realtime)]** button in the Operating Mode area to specify the operating mode.

The example configuration is completed.

9.1.1.3 Testing Process Data Exchange

The device must be configured correctly to test the exchange of process data successfully. When LED E of the master flashes a constant green, there is a connection between master and slave.

1. Close the *Bluetooth*® parameterization dialog.
2. Right-click on the master and slave one after the other.
3. Select **Process Data** in the context menu. The process data dialog opens where you can view the raw data.
4. Right-click on the word ***Bluetooth*® RF Transceiver** in the table headers of the dialog.

You can choose between **Input Data**, **Output Data** and **Reset**. From now on, you can switch between the display for input and output data using this menu (see figure “View of the process data”).

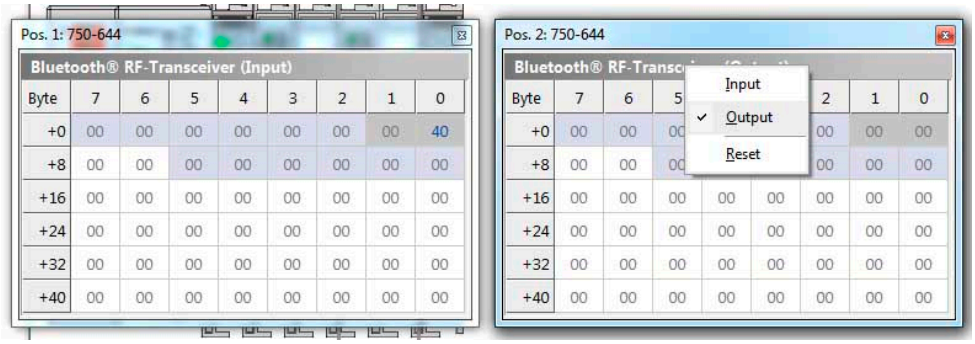


Figure 47: View of the process data

5. Select **Output Data** for one of the 750-644 I/O modules. Enter any test data in the “Output” dialog by double-clicking in the required table field to enable overwrite.
   (In the process image from offset + 2, byte 0 and 1 are reserved for status information.)
6. Select **Input Data** for the other 750-644 I/O module. The “Input” dialog opens.
7. Check if the output data of the one 750-644 I/O module leads to the correct input data in the other 750-644 I/O module.

9.1.2 Startup via Mailbox Commands in the Process Data Dialog

In addition to using the *Bluetooth*® parameterization dialog, it is also possible to use Mailbox commands to configure 750-644 I/O modules. Mailbox commands are entered using function blocks in the WAGO-I/O-*PRO* CAA or in the process data dialog of WAGO-I/O-*CHECK*. WAGO-I/O-*CHECK* is used here.

Information

Configuration program
The WAGO-I/O-*CHECK* configuration program is a convenient tool you can use to enter/execute Mailbox commands as hexadecimal opcodes and to view the result in the input data. You can obtain the software on a CD-ROM with item number 759-302 from WAGO Kontakttechnik GmbH & Co. KG.

Note

Mailbox commands
Mailbox commands are executed when a new opcode is entered and/or when the toggle bit is changed.

9.1.2.1 Network Configuration

In the following example, a master with four slaves is configured. There should be five 750-644 I/O modules in your network.

9.1.2.2 Startup of the 750-644 I/O Modules

Note

Parameterization in "Configuration" mode only!
Make sure that the 750-644 I/O modules are in "Configuration" mode.

1. Click **[Identify]** in WAGO-I/O-*CHECK* to display your node graphically.
2. Right-click on a 750-644 I/O module. Select **Process Data** in the context menu. The process data dialog opens.
3. In the process data dialog, right click on the word ***Bluetooth*® RF Transceiver** in the table header of the dialog.
4. Select **Output Data** in the context menu:

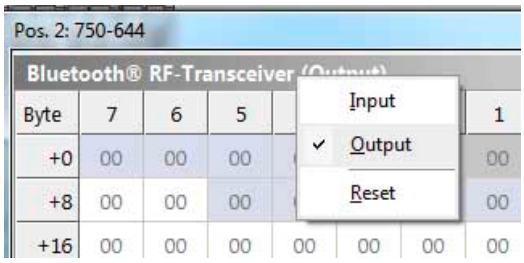


Figure 48: Display *Bluetooth*® output data

9.1.2.2.1 Switching On the Mailbox

1. Switch on the Mailbox for all 750-644 I/O module by setting the control byte to 0x20 (bit $2^5 = 1$) (see table "Switching on the Mailbox").

Depending on the version, various error/warning bits can be set in the status byte. The Mailbox is ON when acknowledged in byte 0 (status byte) with 0x60:

60hex = 0110.0000bin → Bit 2^5 and 2^6 are set.

Bit 2^5 acknowledges that the Mailbox is ON while bit 2^6 displays the still disabled radio connection.

A description of the control and status bytes is available in section "Process Image" > … > "Control/Status Byte (C/S Byte)".

Table 52: Switching ON the Mailbox

Byte	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x00	0x00	0x00	0x00	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x60

9.1.2.2.2 Resetting 750-644 I/O Modules to Factory Settings

1. Use the "SetFactorySettings" Mailbox command (Opcode 0x57) to reset all 750-644 I/O modules to factory settings (see table "Mailbox command *SetFactorySettings*").

2. Wait 5 seconds after executing the command before you continue. This gives the internal *Bluetooth*® subsystem time to reset.

Table 53: Mailbox command *SetFactorySettings*

Byte	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x00	0x00	0x00	0x57	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x57	0x00	0x60

3. Execute the command "FlashRebootHost" (Opcode 0x11) for all 750-644 I/O modules to restart them all (see table "Mailbox command *FlashRebootHost*").

4. Wait 5 seconds after executing the command.

Table 54: Mailbox command *FlashRebootHost*

Byte	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x00	0x00	0x00	0x11	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x11	0x00	0x60

9.1.2.2.3 Specifying the Master

1. Specify one of the 750-644 I/O modules as the master and set byte 4 to 0x01 (MBX_DEVICE_ROLE).

2. Execute the Mailbox command "SetLocalDeviceRole" (Opcode 0x56) for this 750-644 I/O module to assign it the role of master (see table "Mailbox command SetLocalDeviceRole").

3. Wait 5 seconds after executing the command so that the *Bluetooth*® subsystem can adjust.

The remaining 4 750-644 I/O modules are already configured as slaves by default.

Table 55: Mailbox command *SetLocalDeviceRole*

Byte	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x00	0x01	0x00	0x56	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x56	0x00	0x60

9.1.2.2.4 Querying MAC IDs

1. To query the MAC IDs of the master and slaves, use the Mailbox command "GetLocalMacID" (Opcode 0x42) (see table "Mailbox command *GetLocalMacID*"). Execute the command for all 750-644 I/O modules.

Table 56: Mailbox command *GetLocalMacID*

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x42	0x00	0x20
PD/I	0x00	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x00	0x42	0x00	0x60

2. Note the return values for bytes 4 to 9 in the table "Entering the return values":

Table 57: Entering the return values

Byte	9	8	7	6	5	4
Master	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _
Slave 1	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _
Slave 2	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _
Slave 3	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _
Slave 4	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _

The return values (MAC IDs) of the master are loaded in the device lists and the return values of the slaves in the device list of the master.

9.1.2.2.5 Loading the MAC IDs of the Slave in the Device List of the Master

1. Write the MAC IDs of the 1st slave in bytes 5 to 10 of the master (see table "Entering *AllowRemoteDevice*, slave 1").

2. Using 0x20 (TABLE_ENTRY) in byte 4, specify the 1st table entry in which the MAC ID should be written.

3. Use the Mailbox command "AllowRemoteDevice" (Opcode 0x83) to load the MAC ID of the slave in the device list of the master.

Table 58: Entering Mailbox command *AllowRemoteDevice*, slave 1

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x20	0x00	0x83	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x83	0x00	0x60

4. Write the MAC ID of the 2nd slave in bytes 5 to 10 of the master (see table "Entering Mailbox command *AllowRemoteDevice*, slave 2").

5. Using 0x21 (TABLE_ENTRY) in byte 4, specify the 2nd table entry in which the MAC ID should be written.

6. Since opcode 0x83 has not changed, but the Mailbox command should be executed again with the MAC entered under 4, change the toggle bit to 0x80.

Table 59: Entering Mailbox command *AllowRemoteDevice*, slave 2

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x21	0x80	0x83	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x80	0x83	0x00	0x60

7. Write the MAC ID of the 3rd slave in bytes 5 to 10 of the master (see table "Entering Mailbox command *AllowRemoteDevice*, slave 3").

8. Using 0x22 (TABLE_ENTRY) in byte 4, specify the 3rd table entry in which the MAC ID should be written.

9. Change the toggle byte to 0x00 to execute the Mailbox command (opcode 0x83) again.

Table 60: Entering Mailbox command *AllowRemoteDevice*, slave 3

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x22	0x00	0x83	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x83	0x00	0x60

10. Write the MAC ID of the 4th slave in bytes 5 to 10 of the master (see table "Entering Mailbox command *AllowRemoteDevice*, slave 4").

11. Using 0x23 (TABLE_ENTRY) in byte 4, specify the 4th table entry in which the MAC ID should be written.

12. Change the toggle byte to 0x80 to execute the Mailbox command (opcode 0x83) again.

Table 61: Entering Mailbox command *AllowRemoteDevice*, slave 4

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x23	0x80	0x83	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x80	0x83	0x00	0x60

9.1.2.2.6 Loading the MAC ID of the Master in the Device Lists of the Slaves

1. Write the MAC IDs of the master in bytes 5 to 10 of the 1st slave (see table "Entering *AllowRemoteDevice*, master").
2. Set byte 4 to 0x20 (TABLE_ENTRY).
3. Use the command "AllowRemoteDevice" (Opcode 0x83) to load the master in the device list of the slave.
4. Follow the same steps for the device lists of the other slaves.

Table 62: Entering Mailbox command *AllowRemoteDevice*, master

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x20	0x80	0x83	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x80	0x83	0x00	0x60

9.1.2.2.7 Binding Slaves

1. Write the MAC IDs of the 1st slave in bytes 5 to 10 of the master (see table "Mailbox command *BindRemoteDevice* for binding slave 1").
2. Set byte 4 to 0x20 (TABLE_ENTRY).
3. Use the Mailbox command "BindRemoteDevice" (Opcode 0x85) to load the MAC ID of this slave in the device list of the master.

Table 63: Mailbox command *BindRemoteDevic*e for binding slave 1

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x20	0x00	0x85	0x00	0x20
PD/I	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x00	0x00	0x85	0x00	0x60

4. Write the MAC IDs of the 2nd slave in bytes 5 to 10 of the master (see table "Mailbox command *BindRemoteDevice* for binding slave 2").
5. Set byte 4 to 0x21 (TABLE_ENTRY).
6. Change the toggle byte to 0x80 to execute the Mailbox command (opcode 0x85) again.

Table 64: Mailbox command *BindRemoteDevice* for binding slave 2

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x21	0x80	0x85	0x00	0x20
PD/I	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x00	0x80	0x85	0x00	0x60

7. Write the MAC IDs of the 3rd slave in bytes 5 to 10 of the master (see table "Mailbox command *BindRemoteDevice* for binding slave 3").

8. Set byte 4 to 0x22 (TABLE_ENTRY).

9. Change the toggle byte to 0x00 to execute the Mailbox command (opcode 0x85) again.

Table 65: Mailbox command *BindRemoteDevice* for binding slave 3

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x22	0x00	0x85	0x00	0x20
PD/I	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x00	0x00	0x85	0x00	0x60

10. Write the MAC IDs of the 4th slave in bytes 5 to 10 of the master (see table "Mailbox command *BindRemoteDevice* for binding slave 4").

11. Set byte 4 to 0x23 (TABLE_ENTRY).

12. Change the toggle byte to 0x80 to execute the mailbox command (opcode 0x85) again.

Table 66: Mailbox command *BindRemoteDevice* for binding slave 4

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x23	0x80	0x85	0x00	0x20
PD/I	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x00	0x80	0x85	0x00	0x60

9.1.2.2.8 Binding the Master

1. Use the Mailbox command "BindRemoteDevice" (Opcode 0x85) to bind the master (see table "Mailbox command *BindRemoteDevice* for binding the master").

2. Execute this command with each slave.

Table 67: Mailbox command *BindRemoteDevice* for binding the master

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x20	0x80	0x85	0x00	0x20
PD/I	0x00	0x06	0xC6	0x_ _	0x_ _	0x_ _	0x00	0x80	0x85	0x00	0x60

9.1.2.2.9 Setting “Communication” Mode for Master and Slaves

1. Set all slaves in “Communication” mode using Mailbox command “SetLocalOperationMode” (Opcode 0x4A)
   (Opcode 0x03 for the ad-hoc profile)
   (see table “Mailbox command *SetLocalOperationMode*”).

2. Follow the same steps for the master.

3. Wait 5 seconds after executing the command.

Table 68: Mailbox command *SetLocalOperationMode*

Byte	10	9	8	7	6	5	4	Toggle	Opcode	blank	C/S
PD/O	0x00	0x00	0x00	0x00	0x00	0x03	0x02	0x80	0x4A	0x00	0x20
PD/I	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x80	0x4A	0x00	0x60

Test exchange of process data as described in section “Example Configuration” > … > “Testing Process Data Exchange“.

9.2 Connecting with WAGO 757-801

Proceed as follows to bind your 750-644 I/O module with a WAGO *Bluetooth*® module 757-801:

9.2.1 Preparing a 757-801

Note

Configuration of the *Bluetooth*® module 757-801
The process for configuring the *Bluetooth*® module 757-801 using AT commands is described in the manual for the *Bluetooth*® module 757-801!

1. Reset the *Bluetooth*® module 757-801 to its factory settings.
2. Enter the MAC ID of your 750-644 I/O module in the list of allowed devices (AT BTSEC ADEV ADD = <MAC ID of the 750-644 I/O module>).
3. Enter the MAC ID of your 750-644 I/O module in the list of trusted devices (AT CON TAR ADR = <MAC ID 750-644 I/O module>).
4. Enforce secure authentication (AT BTSEC AUTH = 1).
5. Exit configuration mode (AT LOGOUT) to start data transmission.

9.2.2 Preparing a 750-644

1. Reset the 750-644 I/O module to its factory settings.
2. Enter the *Bluetooth*® module 757-801 password (default: 1234) for the 750-644 I/O module.
3. Bind the *Bluetooth*® module 757-801 by entering its MAC ID in the list of ad-hoc devices and activating the respective slot (“Bind”).
4. Put the 750-644 I/O module in “Ad-hoc Communication” mode.

9.2.3 Exchanging Data

1. Use a terminal program (e.g. HyperTerminal, HTerm or Realterm) to send a character string from the *Bluetooth*® module 757-801 to the 750-644 I/O module, e.g. the character “a”.
2. Go to the output data of the process image of the 750-644 I/O module and set bit 2 to the value “0x04”.
   The value “0x16” must be visible in the process image of the inputs on the first place of the slot (byte 2). Signaling “0x02 – RR – Receive request”, “0x04 – CS – Connection established” and “0x10 – LEN=1 – 1 byte data received” are encoded in this value. The following byte should now contain the ordinal numbers of the transmitted string (“0x61”).

3. Invert bit RA (Receive acknowledgement) in the control byte of the ad-hoc slot for the 750-644 I/O module to signal successful acquisition of the data. In this example, you should write byte 2 in the output process image with the value “0x02”.

You can now exchange more data between the *Bluetooth*® module 757-801 and the 750-644 I/O module.

For further information on data exchange, see section “Process Image” > … > “‘Ad-hoc Communication’ Mode”.

9.3 Connecting to an Android Smartphone

Note

We cannot guarantee this functionality

The instructions steps below have been tested representatively for the following devices:

- HTC Sensation (Android 4.0.3)
- Sony XPERIA U (Android 4.0.4)
- Samsung Galaxy S3 (Android 4.1.1)
- Samsung Galaxy Note 1 (Android 4.1.2)

WAGO cannot, however, provide any guarantee that a link can be set up with your smartphone, on account of the large variety and rapid development of standard versions and configurations.

Proceed as follows to link your Android smartphone to the 750-644 I/O module:

1. Reset the 750-644 I/O module to the factory settings.
2. Activate the *Bluetooth*® function of your smartphone.

If you already know the *Bluetooth*® MAC ID of your smartphone, jump to step 6.

3. In the *Bluetooth*® settings for your smartphone, set its discoverability to “Discoverable for all devices”.
4. Using the 750-644 I/O module, scan for other available *Bluetooth*® devices. Use an unfiltered search, i.e. a search where you specify 0x00 as the CoD filter.
5. Identify your smartphone in the list of devices found. If your smartphone is not in the list, repeat the search as necessary.
6. Enter the *Bluetooth*® MAC ID of your smartphone as an allowed ad-hoc device. For example, use slot 8, i.e. slot address 0x10. Activate this slot for the connection (“Bind”).
7. Set the cut-off for slot 1 … 7 to 0 and set the cut-off of the slot specified for the slot to a number between 2 and 16, e.g. 9.
8. Put the 750-644 I/O module in “Ad-hoc Communication” mode.
9. Launch an app on your smartphone that supports *Bluetooth*® SPP communication.
10. Use the app to establish a connection to the 750-644 I/O module. If prompted to enter a PIN, use the PIN configured on the 750-644 I/O module (default “0”).

11. Use the app to send a character string from the smartphone to the 750-644 I/O module, e.g. the character “a”.

The value “0x16” must be visible in the process image of the inputs at the first position of the slot (byte 2). Signaling “0x02 – RR – Receive request”, “0x04 – CS – Connection established” and “0x10 – LEN=1 – 1 byte data received” are encoded in this value. The following byte should now contain the ordinal numbers of the transmitted string (“0x61”).

12. Invert bit RA (Receive acknowledgement) in the control byte of the ad-hoc slot for the 750-644 I/O module to signal successful acquisition of the data. In this example, you should write byte 2 in the output process image with the value “0x02”.

You can now exchange other data between the smartphone and 750-644 I/O module.

9.4 Connecting to Windows 7 PC

Proceed as follows to establish a connection between a 750-644 I/O module and your PC:

9.4.1 Preparing a PC

To link a 750-644 I/O module and PC, you have to know the device name or MAC ID of your PC. The MAC ID is available in the settings of your *Bluetooth*® adapter:

1. Right-click on the *Bluetooth*® icon in the system tray.
2. Select **Open Settings** in the context menu.
3. Go to the **Hardware** tab.
4. Press the **[Properties]** button.
5. Go to the **Advanced** tab. The radio information is displayed. The requested MAC ID is the address.

9.4.2 Preparing a 750-644 I/O Module

1. Configure the 750-644 I/O module in WAGO-I/O-*CHECK* 3 (see also section “Commissioning” > … > “Configuration and Parameterization with WAGO-I/O-*CHECK*”). Make sure that the **[Configuration]** button is selected in the **Operation Mode** area.
2. Enter the following parameters on the **Settings** page:

Table 69: Configuring the 750-644 I/O module to connect to a PC

Description	Input
Device role	Slave
Encryption	On
Authorization	Password
Name	My644
Password	0000

3. Go to the **Net Forming** page.
4. Scan for available devices: Choose the **All** radio button and press the **[Search]** button.
5. In the list of **Accessible Devices**, select the MAC ID of your PC and add it to the list of **Ad hoc Devices** (slot **08**).
6. Bind your PC by clicking the entry in the **Bind** column and selecting “**Yes**” in the respective selection list.
7. Go to the **PI Mapping** page.

8. **Cutoff Ad hoc Devices (Bytes)** list: Set the cut-off for slot 8 to the value “10”.

9. Transfer the settings concerned to the 750-644 I/O module by pressing the **[Write]** button in the toolbar.

10. Change the mode by clicking the **[Communication (Ad hoc)]** button in the **Operation Mode** area.

9.4.3 Adding the 750-644 I/O Module to the PC Device List

1. Right-click on the *Bluetooth*® icon in the system tray.

2. Select **Show *Bluetooth* Devices** in the context menu.

3. Press the **[Add a Device]** button. A list of found devices is displayed.

4. Double-click on the 750-644 I/O module in the list. It lists the name that you assigned during the configuration, i.e. “My644” in this example:

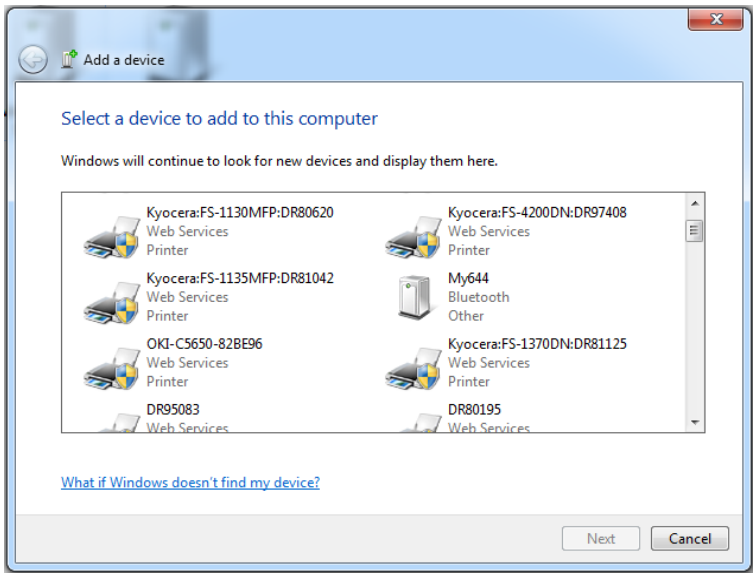


Figure 49: Add I/O module to the PC

2. Enter the pairing code you assigned to the 750-644 I/O module during the configuration, i.e. “0000” in this example.

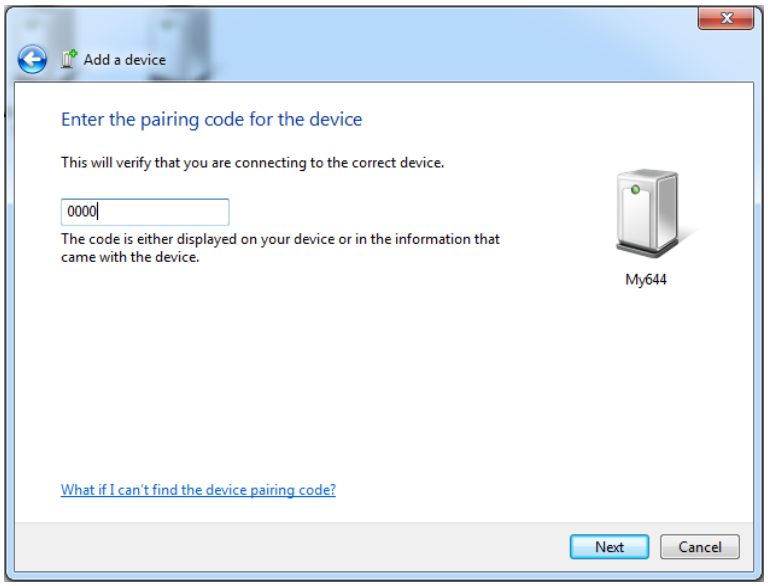


Figure 50: Enter the pairing code

3. Click the **[Next]** button. The I/O module is added to the computer.

9.4.4 Starting Data Exchange via Terminal Program

1. Right-click on the 750-644 I/O module (“My644”) and select **Properties** in the context menu.

2. Go to the **Hardware** tab.

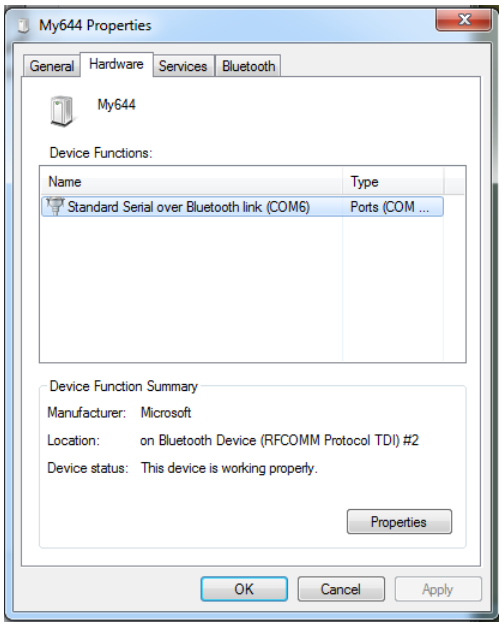


Figure 51: Get COM port

In **Device Functions**, the assigned COM port of the 750-644 I/O module is displayed. Note the specified COM port.

3. Click the **[Properties]** button.

4. Go to the **Port Settings** tab and enter the following values:

Table 70: COM port settings

Description	Value
Bits per second	115200
Data bits	8
Parity	None
Stop bits	1
Flow control	None

5. Launch the terminal program, e.g. Hyperterminal, HTerm or Realterm.

6. Select the COM Port that you noted.

7. Set the baud (“Bits per second”), data (“Data bits”), stop (“Stop bits”) and parity (“Parity”) (see table “COM port settings”).

8. Establish the connection. You can now exchange data between the terminal program on your PC and the 750-644 I/O module.

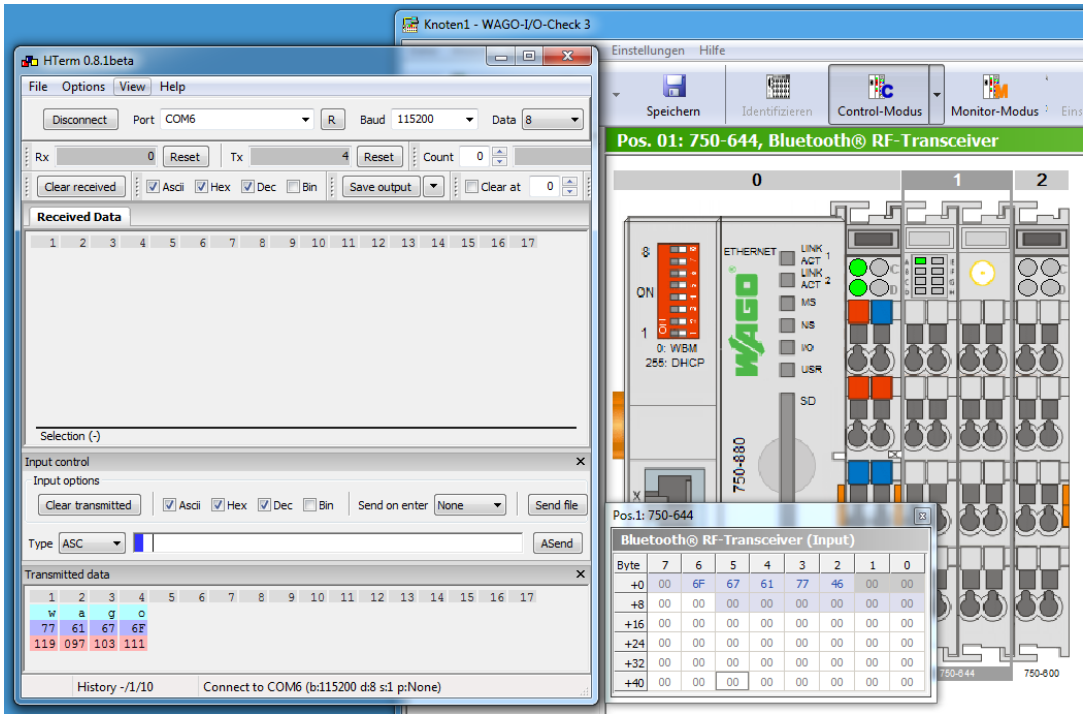


Figure 52: Example of data exchange via HTerm and WAGO-I/O-*CHECK* 3

10 Diagnostics

The 750-644 I/O module provides diagnostic information about various interfaces:

- Display elements (see section “Device Description” > … > “Display Elements”)
- Process image (see section “Process Image”)

The WAGO-I/O-*CHECK* software can also be used to query and display graphically diagnostics from the I/O module.

10.1 Diagnostics in WAGO-I/O-*CHECK*

WAGO-I/O-*CHECK* provides a visualization of the diagnostic information available from the I/O module. WAGO-I/O-*CHECK* automatically queries the information displayed via the process image. How to use the diagnostic view in WAGO-I/O-*CHECK* is described below.

1. Choose **Settings** in the context menu of the selected 750-644 I/O module to open the parameterization dialog.

2. Click the respective button to switch to “Ad-hoc Communication” or “Real-Time Communication” mode if necessary.

3. Click **Diagnostics** in the navigation. The **Channel monitor** table and the following status fields are displayed on this page:

   - Device Role
   - Operation Mode
   - Communication Profile
   - Diagnostic State
   - Network State

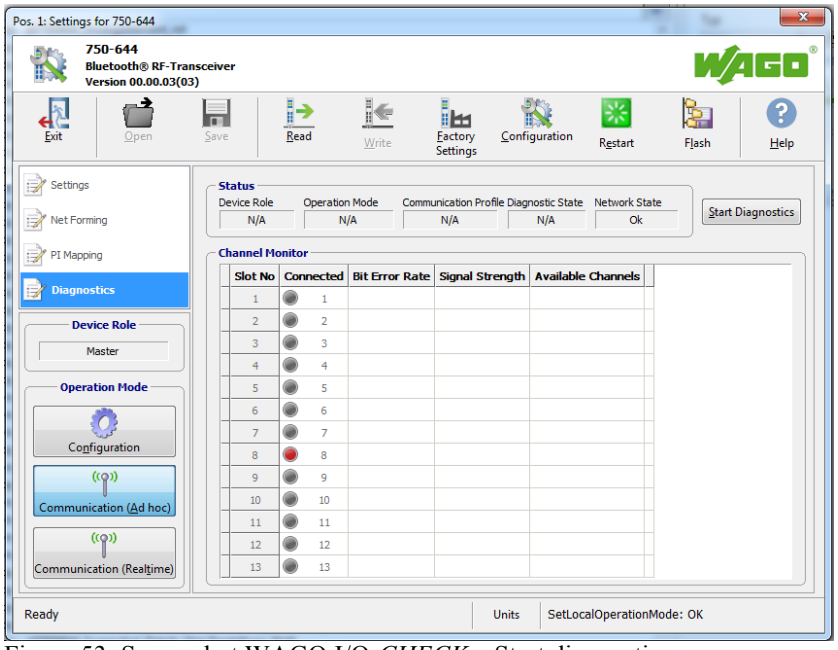


Figure 53: Screenshot WAGO I/O-*CHECK* – Start diagnostics

You can start various queries:

- To query the current values continuously:
  1. Click the **[Start Diagnostics]** button.
     The values are continuously queried and displayed in the Channel monitor table.
  2. Click the **[Stop Diagnostics]** button to stop the query and to display the status last received without updating again:

- To query status information for individual slots or all existing connections:
  1. Right-click on the last column of the table. A dialog appears:

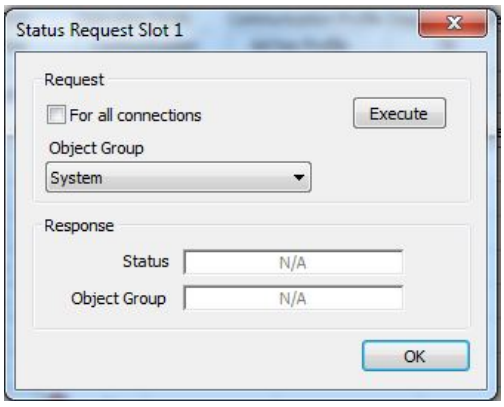


Figure 54: *Status query for slot 1*

  2. Select an **Object Group**.
  3. Click the **[Execute]** button.
  4. Click **[OK]** to close the dialog.

Information

Additional information
An example configuration with WAGO-I/O-*CHECK* and the meaning of the individual values in the status information are available in the section “Example Configuration” > … > “Diagnostics”.

10.2 Diagnostics via Process Image or LED Display

The most important diagnostic information and statuses available via process image or LED display are explained below.

- No field supply
- No radio connection
- Signal quality of the radio connections
- Validity of the process data
- Topicality of the process data
- Configuration errors

10.2.1 No Field Supply

If there is no field supply, LED A lights up red and all other LEDs are off. The status byte (byte 0) in the process image of the inputs then have the value 0x40. The 750-644 I/O module processes register communication or parameter channel data, but does not respond to the Mailbox protocol.

The status byte in the process image of the inputs can also take the value 0x40 under other conditions. However, you can easily avoid misinterpretation:

- Initialization after restart not yet completed:
  Wait several seconds (see section “Function Description” > … > “Startup Behavior”). If the LED indicators or the value of the status byte does not change during this time:
  Check if one of the following cases applies.

- “Configuration” mode when the Mailbox is hidden or “Real-Time Communication” mode when the Mailbox is hidden and at least one interrupted radio connection:
  Show the Mailbox (see section “Process Image” > … > “Control/Status Byte” and “Mailbox”). If the I/O module confirms displaying the Mailbox, there is sufficient field supply available.

Fehlt die Feldversorgung, leuchtet LED A rot, alle übrigen LEDs sind aus. Das Statusbyte (Byte 0) im Prozessabbild der Eingänge weist dann den Wert 0x40 auf. Die Busklemme 750-644 verarbeitet zwar Daten der Registerkommunikation oder des Parameterkanals, reagiert jedoch nicht auf das Mailbox-Protokoll.

10.2.2 No Radio Connection

Radio connections are absent when the following applies:

- The I/O module is in “Real-Time Communication” or “Ad-hoc Communication” mode.
- At least one of the slots relevant in the respective mode has been configured correctly and enabled for the connection, but there is no connection.

Depending on the operating mode and role configured, the 750-644 I/O module indicates no radio connection in different ways:

- “Real-Time Communication” mode, operating as the master:
  - The LED assigned to the respective slot is red or yellow.
  - Bit 6 in the status byte (byte 0) has the value 1.
  - In the response to Mailbox command “GetNetworkStatus (0xD1)”, the I/O module reports not radio connection.
- “Real-Time Communication” mode, operating as a slave:
  - LED E is red or yellow.
  - Bit 6 in the status byte (byte 0) has the value 1.
  - In the response to Mailbox command “GetNetworkStatus (0xD1)”, the I/O module reports not radio connection.
- “Ad-hoc Communication” mode
  - The LED assigned to the respective slot is red or yellow.
  - Bit 2 in the status byte of the respective ad-hoc slot (e.g. byte 2 for slot address 0x10) has the value 0.
  - In the response to Mailbox command “GetNetworkStatus (0xD1)”, the I/O module reports not radio connection.

To prevent misinterpretations, check the operating mode and configuration of the 750-644 I/O module in case of doubt using the following Mailbox commands (overview for connecting enabled slots, current operating mode):

- GetLocalDeviceStatus (0xD0)
- GetAllowedRemoteDevices (0x84)
- GetBoundRemoteDevices (0x87)

10.2.3 Signal Status of the Radio Connections

For all existing radio connections, the 750-644 I/O module provides the following status information:

- Signal quality based on the bit error rate (**BER**):
  The bit error rate can be queried using the Mailbox command “GetLinkQuality, 0xD5”. Because an error-correcting procedure is used for radio transmission, transmission errors do not affect transmitted user data directly. However, a high bit error rate indicates significant interference. You can expect automatic error correction to negatively affect timing.

- Signal strength indicator (**RSSI**):
  The signal strength indicator is a measurement of received signal strength.

The Mailbox command “GetLinkSignalStrength, 0xD7” can be used to query the signal strength. If the radio connection has to bridge a long distance, the RSSI is negative. Because the 750-644 I/O module has a very sensitive receiver, a stable radio connection itself is still possible with an RSSI value of -15 when the ambient conditions are invariably stable.

- Channel mask (**AFH**):
  With *Bluetooth*® technology, interrupted frequency ranges are automatically detected and own transmissions avoided. The channel mask provides information about which of the 79 available frequency ranges is currently in use. The channel mask can be read using the Mailbox command “GetAvalailableChannelMap, 0xD8”. The best transmission conditions exist when all frequencies are available. By contrast, if only a few frequencies are available, there are probably other radio systems with high transmission power in the vicinity.

Depending on the operating mode and role configured, the 750-644 I/O module indicates the signal quality in different ways:

- “Real-Time Communication” mode, operating as the master:
  - The specified Mailbox commands can be used to query status information about the radio connections to the slaves.
- “Real-Time Communication” mode, operating as a slave:
  - LEDs B and F visualize the signal strength indicator.
  - LEDs C and G visualize the signal quality.
  - LEDs D and H visualize the channel mask.
  - The above Mailbox commands can be used to query status information about the radio connection to the master.
- “Ad-hoc Communication” mode
  - The specified Mailbox commands can be used to query status information about each radio connection.

10.2.4 Validity of Process Data

Parts of the process image can be overlaid by the register communication, parameter channel or Mailbox configuration protocols (see section “Process Image” > … > “Overlaid Configuration Protocols”).

If overlaid in “Real-Time Communication” or “Ad-hoc Communication” modes, data exchange with remote devices may be temporarily interrupted for one or more slots. The relevant parts of the process image are then invalid in regards to data transmission. The 750-644 I/O module thus indicates possible invalidity of the process data in the input process image.

If bit 7 is set on the local 750-644 I/O module in the status byte (byte 0), register communication is enabled. Is such case, neither the input nor output process image contain valid process data, i.e. in this state, data cannot be sent to nor received from remote *Bluetooth*® devices.

If bit 5 on the local 750-644 I/O module is set in the status byte (byte 0), the Mailbox is shown. In this case, the data of one or more slots is invalid depending on the configuration of the cut-off values.

10.2.5 Topicality of Process Data

In “Real-Time Communication” mode, the 750-644 I/O module monitors the timing of cyclic data exchange with remote 750-644 I/O modules.

In normal operation, data is exchanged at relatively short cycle time. However, circumstances can arise that delay data exchange. In this regard, two basic cases can be distinguished:

- Data is not exchanged with one or more slots for known reasons.
- Data exchange with one or more slots is delayed for other or unknown reasons.

10.2.5.1 Topicality Not Specified for Known Reasons

Such is the case when one or more configured radio connections are not present (see section “Diagnostics” > … > “No Radio Connection”). Such can also be the case if the Mailbox is shown on a connected remote 750-644 I/O module (see section “Process Image” > … > “Overlaid Configuration Protocols”). The local 750-644 I/O module indicates a Mailbox shown on the other side with the value “1” in bit 2 of the status byte (byte 0).

10.2.5.2 Topicality Not Specified for Other Reasons

This can be caused by massive interruption of the radio connection or immediately upon failure of a remote I/O module. The 750-644 I/O module monitors the time interval for receiving any new process data from connected 750-644 I/O modules. If there is no known reason for any delay in exchanging data, the I/O module first indicates a warning and then an error (see also section “Process Image” > … > “Process Data Overlaid by Mailbox”).

The 750-644 I/O module indicates **Warning limit exceeded** with the value “1” in bit 3 of the status byte (byte 0).

The 750-644 I/O module indicates **Error limit exceeded** with the value “1” in bit 1 of the status byte (byte 0).

Worsening of the timing can indicate an imminent loss of connection. If a connection is lost completely, it can negatively affect other existing radio connections (see section “Function Description” > … > “Operation as (...) in 'Real-Time Communication' Mode”). Therefore, available diagnostic options should be exhausted as long as the connection continues (see section “Diagnostics” > … > “Signal Status of the Radio Connections”).

10.2.6 Configuration Errors

Improper use of the configuration protocols can cause the 750-644 I/O module to behave unexpectedly.

There is a known configuration error when LED lights up or flashes yellow. In this case, normal operations can be restored using the following sequence of steps:

1. Set the values of bytes 0, 1, 2 and 3 of the output process image to 0x00, 0x00, 0x00, 0x00.

2. Wait approx. 1 second.

3. Set the values of bytes 0, 1, 2 and 3 of the output process image to 0x20, 0x00, 0x15, 0x80.

4. Wait approx. 6 seconds.

Other configuration errors can be express in different ways. Normal operations can usually be restored using the following sequence of steps:

1. Reset the system parameters back to factory settings by setting the value of parameter 255 to 0x02 (see section “Process Image” > … > “Parameter Channel”).

2. Reboot the fieldbus node, e.g. by briefly interrupting the power supply.

3. Reset all other parameters to factory settings by executing the Mailbox command “SetFactorySettings, 0x57”.

11 Use in Hazardous Environments

The **WAGO-I/O-SYSTEM 750** (electrical equipment) is designed for use in Zone 2 hazardous areas.

The following sections include both the general identification of components (devices) and the installation regulations to be observed. The individual subsections of the “Installation Regulations” section must be taken into account if the I/O module has the required approval or is subject to the range of application of the ATEX directive.

11.1 Marking Configuration Examples

11.1.1 Marking for Europe According to CENELEC and IEC

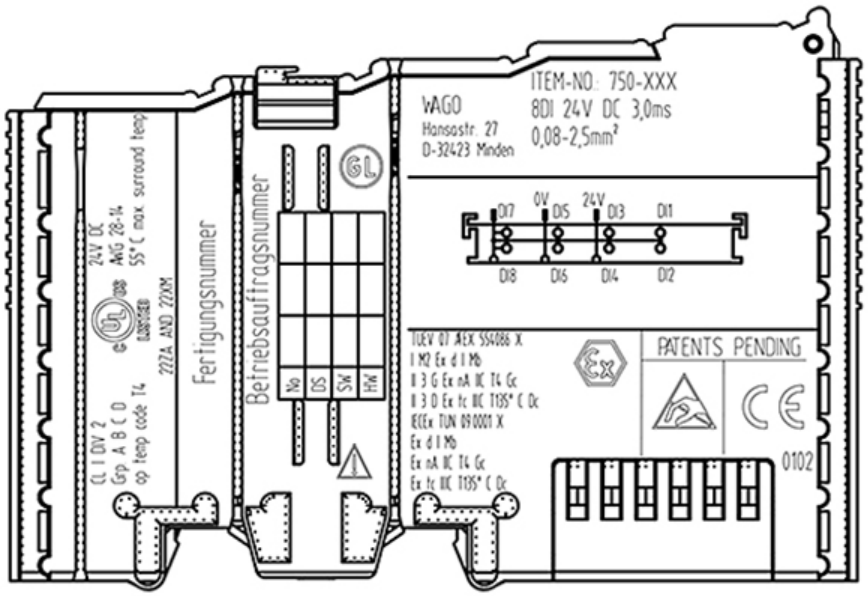


Figure 55: Side marking example for approved I/O modules according to ATEX and IECEx

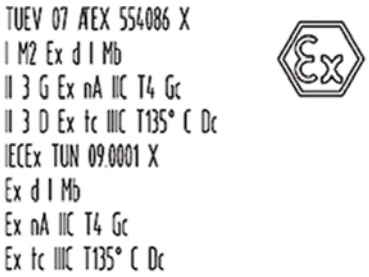


Figure 56: Printing Text detail – Marking example for approved I/O modules according to ATEX and IECEx.

Table 71: Description of marking example for approved I/O modules according to ATEX and IECEx

Printing on Text	Description
TÜV 07 ATEX 554086 X IECEx TUN 09.0001 X	Approving authority and certificate numbers
Dust	
II	Equipment group: All except mining
3D	Category 3 (Zone 22)
Ex	Explosion protection mark
tc Dc	Type of protection and equipment protection level (EPL):protection by enclosure
IIIC	Explosion group of dust
T 135°C	Max. surface temperature of the enclosure (without a dust layer)
Mining	
I	Equipment group: Mining
M2	Category: High level of protection
Ex	Explosion protection mark
d Mb	Type of protection and equipment protection level (EPL): Flameproof enclosure
I	Explosion group for electrical equipment for mines susceptible to firedamp
Gases	
II	Equipment group: All except mining
3G	Category 3 (Zone 2)
Ex	Explosion protection mark
nA Gc	Type of protection and equipment protection level (EPL): Non-sparking equipment
nC Gc	Type of protection and equipment protection level (EPL): Sparking apparatus with protected contacts. A device which is so constructed that the external atmosphere cannot gain access to the interior
IIC	Explosion group of gas and vapours
T4	Temperature class: Max. surface temperature 135°C

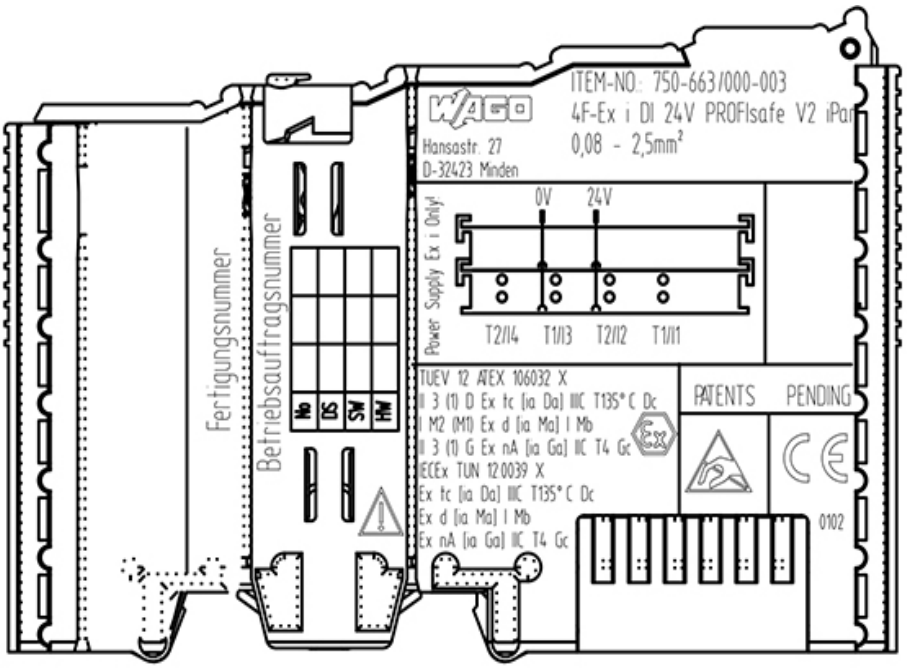


Figure 57: Side marking example for approved Ex i I/O modules according to ATEX and IECEx.

TUEV 12 ATEX 106032 X
II 3 (1) D Ex tc [ia Da] IIIC T135°C Dc
I M2 (M1) Ex d [ia Ma] I Mb
II 3 (1) G Ex nA [ia Ga] IIC T4 Gc
IECEx TUN 12.0039 X
Ex tc [ia Da] IIIC T135°C Dc
Ex d [ia Ma] I Mb
Ex nA [ia Ga] IIC T4 Gc

Figure 58: Text detail – Marking example for approved Ex i I/O modules according to ATEX and IECEx.

Table 72: Description of marking example for approved Ex i I/O modules according to ATEX and IECEx

Inscription text	Description
TÜV 07 ATEX 554086 X IECEx TUN 09.0001X TÜV 12 ATEX 106032 X IECEx TUN 12.0039 X	Approving authority and certificate numbers
Dust	
II	Equipment group: All except mining
3(1)D	Category 3 (Zone 22) equipment containing a safety device for a category 1 (Zone 20) equipment
3(2)D	Category 3 (Zone 22) equipment containing a safety device for a category 2 (Zone 21) equipment
Ex	Explosion protection mark
tc Dc	Type of protection and equipment protection level (EPL): protection by enclosure
[ia Da]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 20
[ib Db]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 21
IIIC	Explosion group of dust
T 135°C	Max. surface temperature of the enclosure (without a dust layer)
Mining	
I	Equipment Group: Mining
M2 (M1)	Category: High level of protection with electrical circuits which present a very high level of protection
Ex d Mb	Explosion protection mark with Type of protection and equipment protection level (EPL): Flameproof enclosure
[ia Ma]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety electrical circuits
I	Explosion group for electrical equipment for mines susceptible to firedamp

Table 72: Description of marking example for approved Ex i I/O modules according to ATEX and IECEx

Gases	
II	Equipment group: All except mining
3(1)G	Category 3 (Zone 2) equipment containing a safety device for a category 1 (Zone 0) equipment
3(2)G	Category 3 (Zone 2) equipment containing a safety device for a category 2 (Zone 1) equipment
Ex	Explosion protection mark
nA Gc	Type of protection and equipment protection level (EPL): Non-sparking equipment
[ia Ga]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 0
[ia Gb]	Type of protection and equipment protection level (EPL): associated apparatus with intrinsic safety circuits for use in Zone 1
IIC	Explosion group of gas and vapours
T4	Temperature class: Max. surface temperature 135°C

11.1.2 Marking for America according to NEC 500

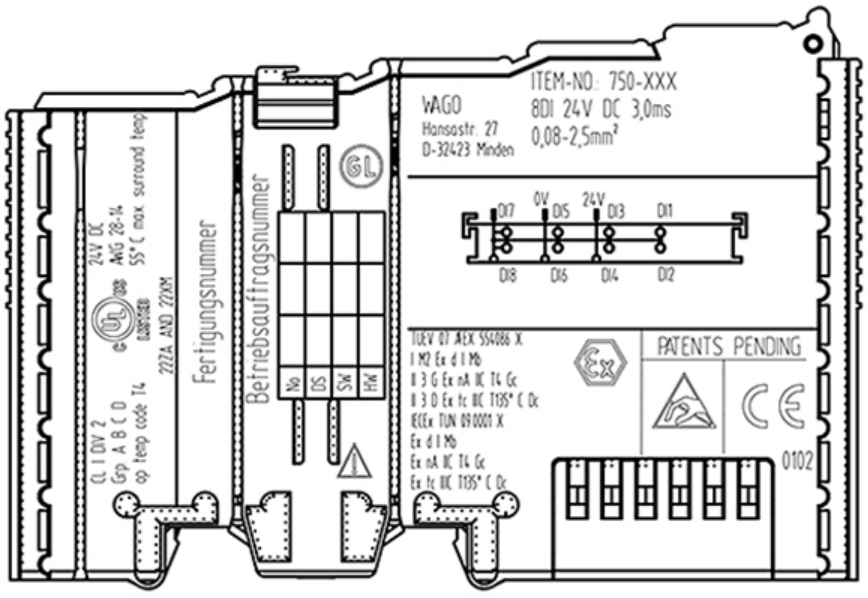


Figure 59: Side marking example for I/O modules according to NEC 500

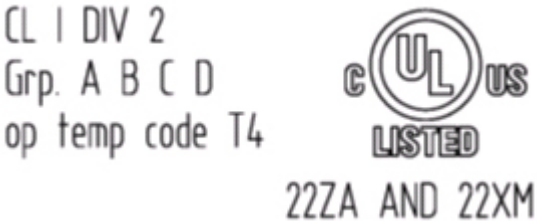


Figure 60: Text detail – Marking example for approved I/O modules according to NEC 500

Table 73: Description of marking example for approved I/O modules according to NEC 500

Printing on Text	Description
CL I	Explosion protection group (condition of use category)
DIV 2	Area of application
Grp. ABCD	Explosion group (gas group)
Op temp code T4	Temperature class

11.2 Installation Regulations

In the **Federal Republic of Germany**, various national regulations for the installation in explosive areas must be taken into consideration. The basis for this forms the working reliability regulation, which is the national conversion of the European guideline 99/92/E6. They are complemented by the installation regulation EN 60079-14. The following are excerpts from additional VDE regulations:

Table 74: VDE installation regulations in Germany

DIN VDE 0100	Installation in power plants with rated voltages up to 1000 V
DIN VDE 0101	Installation in power plants with rated voltages above 1 kV
DIN VDE 0800	Installation and operation in telecommunication plants including information processing equipment
DIN VDE 0185	lightning protection systems

The **USA** and **Canada** have their own regulations. The following are excerpts from these regulations:

Table 75: Installation regulations in USA and Canada

NFPA 70	National Electrical Code Art. 500 Hazardous Locations
ANSI/ISA-RP 12.6-1987	Recommended Practice
C22.1	Canadian Electrical Code

NOTICE

Notice the following points
When using the **WAGO-I/O SYSTEM 750** (electrical operation) with Ex approval, the following points are mandatory:

11.2.1 Special Conditions for Safe Operation of the ATEX and IEC Ex (acc. DEMKO 08 ATEX 142851X and IECEx PTB 07.0064)

The fieldbus-independent I/O modules of the WAGO-I/O-SYSTEM 750-.../...-... must be installed in an environment with degree of pollution 2 or better. In the final application, the I/O modules must be mounted in an enclosure with IP 54 degree of protection at a minimum with the following exceptions:

- I/O modules 750-440, 750-609 and 750-611 must be installed in an IP 64 minimum enclosure.
- I/O module 750-540 must be installed in an IP 64 minimum enclosure for 230 V AC applications.
- I/O module 750-440 may be used up to max. 120 V AC.

When used in the presence of combustible dust, all devices and the enclosure shall be fully tested and assessed in compliance with the requirements of IEC 61241-0:2004 and IEC 61241-1:2004.

When used in mining applications the equipment shall be installed in a suitable enclosure according to EN 60079-0:2006 and EN 60079-1:2007.

I/O modules fieldbus plugs or fuses may only be installed, added, removed or replaced when the system and field supply is switched off or the area exhibits no explosive atmosphere.

DIP switches, coding switches and potentiometers that are connected to the I/O module may only be operated if an explosive atmosphere can be ruled out.

I/O module 750-642 may only be used in conjunction with antenna 758-910 with a max. cable length of 2.5 m.

To exceed the rated voltage no more than 40%, the supply connections must have transient protection.

The permissible ambient temperature range is 0 °C to +55 °C.

11.2.2 Special conditions for safe use (ATEX Certificate TÜV 07 ATEX 554086 X)

1. For use as Gc- or Dc-apparatus (in zone 2 or 22) the Field bus Independent I/O Modules WAGO-I/O-SYSTEM 750-*** shall be erected in an enclosure that fulfils the requirements of the applicable standards (see the marking) EN 60079-0, EN 60079-11, EN 60079-15 and EN 60079-31.
   For use as group I electrical apparatus M2 the apparatus shall be erected in an enclosure that ensures a sufficient protection according to EN 60079-0 and EN 60079-1 and the degree of protection IP64.
   The compliance of these requirements and the correct installation into an enclosure or a control cabinet of the devices shall be certified by an ExNB.
2. Measures have to be taken outside of the device that the rating voltage is not being exceeded of more than 40 % because of transient disturbances.
3. Dip-switches, binary-switches and potentiometers, connected to the module may only be actuated when explosive atmosphere can be excluded.
4. The connecting and disconnecting of the non-intrinsically safe circuits is only permitted during installation, for maintenance or for repair purposes. The temporal coincidence of explosion hazardous atmosphere and installation, maintenance resp. repair purposes shall be excluded.
   This is although and in particular valid for the interfaces “Memory-Card”, “USB”, “Fieldbus connection”, “Configuration and programming interface”, “antenna socket”, “D-Sub”, “DVI-port” and the “Ethernet interface”. These

interfaces are not energy limited or intrinsically safe circuits. An operating of those circuits is in the behalf of the operator.

5. For the types 750-606, 750-625/000-001, 750-487/003-000, 750-484, 750-633 the following shall be considered: The Interface circuits shall be limited to overvoltage category I/II/III (non mains/mains circuits) as defined in EN 60664-1.
6. For replaceable fuses the following shall be considered: Do not remove or replace the fuse when the apparatus is energized.
7. The following warnings shall be placed nearby the unit:
   WARNING – DO NOT REMOVE OR REPLACE FUSE WHEN ENERGIZED
   WARNING – DO NOT SEPARATE WHEN ENERGIZED
   WARNING – SEPARATE ONLY IN A NON-HAZARDOUS AREA

11.2.3 Special conditions for safe use (IEC-Ex Certificate TUN 09.0001 X)

1. For use as Gc- or Dc-apparatus (in zone 2 or 22) the Field bus Independent I/O Modules WAGO-I/O-SYSTEM 750-*** shall be erected in an enclosure that fulfils the requirements of the applicable standards (see the marking) IEC 60079-0, IEC 60079-11, IEC 60079-15 and IEC 60079-31.
   For use as group I electrical apparatus M2 the apparatus shall be erected in an enclosure that ensures a sufficient protection according to IEC 60079-0 and IEC 60079-1 and the degree of protection IP64.
   The compliance of these requirements and the correct installation into an enclosure or a control cabinet of the devices shall be certified by an ExCB.
2. Measures have to be taken outside of the device that the rating voltage is not being exceeded of more than 40 % because of transient disturbances.
3. DIP-switches, binary-switches and potentiometers, connected to the module may only be actuated when explosive atmosphere can be excluded.
4. The connecting and disconnecting of the non-intrinsically safe circuits is only permitted during installation, for maintenance or for repair purposes. The temporal coincidence of explosion hazardous atmosphere and installation, maintenance resp. repair purposes shall be excluded.
   This is although and in particular valid for the interfaces “Memory-Card”, “USB”, “Fieldbus connection”, “Configuration and programming interface”, “antenna socket”, “D-Sub”, “DVI-port” and the “Ethernet interface”. These interfaces are not energy limited or intrinsically safe circuits. An operating of those circuits is in the behalf of the operator.
5. For the types 750-606, 750-625/000-001, 750-487/003-000, 750-484, 750-633 the following shall be considered: The Interface circuits shall be limited to overvoltage category I/II/III (non mains/mains circuits) as defined in IEC 60664-1.
6. For replaceable fuses the following shall be considered: Do not remove or replace the fuse when the apparatus is energized.
7. The following warnings shall be placed nearby the unit:
   WARNING – DO NOT REMOVE OR REPLACE FUSE WHEN ENERGIZED
   WARNING – DO NOT SEPARATE WHEN ENERGIZED
   WARNING – SEPARATE ONLY IN A NON-HAZARDOUS AREA

11.2.4 ANSI/ISA 12.12.01

A. This equipment is suitable for use in Class I, Division 2, Groups A, B, C, D or non-hazardous locations only.
B. This equipment is to be fitted within tool-secured enclosures only.
C. WARNING Explosion hazard - substitution of components may impair suitability for Class I, Div. 2.
D. “WARNING – Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous” has to be placed near each operator accessible connector and fuse holder.
E. When a fuse is provided, the following information shall be provided: “A switch suitable for the location where the equipment is installed shall be provided to remove the power from the fuse.”
F. For devices with Ether CAT/Ethernet connectors “Only for use in LAN, not for connection to telecommunication circuits”.
G. WARNING - Use Module 750-642 only with antenna module 758-910.
H. For Couplers/Controllers and Economy bus modules only: “The configuration interface Service connector is for temporary connection only. Do not connect or disconnect unless the area is known to be non-hazardous. Connection or disconnection in an explosive atmosphere could result in an explosion.”
I. Modules containing fuses only: “WARNING - Devices containing fuses must not be fitted into circuits subject to over loads, e.g. motor circuits”
K. Modules containing SD card reader sockets only: “WARNING - Do not connect or disconnect SD-Card while circuit is live unless the area is known to be free of ignitable concentrations of flammable gases or vapors.”

Information

Additional Information

Proof of certification is available on request.
Also take note of the information given on the operating and assembly instructions.
The manual, containing these special conditions for safe use, must be readily available to the user.

12 Appendix

12.1 Mailbox Commands

This section provides an overview of all available Mailbox commands, sorted according to groups and opcodes (see appendix “Overview Sorted According to Groups and Opcodes”) and according to Mailbox commands (see appendix “Overview Sorted According to Mailbox Commands”).

Which commands can be executed with which mailbox size is indicated by symbols as follows:

- ● **Available**
  The command can be executed.

- (●) **Available to a limited extent**
  Execution of the command is possible, but only as much data as is possible for the current mailbox size is returned.

- \- **Not available**
  The command cannot be executed.

12.1.1 Übersicht sortiert nach Gruppen und Opcodes

Table 76: Overview sorted according to groups and opcodes

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
General commands							
IDLE	0x00	No job	2	2	●	●	●
Block transfer							
DLD_START	0x01	Download start of a block	6	3	●	●	●
DLD_CONT	0x02	Continue block download or upload	2/6/12/ 18	6/12/ 18	●	●	●
DLD_END	0x03	End block download or upload	5	6	●	●	●
Maintenance and firmware							
RebootHost	0x10	Warm boot of the *Bluetooth*® subsystems	2	2	●	●	●
FlashRebootHost	0x11	Save configuration with subsequent warm boot	2	2	●	●	●
GetHostFwVersion	0x12	Read host firmware version	3	6	●	●	●
GetBbFwVersion	0x13	Read firmware version of the baseband controller	2	6/12	(●)	●	●

Table 76: Overview sorted according to groups and opcodes

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
Process Image							
SetRemotePiSize	0x32	Determine the size of a slot for data transfer in the master process image	4	2	●	●	●
GetRemotePiMapping	0x33	Query remote process image parameters within the master PA	3	6	●	●	●
Device Configuration							
GetLocalDeviceName	0x40	Read local device name	2	3 ... 18	(●)	(●)	●
SetLocalDeviceName	0x41	Write local device name	3 ... 18	2	(●)	(●)	●
GetLocalMacID	0x42	Read local MAC ID	2	8	-	●	●
GetLocalDeviceClass	0x47	Read local WAGO device class	2	4	●	●	●
SetLocalDeviceClass	0x48	Write local device class	4	2	●	●	●
GetLocalOperationMode	0x49	Read local operation mode	2	4	●	●	●
SetLocalOperationMode	0 x 4 A	Set local operation mode	2	4	●	●	●
GetLocalEncryptionMode	0x4D	Read local encryption mode	2	3	●	●	●
SetLocalEncryptionMode	0x4E	Set local encyrption mode	3	2	●	●	●
GetLocalAuthenticationMode	0x4F	Read local authentication mode	2	3	●	●	●
SetLocalAuthenticationMode	0x50	Write local authentication mode	2	2	●	●	●
GetLocalPassphrase	0x51	Read local *Bluetooth*® password	2	7 ... 18	-	(●)	(●)
SetLocalPassphrase	0x52	Read *Bluetooth*® password	7 ... 18	2	-	(●)	(●)
EraseLocalAuthentication	0x53	Delete local authorization	2	2	●	●	●
GetLocalDeviceConfigLen	0x54	Read length of the flash configuration	2	4	●	●	●
GetLocalDeviceRole	0x55	Read role of the local device	2	3	●	●	●
SetLocalDeviceRole	0x56	Set role of the local device	3	2	●	●	●
SetFactorySettings	0x57	Restore factory settings	2	2	●	●	●
GetMaxTXPower	0x58	Read maximum transmission power	2	3	●	●	●
SetMaxTXPower	0x59	Set maximum transmission power	3	2	●	●	●
GetDefaultTXPower	0x60	Read default transmission power	2	3	●	●	●
SetDefaultTXPower	0x61	Set default transmission power	3	2	●	●	●

Table 76: Overview sorted according to groups and opcodes

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
Network							
ScanRemoteDevices	0x80	Scan for remote device*) in the wireless network	5	2	●	●	●
GetRemoteDeviceMacID	0x81	Read MAC ID of a remote device*)	2	9	(●)	●	●
GetRemoteDeviceName	0x82	Read device name of a remote device*)	2	6 ... 18	(●)	(●)	(●)
AllowRemoteDevice	0x83	Enter device*) in the table of allowed devices	9	2	-	●	●
GetAllowedRemoteDevices	0x84	Read remote device*) from the table of allowed devices	2	8	-	●	●
BindRemoteDevice	0x85	Grant access authorization for a device*)	3	2	●	●	●
UnbindRemoteDevices	0x86	Revoke access authorization for a device*)	3	2	●	●	●
GetBoundRemoteDevices	0x87	Read access authorization for remote devices	2	3	●	●	●
GetConnectionQoS	0x88	Read QoS settings	2	3	●	●	●
SetConnectionQoS	0x89	Set QoS settings	4	2	●	●	●
GetReconnectionTimePeriod	0 x 8 A	Read back time setting – between two attempts to establish a connection	2	4	●	●	●
SetReconnectionTimePeriod	0x8B	Set time settings – between two attempts to establish a connection	4	2	●	●	●
GetUserfriendlyName	0x8C	Read user-friendly name of an allowed device	2	3 ... 18	(●)	(●)	●
SetUserfriendlyName	0x8D	Write user-friendly name of an allowed device	3 ... 18	2	(●)	(●)	●

Table 76: Overview sorted according to groups and opcodes

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
Diagnostics							
GetLocalDeviceStatus	0xD0	Read status of the local bus module	2	6	●	●	●
GetNetworkStatus	0xD1	Read status of the wireless network	2	4	●	●	●
GetStatusMessage	0xD2	Read diagnostic information	4	6	●	●	●
GetLinkQuality	0xD5	Read link quality	3	3	●	●	●
GetLinkSignalStrength	0xD7	Read signal strength for a connection	3	3	●	●	●
GetAvailableChannelMap	0xD8	Read available hopping channels	3	12	-	●	●
SetLED	0xD9	Set LED	5	2	●	●	●
MirrorMailboxCommand	0xDA	Mirror Mailbox for test purposes	6/12/18	6/12/18	●	●	●
GetLocalUpTime	0xDB	Read operating time of the I/O module	6/8	6/8	(●)	●	●

*) 750-644 I/O module or *Bluetooth*® SPP device

12.1.2 Overview Sorted According Mailbox Commands

Table 77: Overview sorted according to Mailbox commands

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
AllowRemoteDevice	0x83	Enter device*) in the table of allowed devices	9	2	-	●	●
BindRemoteDevice	0x85	Grant access authorization for a device*)	3	2	●	●	●
DLD_CONT	0x02	Continue block download or upload	2/6//12 /18	2/6/12/1 8	●	●	●
DLD_END	0x03	End block download or upload	5	6	●	●	●
DLD_START	0x01	Download start of a block	6	3	●	●	●
EraseLocalAuthentication	0x53	Delete local authorization	2	2	●	●	●
FlashRebootHost	0x11	Save configuration with subsequent warm boot	2	2	●	●	●
GetAllowedRemoteDevices	0x84	Read remote device*) from the table of allowed devices	2	8	-	●	●
GetAvailableChannelMap	0xD8	Read available hopping channels	3	14	-	●	●
GetBbFwVersion	0x13	Read firmware version of the baseband controller	2	6/12	(●)	●	●
GetBoundRemoteDevices	0x87	Read access authorization for remote devices*)	2	3	●	●	●
GetConnectionQoS	0x88	Read QoS settings	2	3	●	●	●
GetDefaultTXPower	0x60	Read default transmission power	2	3	●	●	●
GetHostFwVersion	0x12	Read host firmware version	3	6	●	●	●
GetLinkQuality	0xD5	Read link quality	3	3	●	●	●
GetLinkSignalStrength	0xD7	Read signal strength for a connection	3	3	●	●	●
GetLocalAuthenticationMode	0x4F	Read local authentication mode	2	3	●	●	●
GetLocalDeviceClass	0x47	Read local WAGO device class	2	4	●	●	●
GetLocalDeviceConfigLen	0x54	Read length of the flash configuration	2	4	●	●	●
GetLocalDeviceName	0x40	Read local device name	2	3 ... 18	(●)	(●)	●
GetLocalDeviceRole	0x55	Read role of the local device	2	3	●	●	●
GetLocalDeviceStatus	0xD0	Read status of the local bus module	2	6	●	●	●

Table 77: Overview sorted according to Mailbox commands

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
GetLocalEncryptionMode	0x4D	Read local encryption mode	2	3	●	●	●
GetLocalMacID	0x42	Read local MAC ID	2	8	-	●	●
GetLocalOperationMode	0x49	Read local operation mode	2	4	●	●	●
GetLocalPassphrase	0x51	Read local *Bluetooth*® password	2	7 ... 18	-	(●)	(●)
GetLocalUpTime	0xDB	Read operating time of the I/O module	6/8	6/8	(●)	●	●
GetMaxTXPower	0x58	Read maximum transmission power	2	3	●	●	●
GetNetworkStatus	0xD1	Read status of the wireless network	2	4	●	●	●
GetReconnectionTimePeriod	0 x 8 A	Read back time setting – between two attempts to establish a connection	2	4	●	●	●
GetRemoteDeviceMacID	0x81	Read MAC ID of a remote device*)	2	9	(●)	●	●
GetRemoteDeviceName	0x82	Read device name of a remote device*)	2	6 ... 18	(●)	(●)	(●)
GetRemotePiMapping	0x33	Query remote process image parameters within the master PA	3	6	●	●	●
GetStatusMessage	0xD2	Read diagnostic information	4	6	●	●	●
GetUserfriendlyName	0x8C	Read user-friendly name of an allowed device	2	3 ... 18	(●)	(●)	●
IDLE	0x00	No job	2	2	●	●	●
MirrorMailboxCommand	0xDA	Mirror Mailbox for test purposes	6/12/18	6/12/18	●	●	●
RebootHost	0x10	Warm boot of the *Bluetooth*® subsystems	2	2	●	●	●
ScanRemoteDevices	0x80	Scan for remote device*) in the wireless network	5	2	●	●	●
SetConnectionQoS	0x89	Set QoS settings	4	2	●	●	●
SetDefaultTXPower	0x61	Set default transmission power	3	2	●	●	●
SetFactorySettings	0x57	Restore factory settings	2	2	●	●	●
SetLED	0xD9	Test LED function	5	2	●	●	●
SetLocalAuthenticationMode	0x50	Write local authentication mode	2	2	●	●	●
SetLocalDeviceClass	0x48	Write local device class	4	2	●	●	●
SetLocalDeviceName	0x41	Set local device name	3 ... 18	2	(●)	(●)	●
SetLocalDeviceRole	0x56	Set role of the local device	3	2	●	●	●
SetLocalEncryptionMode	0x4E	Set local encryption mode	3	2	●	●	●

Table 77: Overview sorted according to Mailbox commands

Mailbox commands	Op-code	Description	Length inquiry	Length response	May be executed with Mailbox		
					6	12	18
SetLocalOperationMode	0 x 4 A	Set local operation mode	2	4	●	●	●
SetLocalPassphrase	0x52	Read *Bluetooth*® password	7 ... 18	2	-	(●)	(●)
SetMaxTXPower	0x59	Set maximum transmission power	3	2	●	●	●
SetReconnectionTimePeriod	0x8B	Set time settings – between two attempts to establish a connection	4	2	●	●	●
SetRemotePiSize	0x32	Determine the size of a slot for data transfer in the master process image	4	2	●	●	●
SetUserfriendlyName	0x8D	Write user-friendly name of an allowed device	3 ... 18	2	(●)	(●)	●
UnbindRemoteDevices	0x86	Revoke access authorization for a device*)	3	2	●	●	●

*) 750-644 I/O module or *Bluetooth*® SPP device

12.2 Return Values of Mailbox Commands

The following standard values are defined for the return values (MBX_RESULT) of mailbox commands:

Label	Return value	Description
MBX_CMD_OK	0x00	Successful execution
MBX_CMD_GENERAL_ERROR	0x01	General error
MBX_CMD_DENIED_UNKNOWN	0x02	Unknown command
MBX_CMD_OUT_OF_RANGE	0x03	Values outside of the valid range (overrun or underrun)
MBX_CMD_INVALID_ARG	0x04	False or invalid argument
MBX_CMD_INTERNAL_ERROR	0x05	Internal error
MBX_CMD_TIMEOUT	0x06	Time overrun of the command
MBX_CMD_DENIED_NOT_APPLICABLE	0x07	Prerequisites for command not fulfilled: false operation mode, false device role or necessary precursor command not executed
MBX_CMD_DENIED_NOT_IMPLEMENTED	0x08	Command reserved for later implementation
MBX_CMD_DENIED_MBX_TOO_SMALL	0x09	Mailbox too small for return value
MBX_CMD_DENIED_BUSY	0x0A	Current or precursor command being executed, no valid data is available yet. Recommendation: call up command again after a short waiting period.
MBX_CMD_INVALID	0x0B	System or network configuration is defective

All Mailbox commands use these return values to signal the status of the command execution. If individual return values offer additional interpretations for specific Mailbox commands, this is explained in more detail in the description of the respective command.

In principle, the first return value with a mirrored mailbox command and toggle bit is considered a valid response. As soon as this happens, the next mailbox command can be executed. Many commands result in a restart of the 750-644 I/O module. If a Mailbox command that triggers a restart is not replaced by another command after receiving the response, the 750-644 I/O module recognizes the unaltered, existing command when it restarts. The command is rejected with MBX_CMD_DENIED_BUSY to prevent an endless loop of resets. This offers the possibility of determining the successful conclusion of a reset by monitoring change of the return value.

12.3 Reference for Mailbox Commands

In this section, the requirements for the execution of each Mailbox command in question are represented as follows:

Mailbox size (6, 12 or 18 bytes)

●	**Available** The command can be executed.
(●)	**Available to a limited extent** Execution of the command is possible, but only as much data as is possible for the current mailbox size is returned.
-	**Not available** The command cannot be executed.

Operating mode (Configuration/real-time or ad-hoc communication)

●	**Available** The command can be executed.
-	**Not available** The command cannot be executed.

Device role (master, slave)

●	**Available** The command can be executed.
-	**Not available** The command cannot be executed.

Save configuration

●	With this command, module settings are changed. This change is first undertaken on a temporary image of the module configuration. The temporary image is loaded during a restart of the 750-644 module from the non-temporary image (flash memory). Changes must be applied to the non-temporary image so that they remain after a cold start or power loss. To update the non-temporary image, perform a warm boot (see appendix “FlashRebootHost”). Alternatively, change the operating mode (see appendix “setLocalOperationMode”). A warm boot is automatically performed (see figure “Saving the configuration”).
-	No data are saved for this command.

Reboot

●	The 750-644 I/O module performs a restart after executing the command.
-	The 750-644 I/O module does not restart after executing the command.

In addition, configuration of the bytes is described during query and response with arguments and return values. If no return values are present, the related tables are presented in gray.

Note

Populating the bytes
If the query is smaller than the Mailbox, the remaining bytes in the Mailbox should be populated with 0x00 during the query. If the size of the response is smaller than the size of the Mailbox, the remaining bytes in the Mailbox of the 750-644 I/O module are populated with 0x00.

12.3.1 General Commands

12.3.1.1 No Task

(IDLE, 0x00)
If the opcode = 0x00, no task is performed. This command is available in all operating modes for all Mailbox sizes.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_IDLE							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_IDLE							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
-	-	-

12.3.2 Block Transfer

12.3.2.1 Download Start of a Block

DLD_START, 0x01
The block transfer is started with the call up. A new DLD_START with no previous DLD_END breaks the transfer off and initializes a new transfer.
The command block is concluded by DLD_END.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_DLD_START							
1	T	-						
2	MBX_MB3							
3	MBX_MB4							
4	MBX_MB5							
5	MBX_MB6							

Arguments

Mode	Parameters	Value	Description
Configuration	MBX_MB3	0x01	Block type: RAM block
	MBX_MB4	0x06	Write/read configuration
		0x07	Read query result
		0x09	Read name of the remote 750-644 I/O module
		0x0A	Read/write complete name of the local 750-644 I/O module
		0x0B	Read/write password of the local 750-644 I/O module
	MBX_MB5	0x00	Number of the block in the whole transmission (LSB)
	MBX_MB6	Bit 0 ... 5	Number of the block in the whole transmission (MSB)
		Bit 6, 7	0x80 – Download, write to the 750-644 I/O module
			(bit 6=0, bit 7=1)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_IDLE							
1	T	MBX_RESULT						
2	MBX_DLD_RESULT							

Return value

Parameters	Value	Description
MBX_DLD_RESULT	DLD_OK (0x00)	No error The block transfer has been started.
	DLD_DOWNLOAD_NOT_STARTED (0x01)	The block transfer has not been started. An undefined block is supposed to be transmitted.
	DLD_OK_ABORTED (0x02)	A block transfer is currently active. No new transfer is being started.
	DLD_ERROR (0x31)	A non-supported writing or ready operation has been started.
	DLD_ERROR_TABLE_READ_ONLY (0x32)	A download is supposed to take place to a protected area of the configuration.
MBX_RESULT	MBX_CMD_GENERAL_ERROR	"Configuration" mode. An error has occurred. More detailed information in MBX_DLD_RESULT
	MBX_CMD_DENIED_NOT_APPLICABLE	The command was called up in "Communication" mode.

12.3.2.2 Continuing Block Download or Upload

DLD_CONT, 0x02

When called, the uploading/downloading of a block is continued. During an upload of data to the 750-644 I/O module, the data bytes from byte 2 may be ignored.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_DLD_CONT							
1	T	-						
2	DATA							
3	DATA							
4	DATA							
5	DATA							
6	OPTIONAL DATA							
…	…							
17	OPTIONAL DATA							

Arguments

Parameters	Value	Description
DATA OPTIONAL DATA	[0x00 …0xFF]	Transmitted data bytes In "Configuration" mode, the number of data bytes results from Mailbox size -2. During a download from the 750-644 I/O module, the values of the data bytes are ignored.

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_DLD_CONT							
1	T	MBX_RESULT						
2	DATA							
3	DATA							
4	DATA							
5	DATA							
6	OPTIONAL DATA							
…	...							
33	OPTIONAL DATA							

Return value

Parameters	Wert	Description
MBX_RESULT	MBX_CMD_OK	No error occurred. The block transfer has been continued.
	MBX_CMD_DENIED_ NOT_APPLICABLE	There is no active transfer. The command is invalid.
	MBX_CMD_OUT_OF_RANGE	An attempt was made to transfer more than 512 bytes.
DATA OPTIONAL DATA	[0x00 … 0xFF]	Transmitted data bytes: In "Configuration" mode, the number of data bytes results from Mailbox size -2. During an upload of data to the 750-644 I/O module, the data bytes are initialized in the response with 0x00 and can be ignored.

12.3.2.3 Stopping Block Download or Upload

DLD_END, 0x03

When called, the uploading and downloading of a block is ended.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●*)	●*)

*) A restart is only performed after successful write operations in "Configuration" mode.

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_DLD_END							
1	T	-						
2	MBX_CHECKSUM (LSB)							
3	MBX_CHECKSUM							
4	MBX_CHECKSUM (MSB)							

Arguments

Parameter	Value	Description
MBX_CHECKSUM	The value is determined by bytewise addition of the transmitted values.	Checksum of the block: The content is dependent on the transmitted data bytes.

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_DLD_END							
1	T	MBX_RESULT						
2	DLD_RESULT							
3	MBX_CHECKSUM (LSB)							
4	MBX_CHECKSUM							
5	MBX_CHECKSUM (MSB)							

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OK	Transfer successful
	MBX_CMD_GENERAL_ERROR	An error has occurred. More detailed information in DLD_RESULT
DLD_RESULT	DLD_OK (0x00)	No error occurred
	DLD_ERROR_DOWNLOAD_NOT _STARTED (0x01)	There is no active transfer. The command is invalid.
	DLD_ERROR_CHECKSUM (0x32)	Checksum error
	DLD_ERROR_UNDERFLOW (0x33)	Underflow, not enough data.
	DLD_ERROR_DATASET_CORRUPT (0x38)	“Configuration” mode: Written block in the "Extended Register" is faulty.
DLD_CHECKSUM	The checksum calculated on the *Bluetooth*® subsystem for the upload (*Bluetooth*® subsystem to PLC)	Checksum

12.3.3 Maintenance and Firmware

12.3.3.1 Warm Booting the *Bluetooth*® Subsystem

RebootHost, 0x10
When called, the *Bluetooth*® subsystem is restarted. All radio connections are broken off.

Note

Breaking off all radio connections and rebooting without saving!
If "RebootHost" is called up, all radio connections are broken off. This command causes a reboot with no prior saving of the configuration. Therefore, changes made since the last time the configuration was saved are lost.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	●

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_REBOOTHOST							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_REBOOTHOST							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
-	-	-

12.3.3.2 Save Configuration with Subsequent Warm Boot

FlashRebootHost, 0x11
When called, the current configuration of the *Bluetooth*® subsystem is written to the flash memory. Then the *Bluetooth*® subsystem is rebooted.

Note

Breaking off all radio connections
If "FlashRebootHost" is called up, all radio connections are broken off.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	●

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_FLASHREBOOTHOST							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_FLASHREBOOTHOST							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_DENIED_BUSY	A block-oriented command is active in the execution.

12.3.3.3 Reading the Host Firmware Version

GetHostFwVersion, 0x12
When called, version information of the firmware components of the *Bluetooth*® subsystem are read.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETHOSTFWVERSION							
1	T	-						
2	MBX_FW_ID							

Arguments

Parameters	Value	Description
MBX_FW_ID	MBX_CM_GETHOSTFWVERSION_ BOOTLOADER (0x01)	Read version of boot loader
	MBX_CM_GETHOSTFWVERSION _FIRMWARE (0x02)	Read the version of the *Bluetooth*® subsystem firmware
	MBX_CM_GETHOSTFWVERSION _CONFIGURATION (0x03)	Read version of configuration

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETHOSTFWVERSION							
1	T	MBX_RESULT						
2	MBX_FW_ID							
3	MBX_VN_MAJOR							
4	MBX_VN_MINOR							
5	MBX_VN_RELEASE							

Retrurn value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	Invalid value for MBX_FW_ID
MBX_FW_ID	MBX_CM_GETHOSTFWVERSION _BOOTLOADER (0x01)	Version Boot Loader
	MBX_CM_GETHOSTFWVERSION _FIRMWARE (0x02)	Version of the *Bluetooth*® subsystem
	MBX_CM_GETHOSTFWVERSION _CONFIGURATION (0x03)	Version Configuration
MBX_VN_MAJOR	[0 … 255]	Main version number
MBX_VN_MINOR	[0 … 255]	Subversion number
MBX_VN_RELEASE	[0 … 255]	Release of subversion

12.3.3.4 Reading the Firmware Version of the Baseband Controller

GetBbFwVersion, 0x13
When called, the version information of the baseband controller can be read.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETBBFWVERSION							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETBBFWVERSION							
1	T	MBX_RESULT						
2	Fw_Status							
3	Fw_HCI_Version							
4	Fw_HCI_Revision (High)							
5	Fw_HCI_Revision (Low)							
6	Fw_LMP_Version							
7	Fw_Manufacturer_Name (High)							
8	Fw_Manufacturer_Name (Low)							
9	Fw_LMP_Subversion (High)							
10	Fw_LMP_Subversion (Low)							

Return value

Parameters	Value	Description
-	-	-

12.3.4 Process Image (PI)

12.3.4.1 Determining the Size of a Slot for Data Transfer in the Master PI

SetRemotePiSize, 0x32
This command is used to limit the process image (PI) of a remote 750-644 I/O module within the master PI to n bytes. All slave process images in the master together must not be larger than the set master PI -2. Two bytes of the total size are necessary for the control/status byte and an internal byte (null byte). The size of the master process image can be queried and configured over the parameter channel. It is contained in the LSB by parameter 0.

By downsizing the available PI, the sum of the configured cut-offs may exceed the size of the master PI. In this case, the output configuration is already invalid. In such a case, the configuration is executed, but the error value MBX_CMD_INVALID_CONFIGURATION is displayed. If the output configuration is correct, a command that leads to an invalid configuration is acknowledged and rejected with an error.

For ad-hoc slots 8 … 12, the cut-off must be at least 2 and may not exceed 16**.**

Note

Only effective when changed to the master role
The process image mapping can also be configured in the slave role, but only takes effect when changed to the master role.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETREMOTEPISIZE							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							
3	CUTOFF_N_BYTES							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE_ AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)
CUTOFF_N_BYTES	[0 … 46]	Number of bytes after which the slave process image is cut off. The redundant bytes are lost.

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETREMOTEPISIZE							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_CONFIGURATION	Before and after the command, the sum of all CUTOFF_N_BYTES is larger than the available master process image.
	MBX_CMD_INVALID_ARG	No valid target table has been chosen.
	MBX_CMD_OUT_OF_RANGE	With the given value, the sum of all CUTOFF_N_BYTES would exceed the limit of the available master process image or the indicated index is too large.

12.3.4.2 Querying Remote Process Image Parameters within the Master PI

GetRemotePiMapping, 0x33

With this command, the settings for a slot in the local PI are queried. 13 slots are available. Slots 1 to 7 are occupied by the fields of the table “WAGO devices” and slots 8 to 13 by the fields of the table “External Devices”.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETREMOTEPIMAPPING							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_ TABLE_AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table "2" for WAGO_DEVICE (slots 1 … 7) "1" for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETREMOTEPIMAPPING							
1	T	MBX_RESULT						
2	MBX_DEVICE_INDEX							
3	OFFSET							
4	- reserved -							
5	CUTOFF_N_BYTES							

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	No valid target table has been chosen.
	MBX_CMD_OUT_OF_RANGE	The specified index is too large.
MBX_DEVICE_INDEX	[0 … 12]	Slot of the slave PI within the master PI
OFFSET	[0 … 45]	Position of the first byte of the slot in the local PI, relative to the C/S byte. Slot 1 always has an offset of 0.
CUTOFF_N_BYTES	Quantity	Number of bytes after which the slave process image is cut off.
- reserved -	0x00	Reserved for later use.

12.3.5 Device Configuration

12.3.5.1 Reading Local Device Names

GetLocalDeviceName, 0x40

The characters of the *Bluetooth*® name of the local 750-644 I/O module are read by this query. The number of characters returned depends on the configured name, but has a maximum of [Mailbox size -3].

Note

Device name
The complete device name can be a maximum of 15 characters. The complete device name can be queried with DLD commands regardless of the Mailbox size.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	(●)	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICENAME							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICENAME							
1	T	MBX_RESULT						
2	MBX_NAME_LENGTH							
3	CHAR1							
…	…							
17	CHAR15							

Return values

Parameters	Value	Description
MBX_NAME_LENGTH	[0 … 255]	Number of characters of the complete name
CHARn	[0 … 255]	Characters of the device name in ASCII code Example: “ABC” A = CHAR1 = 0x41 B = CHAR2 = 0x42 C = CHAR3 = 0x43

12.3.5.2 Wiriting Local Device Names

SetLocalDeviceName, 0x41
With this instruction, the *Bluetooth*® name of the local 750-644 I/O module is set. The normal set of ASCII characters is available.

Note

Device name
The use of special characters (e.g. line breaks) is possible, but should be avoided. The complete device name can be a maximum of 15 characters. The complete device name can be read and written with DLD commands regardless of the Mailbox size.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	(●)	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALDEVICENAME							
1	T	-						
2	MBX_NAME_LENGTH							
3	CHAR1							
…	…							
17	CHAR15							

Arguments

Parameters	Value	Description
MBX_NAME_LENGTH	[1 … 15]	Number of the transferred characters in the name
CHARn	[1 … 15]	Characters of the device name in ASCII code Example: "ABC" A = CHAR1 = 0x41 B = CHAR2 = 0x42 C = CHAR3 = 0x43

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALDEVICENAME							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	MBX_NAME_LENGTH is equal to 0 or greater than (Mailbox size -3)

12.3.5.3 Reading Local MAC IDs

GetLocalMacID, 0x42

With this instruction, the *Bluetooth*® MAC ID (48-bit address) of the local 750-644 I/O module is read.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
-	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALMACID							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALMACID							
1	T	MBX_RESULT						
2	MAC-ID Byte 0 (LSB)							
3	MAC-ID Byte 1							
4	MAC-ID Byte 2							
5	MAC-ID Byte 3							
6	MAC-ID Byte 4							
7	MAC-ID Byte 5 (MSB)							

Return values

Parameters	Value	Description
MAC-ID byte n	[0 … 255]	The bytes of the MAC address

12.3.5.4 Reading a Local WAGO Device Class

GetLocalDeviceClass, 0x47

With this instruction, the WAGO device class of the local 750-644 I/O module is read. The device class can be used to differentiate I/O module types. Grouping of I/O modules by their tasks is also possible. When searching for I/O modules with a certain device class, an inquiry using the *Bluetooth*® Class-of-Device can help. The WAGO device classes have only an indirect relation to the *Bluetooth*® Class-of-Device.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICECLASS							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICECLASS							
1	T	MBX_RESULT						
2	WAGO_Deviceclass							
3	WAGO_SubDeviceclass							

Return values

Parameters	Value	Description
WAGO_Deviceclass	[0 … 7]	Device class
WAGO_SubDeviceclass	[0 … 7]	Subdevice class

12.3.5.5 Writing a Local Device Class

SetLocalDeviceClass, 0x48
With this instruction, the WAGO device class of the local 750-644 I/O module is written. The device class can be used to differentiate I/O module types. Grouping of I/O modules by their tasks is also possible. When searching for I/O modules with a certain device class, an inquiry using the *Bluetooth*® Class-of-Device can help. The device classes have only an indirect relation to the *Bluetooth*® Class-of-Device.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALDEVICECLASS							
1	T	-						
2	WAGO_Deviceclass							
3	WAGO_SubDeviceclass							

Arguments

Parameters	Value	Description
WAGO_Deviceclass	[0 … 7]	Device class
WAGO_SubDeviceclass	[0 … 7]	Subdevice class

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALDEVICECLASS							
1		MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	Invalid value for WAGO_deviceclass or WAGO_SubDeviceclass

12.3.5.6 Reading a Local Operating Mode

GetLocalOperationMode, 0x49
With this command, the operating mode of the local 750-644 I/O module is read.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALOPERATIONMODE							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALOPERATIONMODE							
1	T	MBX_RESULT						
2	MBX_OPMODE_ID							
3	MBX_COMMROFILE_ID							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	Parameter value(s) invalid
MBX_OPMODE_ID	MBX_CM_OPMODE_CONF (0x01)	“Configuration” mode
	MBX_CM_OPMODE_COMM (0x02)	“Communication” mode
MBX_COMMROFILE_ID	MBX_CM_OPPROFILE_REALTIME (0x01)	“Real-Time Communication” mode
	MBX_CM_OPPROFILE_CONFIG (0x02)	“Configuration” mode
	MBX_CM_OPPROFILE_ADHOC (0x03)	“Ad-hoc Communication” mode

12.3.5.7 Setting the Local Operating Mode

SetLocalOperationMode, 0x4A
When called, the operating mode of the *Bluetooth*® subsystems is set. The call is followed by a warm boot of the 750-644 I/O module in the selected operating mode, saving any changes made to the configuration.

Note

Behavior when selecting an operating mode already in use
If an operating mode is selected that has already been accepted by the 750-644 I/O module, then the command is acknowledged with MBX_CMD_OK, but there is no reboot of the 750-644 I/O module and no changes made to the configuration are saved.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	●	●

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALOPERATIONMODE							
1	T	-						
2	MBX_OPMODE_ID							
3	MBX_COMMROFILE_ID							

Arguments

Parameters	Value	Description
MBX_OPMODE_ID	MBX_CM_OPMODE_CONF (0x01)	"Configuration" mode
	MBX_CM_OPMODE_COMM (0x02)	"Communication" mode
MBX_COMMROFILE_ID	MBX_CM_OPPROFILE_REALTIME (0x01)	"Real-Time Communication" mode
	MBX_CM_OPPROFILE_CONFIG (0x02)	"Configuration" mode
	MBX_CM_OPPROFILE_ADHOC (0x03)	"Ad-hoc Communication" mode

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_CMD_SETLOCALOPERATIONMODE							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	An invalid value for one of the arguments or an invalid combination was chosen.

12.3.5.8 Reading the Local Encryption Mode

GetLocalEncryptionMode, 0x4D
When called, the encryption mode for the radio transmission is read.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALENCRYPTIONMODE							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALENCRYPTIONMODE							
1	T	MBX_RESULT						
2	MBX_ENCRYPTION_MODE							

Return values

Parameters	Value	Description
MBX_ENCRYPTION_MODE	MBX_ENCRYPT_ENABLE (0x01)	Encryption is enabled (default)
	MBX_ENCRYPT_DISABLE (0x00)	Encryption is disabled

12.3.5.9 Setting the Local Encryption Mode

SetLocalEncryptionMode, 0x4E
With this command, the encryption of the *Bluetooth*® data transmission is enabled or disabled. This setting can be done independently of the device role, but only affects the master. If encryption is enabled, the 750-644 I/O modules or *Bluetooth*® SPP devices that do not use encryption cannot connection.

Note

Make sure the settings of the devices match!
Encryption can be enabled without enabling an authentication. The actual encryption of the data takes place after an authentication. The security of the encryption is linked to the quality of the password. Connections between devices can only be established if the settings for encryption, authentication and password are synchronized. This can be achieved by having identical settings for the devices to be connected.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALENCRYPTIONMODE							
1	T	-						
2	MBX_ENCRYPTION_MODE							

Arguments

Parameters	Value	Description
MBX_ENCRYPTION_MODE	MBX_ENCRYPT_ENABLE (0x01)	Enable encryption
	MBX_ENCRYPT_DISABLE (0x00)	Disable encryption

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALENCRYPTIONMODE							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	An unknown argument has been passed.

12.3.5.10 Reading the Local Authentication Mode

GetLocalAuthenticationMode, 0x4F
Will call up, the locally set authentication mode of the *Bluetooth*® subsystems is read.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETAUTHENTICATIONMODE							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETAUTHENTICATIONMODE							
1	T	MBX_RESULT						
2	MBX_AUTHENTICATION_MODE							

Return values

Parameters	Value	Description
MBX_AUTHENTICATION_MODE	MBX_AUTHENTICATION_NONE (0x01)	No authorization necessary
	MBX_AUTHENTICATION_PIN (0x02)	Authentication is conducted with a PIN created from the password at each establishment of a connection.
	MBX_AUTHENTICATION_LINKKEY (0x03)	Authorization through “Link Key” (the PIN is not requested each time a connection is established, but the “Link Key” saved in the flash is used).

12.3.5.11 Setting Local Authentication Mode

SetLocalAuthenticationMode, 0x50
When calling up “SetLocalAuthenticationMode”, the local authentication mode of the *Bluetooth*® subsystems is set. If authentication is disabled (MBX_AUTHENTICATION_NONE), the identity of the communication partner is not verified. If authentication is enabled, the 750-644 I/O modules authenticate each other each time a connection is established. This process occurs with cryptographic protection based on the *Bluetooth*® standard.

If MBX_AUTHENTICATION_LINKKEY is set as the authentication mode, an individual “Link Key” is calculated from the configured PIN during the first connection. The PIN is generated from the password. If this key has been generated once, the 750-644 I/O modules are considered to be “paired” (connected) and do not need to conduct a mutual authentication again with a new connection. If the “Link Key” is deleted, for example when restarting the device or by means of “EraseLocalAuthentication”, then the password is requested again in order to calculate the “Link Key”. Accordingly, for *Bluetooth*® SPP devices, you are prompted to enter the password at initial or subsequent authentication. For 750-644 I/O modules, the password is archived in the *Bluetooth*® subsystem and does not have to be reentered once it has been created correctly.

In authentication mode MBX_AUTHENTICATION_PIN, authentication is performed with the PIN instead of using the “Link Key”. For 750-644 I/O modules, this is automatic based on the saved password. For *Bluetooth*® SPP devices, the password usually has to be manually reentered each time a connection is established.

Note

Authentication
Authentication only ensures that communication partners verify each other's identity. Authentication does not prevent data interception. Authentication is required for encryption when transmitting data. 750-644 I/O modules can only connect with each other when they have the same settings for encryption, authentication and password.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETAUTHENTICATIONMODE							
1	T	-						
2	MBX_AUTHENTICATION_MODE							

Arguments

Parameters	Value	Description
MBX_AUTHENTICATION _MODE	MBX_AUTHENTICATION _NONE (0x01)	No authorization necessary
	MBX_AUTHENTICATION _PIN (0x02)	Authentication is conducted with a PIN created from the password at each establishment of a connection.
	MBX_AUTHENTICATION _LINKKEY (0x03)	Authorization by “Link Key” (the PIN is not requested each time a connection is established, rather the “Link Key” saved in the flash is used).

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETAUTHENTICATIONMODE							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	No valid value passed with MBX_ AUTHENTICATION_MODE

12.3.5.12 Reading the Local *Bluetooth*® Password

GetLocalPassphrase, 0x51

When called, the local password for radio transmission is read. The password is transmitted as a byte value representation of ASCII characters and is at least 4 characters long.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
-	(●)	(●)	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALPASSPHRASE							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALPASSPHRASE							
1	T	MBX_RESULT						
2	MBX_PASSPHRASE_Length							
3	MBX_ PASSPHRASE_Byte 1							
4	MBX_ PASSPHRASE_Byte 2							
5	MBX_ PASSPHRASE_Byte 3							
6	MBX_ PASSPHRASE_Byte 4							
7	MBX_ PASSPHRASE_Byte 5							
…	…							
17	MBX_ PASSPHRASE_Byte 15							

Note

The password is not completely displayed if too long!
If the password is longer than the available Mailbox, the excess bytes are cut off. MBX_ PASSPHRASE _Length returns the actual length of the password. Therefore, the real password may deviate from the password displayed.

Return values

Parameters	Value	Description
MBX_PASSPHRASE_Length	[4 … 15]	Complete length of the password
MBX_PASSPHRASE_Byte n	Characters (ASCII)	Password as ASCII representation

12.3.5.13 Writing the Local *Bluetooth*® Password

SetLocalPassphrase, 0x52
This command is used to configure the local password. The 750-644 I/O module calculates the “Link Key” from the local password, which is necessary during active authentication to establish the connection and to encrypt data. The *Bluetooth*® password must be identical for all devices that want to communicate with each other.

Note

Choosing a password
Security depends on the password you choose. The password should be as long and random as possible. 750-644 I/O modules can only connect with each other when they have the same settings for encryption, authentication and password.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
-	(●)	(●)	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALPASSPHRASE							
1	T	-						
2	MBX_PSW_Length							
3	MBX_PSW_Byte 1							
4	MBX_PSW_Byte 2							
5	MBX_PSW_Byte 3							
6	MBX_PSW_Byte 4							
7	OPTIONAL MBX_PASSPHRASE_Byte 5							
…	…							
17	OPTIONAL MBX_PASSPHRASE_Byte 15							

Arguments

Parameters	Value	Description
MBX_PASSPHRASE_Length	[4 … 15]	Password length
MBX_ PASSPHRASE_Byte n	Characters (ASCII)	Password as ASCII representation

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALPASSPHRASE							
1	T	MBX_RESULT						

Return value

Parameter	Value	Description
MBX_RESULT	MBX_CMD_INVALID_ARG	The password length is shorter than 4 characters and is therefore not long enough or MBX_PASSPHRASE_Length indicates a value that is larger than the payload of the Mailbox.

12.3.5.14 Erasing Local Authorization

EraseLocalAuthentication, 0x53
When called, the local password for authorization is erased. Then a warm boot is carried out.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_ERASELOCALAUTHENTICATION							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_ERASELOCALAUTHENTICATION							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
-	-	-

12.3.5.15 Reading the Length of the Flash Configuration

GetLocalDeviceConfigLen, 0x54
When called, the length (in bytes) of the local configuration in the flash of the *Bluetooth*® subsystem is returned. This information is used by the PLC to interpret the data from the block commands.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICECONFIGLEN							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICECONFIGLEN							
1	T	MBX_RESULT						
2	MBX_CONFIG_LENGTH (LSB)							
3	MBX_CONFIG_LENGTH (MSB)							

Return value

Parameters	Value	Description
MBX_CONFIG_LENGTH	[0 … 65535]	Length of the configuration (number of bytes) saved in the flash.

12.3.5.16 Reading the Roll of the Local Device

GetLocalDeviceRole, 0x55
This command queries the role that the local 750-644 I/O module takes in the piconet (master or slave).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICEROLE							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICEROLE							
1	T	MBX_CDM_RESULT						
2	MBX_DEVICE_ROLE							

Return values

Parameters	Value	Description
MBX_DEVICE_ROLE	MBX_ROLE_COORDINATOR (0x01)	Device role: Master
	reserved (0x02)	Reserved
	MBX_ROLE_ENDDEVICE (0x03)	Device role: Slave

12.3.5.17 Setting the Role of the Local Device

SetLocalDeviceRole, 0x56

This command sets the role that the local 750-644 I/O module should take in the piconet (master or slave).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALDEVICEROLE							
1	T	-						
2	MBX_DEVICE_ROLE							

Arguments

Parameters	Value	Description
MBX_DEVICE_ROLE	MBX_ROLE_COORDINATOR (0x01)	Master
	MBX_ROLE_ROUTER (0x02)	Router
	MBX_ROLE_ENDDEVICE (0x03)	Slave

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLOCALDEVICEROLE							
1	T	MBX_CMD_RESULT						

Return values

Parameters	Value	Description
MBX_CMD_RESULT	MBX_CMD_DENIED_NOT _IMPLEMENTED	The parameter is not implemented (router)
	MBX_CMD_INVALID_ARG	Invalid value for MBX_DEVICE_ROLE

12.3.5.18 Restoring Factory Settings

SetFactorySettings, 0x57
This command is used to overwrite the local configuration saved in the flash with the factory settings. The *Bluetooth*® subsystem is then rebooted.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	●	●

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETFACTORYSETTINGS							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETFACTORYSETTINGS							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
-	-	-

12.3.5.19 Reading the Maximum Transmission Power

GetMaxTXPower, 0x58
This command reads the limit of the maximum transmission power. The "SetMaxTXPower" command sets the limit and defines to what value the 750-644 I/O module may increase its maximum transmission power. The "maxTXpower" parameter is always equal to or greater than the "defaultTXpower" parameter (value between 0 and 20 dBm).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETMAXTXPOWER							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETMAXTXPOWER							
1	T	MBX_RESULT						
2	MBX_MAXTXPOWER							

Return value

Parameters	Value	Description
MBX_MAXTXPOWER	[0 … 20]	Transmission power in dBm

12.3.5.20 Setting the Maximum Transmission Power

SetMaxTXPower, 0x59

This command is used to set the limit of the maximum transmission power. *Bluetooth*® devices are able to adjust the transmission power to the current requirements. If reception is strong, the transmitter automatically reduces the power. If reception is weak, the transmitter can increase the power up to the technical upper limit of 20 dBm. In certain applications, it may be desirable to further reduce the power by means of manual configuration. This also causes an effective reduction in the working range. This can prevent any negative impact on other radio systems from a certain distance.

The “maxTXpower” parameter is always equal to or greater than the “defaultTXpower” parameter (value between 0 and 20 dBm).

Note

Use in North America and Taiwan

When used in North America, reduce the “maxTXpower” parameter to 14 dBm or less.

When used in Taiwan, reduce the “maxTXpower” to 18 dBm or less.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETMAXTXPOWER							
1	T	-						
2	MBX_MAXTXPOWER							

Arguments

Parameters	Value	Description
MBX_MAXTXPOWER	[0 … 20]	Maximum transmission power in dBm

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_ GETMAXTXPOWER							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	Invalid value. The MBX_MAXTXPOWER value was outside the permitted range of 0 dBm to 20 dBm. The value was not written.
	MBX_CMD_INVALID_ARG	Bad value. The transferred MBX_MAXTXPOWER value was less than the MBX_DEFAULTTXPOWER value already set. The value was not written.

12.3.5.21 Reading the Default Transmission Power

GetDefaultTXPower, 0x60
This command is used to read what transmission power the 750-644 I/O module uses when scanning for available devices (inquiry) or when establishing connections to other devices (paging). The "SetDefaultTxPower" command can be used to change this setting.

The "defaultTXpower" parameter is always equal to or less than the "maxTXpower" parameter (value between 0 and 20 dBm).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETDEFAULTTXPOWER							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETDEFAULTTXPOWER							
1	T	MBX_RESULT						
2	MBX_DEFAULTTXPOWER							

Return value

Parameters	Value	Description
MBX_DEFAULTTXPOWER	[0 … 20]	Transmission power in dBm

12.3.5.22 Setting the Default Transmission Power

SetDefaultTXPower, 0x61
This command is used to set what transmission power the 750-644 I/O module

uses when scanning for available devices (inquiry) or when establishing new connections to other devices (paging).

In this way, the working range within which remote 750-644 I/O modules or *Bluetooth*® SPP devices can be found and connected to can be restricted. Once a connection has been successfully established, the transmission power can again be automatically adjusted to the needs (see "SetMaxTXpower" command).

The "defaultTXpower" parameter is always equal to or less than the "maxTXpower" parameter (value between 0 and 20 dBm).

Note

Mobile Devices
If the "defaultTXpower" is significantly less than the "maxTXpower", mobile devices can easily increase the distance from one another when communication is stable. When rebooting or the connection is lost, the devices may be unable to reconnect.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETMAXTXPOWER							
1	T	-						
2	MBX_DEFAULTTXPOWER							

Arguments

Parameters	Value	Description
MBX_DEFAULTTXPOWER	[0 … 20]	Default transmission power in dBm

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_ GETMAXTXPOWER							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	Invalid value. The MBX_DEFAULTTXPOWER value was outside the permitted range of 0 dBm to 20 dBm.
	MBX_CMD_INVALID_ARG	Bad value. The transferred MBX_MAXTXPOWER value was greater than the MBX_DEFAULTTXPOWER value already set. The value was not written.

12.3.6 Network

12.3.6.1 Scanning for a Remote Device in the Wireless Network

ScanRemoteDevices, 0x80
When called, the system scans for remote 750-644 I/O modules or *Bluetooth*® SPP devices. The scanning process is asynchronous, i.e. the results are not immediately available. As long as the scan runs, the wireless module is not available for any other function. Functions that do not use the wireless module are carried out normally. When the scan concludes, devices found are entered in a list from which they can be individually queried with the command "GetRemoteDeviceMacID". The complete CoD for the 750-644 I/O module is 0x0020f8 (hexadecimal).

To limit the scan to certain devices, a Class-of-Device (CoD) can be indicated. If a CoD not equal to 0 is used, only those devices are found that have this exact CoD. If CoD = 0 is used, all 750-644 I/O modules and *Bluetooth*® SPP devices in the environment are scanned.

Note

Inquiry results
The DLD commands can be used to read complete result of the inquiry.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SCANREMOTEDEVICES							
1	T							
2	MBX_COD (LSB)							
3	MBX_COD							
4	MBX_COD (MSB)							

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SCANREMOTEDEVICES							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_DENIED_BUSY	A running scan process or another function is blocking the wireless module.
MBX_COD	24 Bit	Class-of-Device of the devices*) to be scanned. With MBX_COD = 0x0, the CoD is ignored.

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.6.2 Reading the MAC ID of a Remote Device

GetRemoteDeviceMacID, 0x81
This command is used to access a list of 750-644 I/O modules or *Bluetooth*® SPP devices visible in the area and to query the *Bluetooth*® MAC ID of a remote 750-644 I/O module or *Bluetooth*® SPP device. This command requires prior execution of a scan process with the command "ScanRemoteDevices", which initiates creation of this list. If an attempt is made to access the list before the scan process is complete, the command responds with MBX_CMD_DENIED_BUSY. In this case, the query should be repeated after a few seconds.

If this command is used with a Mailbox size of 6 bytes, the command only returns the number of devices found. To query the MAC IDs of the devices, however, the block transfer can be used (see DLD_Start, DLD_Cont, DLD_End commands).

Note

Execute the "ScanRemoteDevices" beforehand!
Before calling up "GetRemoteDeviceMacID", the "ScanRemoteDevices" command must be executed (see appendix "Scanning for a Remote Device in the Wireless Network").

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETREMOTEDEVICEMACID							
1	T	-						
2	MBX_DEVICE_INDEX							

Arguments

Parameters	Value	Description
MBX_DEVICE_INDEX	[0 … 15]	Index of the device*) whose MAC ID is to be read. A maximum of 16 found devices*) are administered.

*)750-644 I/O module or *Bluetooth*® SPP device

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETREMOTEDEVICEMACID							
1	T	MBX_RESULT						
2	MBX_NR_FOUND_DEVICES							
3	MBX_MACID_BYTE (LSB)*)							
4	MBX_MACID_BYTE*)							
5	MBX_MACID_BYTE*)							
6	MBX_MACID_BYTE*)							
7	MBX_MACID_BYTE*)							
8	MBX_MACID_BYTE (MSB)*)							

*) Applies to Mailbox sizes 12 and 18 bytes

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_DENIED_BUSY	The scan process has not yet been stated or concluded.
	MBX_CMD_OUT_OF_RANGE	The specified index is greater than or equal to the number of the devices found*) or no devices*) with the specified Class-of-Device were found.
MBX_MACID_BYTE	Bytes of the MAC ID	Valid if MBX_RESULT = MBX_CMD_OK. Parameter is not returned when Mailbox size is 6 bytes.
MBX_NR_FOUND_DEVICES	[0 … 15]	Number of devices found*); if no devices*) were found, this parameter has the value 0; valid if MBX_RESULT = MBX_CMD_OUT_OF_RANGE

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.6.3 Reading the Device Names of a Remote Device

GetRemoteDeviceName, 0x82

When called, the name of a remote 750-644 I/O module or *Bluetooth*® SPP device in the wireless network is queried (see appendix “Reading the MAC ID of a Remote Device”). Since this information must be requested via remote device and no quick response can be guaranteed, the first request starts with the name resolution, but it responds with MBX_CMD_DENIED_BUSY without returning the name. Repeating the query delivers MBX_CMD_OK and the character chain of the queried device name as soon as the name has been determined. The call returns an error if “ScanRemoteDevices” has not been called beforehand and the scan (initial call) ended successfully. As long as the name query is running, the wireless module is not available for any other functions. Functions that do not use the wireless module are carried out normally.

When the name query ends, a new call of “GetRemoteDeviceName” returns the *Bluetooth*® name of the remote device (see appendix “Reading a Local Device Name”). A maximum of [Mailbox size -3] characters are displayed.

The *Bluetooth*® name of remote devices may also exceed the length (15 characters) that can be displayed in the largest Mailbox setting (18 bytes). In this case, the complete name can be read by block transfer (DLD_Start, DLD_Cont, DLD_End commands).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	(●)	(●)	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETREMOTEDEVICENAME							
1	T	-						
2	MBX_DEVICE_INDEX							

Arguments

Parameters	Value	Description
MBX_DEVICE_INDEX	[0 … 15]	List index for the return of the device name. The index must be smaller than the number of devices found in the scan process.

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETREMOTEDEVICENAME							
1	T	MBX_RESULT						
2	MBX_NAME_LENGTH							
3	CHAR1							
…	…							
17	CHAR15							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	No valid device*) was found in the scan list for the transferred index.
	MBX_CMD_DENIED_BUSY	The scan process has not yet been stated or concluded.
	MBX_CMD_TIMEOUT	The remote device*) has rejected the name query or has not responded within the time prescribed by the *Bluetooth*® standard.
MBX_NAME_LENGTH	[0 … 255]	Number of characters of the complete name.
CHARn	ASCII characters	Characters of the device name in ASCII code Example: ABC A = CHAR1 = 0x41 B = CHAR2 = 0x42 C = CHAR3 = 0x43

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.6.4 Entering a Device in the Table of Allowed Devices

AllowRemoteDevice, 0x83
This command is used to allow a remote 750-644 I/O module or *Bluetooth*® SPP device access to the local 750-644 I/O module. The MAC ID of the remote device is also entered in a table of the *Bluetooth*® subsystem. Two types of devices are differentiated. Both types are entered in different tables:

- 750-644 I/O modules for “Real-Time Communication” mode: WAGO_DEVICE (0x20 … 0x26)
- *Bluetooth*® SPP devices for “Ad-hoc Communication” mode*): EXTERNAL_DEVICE (0x10 … 0x15)
  *) *Bluetooth*® specification: Device supports the “Serial Port Profile” (SPP)

Note

Access permission
Before an entered 750-644 I/O module is actually authorized access, it must be enabled using the command “BindRemoteDevice”.
Access permission can be revoked using the command “UnbindRemoteDevice” without requiring deletion of the device from the table.

- Entries can be deleted from the table by overwriting with the MAC ID 00:00:00:00:00:00. The respective slot is populated with 0. No data is transmitted to it.
- Changes to the device blocks do not change anything in the process image mapping.
- A MAC ID (exception: MAC ID 00:00:00:00:00:00) may never occur more than once in a table.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
-	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_ALLOWREMOTEDEVICE							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							
3	MAC_ID_Byte 0 (LSB)							
4	MAC_ID_Byte 1							
5	MAC_ID_Byte 2							
6	MAC_ID_Byte 3							
7	MAC_ID_Byte 4							
8	MAC_ID_Byte 5 (MSB)							
9	RFCOMM_CHANNEL_ID							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)
MAC_ID_Byte_n	[0 … 255]	The bytes of the MAC ID to be entered
RFCOMM_CHANNEL_ID	0 … 63	This parameter only applies to ad-hoc slots. 0: Use the first available channel 1 … 63: Specific channel selection 1 … 63

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_ALLOWREMOTEDEVICE							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	Up to seven 750-644 I/O modules and 6 *Bluetooth*® SPP devices can be configured. This number has been exceeded.
	MBX_CMD_INVALID_ARG	The indicated MAC ID is already in the table or a false table ID has been given.

12.3.6.5 Reading a Remote Device from the Table of Allowed Devices

GetAllowedRemoteDevices, 0x84

This command is used to read and return the MAC ID of a remote device from the table of allowed devices of the *Bluetooth*® subsystem. There are two types of external devices that are entered in different tables:

- 750-644 I/O modules for “Real-Time Communication” mode: WAGO_DEVICE (0x20 … 0x26)
- *Bluetooth*® SPP devices for “Ad-hoc Communication” mode*): EXTERNAL_DEVICE (0x10 … 0x15)

*) *Bluetooth*® specification: Device supports the “Serial Port Profile” (SPP)

Note

Enable access permission

Before an entered device is actually authorized access, it must be enabled using the command “BindRemoteDevice”.
Access permission can be revoked again using the command “UnbindRemoteDevice” without making it necessary to delete the device from the table.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
-	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETALLOWEDREMOTEDEVICE							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETALLOWEDREMOTEDEVICE							
1	T	MBX_RESULT						
2	MAC_ID_Byte 0 (LSB)							
3	MAC_ID_Byte 1							
4	MAC_ID_Byte 2							
5	MAC_ID_Byte 3							
6	MAC_ID_Byte 4							
7	MAC_ID_Byte 5 (MSB)							
8	RFCOMM_CHANNEL_ID							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	An index greater than 6 was used.
	MBX_CMD_INVALID_ARG	No valid target table has been chosen.
MAC_ID_Byte n	[0 … 255]	The bytes of the MAC ID read back
RFCOMM _CHANNEL_ID	0 … 63	Value of the configured RFCOMM_CHANNEL_ID

12.3.6.6 Granting Access Permission to a Device

BindRemoteDevice, 0x85

A remote device from the table of allowed devices of the *Bluetooth*® subsystems is enabled for establishing a connection. The MAC ID of the remote device must have been entered in the table of allowed devices beforehand (see appendix “Entering a Device in the Table of Allowed Devices”).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_BINDREMOTEDEVICE							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_BINDREMOTEDEVICE							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	An index greater than 6 was used.
	MBX_CMD_INVALID_ARG	No valid target table has been chosen.
	MBX_CMD_GENERAL_ERROR	The selected entry does not contain a valid MAC ID.

12.3.6.7 Deleting Access Permission for a Device

UnbindRemoteDevice, 0x86

The access permission of a remote device is disabled. When this happens, the MAC ID entered in the table space and associated data such as the “UserFriendlyName” are retained, but no connection to the device is established and any already existing connection is interrupted. If the command is executed in “Communication” mode, this setting is temporary, i.e. at the next reboot, the connection is once again established. This offers the possibility of temporarily excluding fault remote devices from the network without changing the configuration. If no attempt is made to connect the device after reboot, the command must be called up in the “Configuration” mode.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	●*)	-

*) Setting is nonpermanent in "Communication" mode. In "Configuration" mode, the setting is saved during a warm boot similar to other settings.

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_UNBINDREMOTEDEVICE							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table "2" for WAGO_DEVICE (slots 1 … 7) "1" for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_UNBINDREMOTEDEVICE							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	An index greater than 6 was used.
	MBX_CMD_INVALID_ARG	No valid target table has been chosen.

12.3.6.8 Reading Access Permission for Remote Devices

GetBoundRemoteDevices, 0x87

This command is used to read back which of the remote devices entered in the table enabled access permission. Permission can be enabled using the command "BindRemoteDevice" and disabled using "UnbindRemoteDevice".

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETBOUNDREMOTEDEVICES							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETBOUNDREMOTEDEVICES							
1	T	MBX_RESULT						
2	MBX_BOUND_DEVICES_WAGO							
3	MBX_BOUND_DEVICES_EXTERN							

Return values

Parameters	Value	Description
MBX_BOUND _DEVICES_WAGO	0x00 (keine Busklemme 750-644 gebunden) 0x7F (alle Busklemmen 750-644 gebunden). Bit 7 bleibt immer 0	Bit assignment according to the device index in the table “WAGO devices” for “Real-Time Communication” mode.
MBX_BOUND _DEVICES_EXTERN	0x00 (kein *Bluetooth*® SPP-Gerät gebunden) 0x3F (alle *Bluetooth*® SPP-Geräte gebunden.) Bit 6 und 7 bleiben immer 0	Bit assignment according to the device index in the table “External devices” for “Ad-hoc Communication” mode.

12.3.6.9 Reading QoS Settings

GetConnectionQoS, 0x88

This command is used to read the settings of the “Quality-of-Service” (QoS) of a connection.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETCONNECTIONQOS							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETCONNECTIONQOS							
1	T	MBX_RESULT						
2	MBX_QOS_SETTINGS							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	No error occurred
	MBX_CMD_INVALID_ARG	Index too large or WAGO table not chosen
MBX_QOS _SETTINGS	MBX_BQM_NONE (0x01)	No QoS enabled (standard)
	MBX_BQM_BEST_EFFORT (0x02)	QoS is enabled in “Best Effort” mode
	MBX_BQM_GUARANTEED (0x03)	QoS is enabled in the “Guaranteed” mode

12.3.6.10 Setting QoS Settings

SetConnectionQoS, 0x89

This command is used to set the settings of the “Quality-of-Service” (QoS) of a connection. The settings only take effect when the 750-644 I/O module is switched to the role of master.

Note

Keep factory settings!

The master can connect to a maximum of 3 slaves QoS is enabled. QoS can only be set for 750-644 I/O modules. It improves latency by reducing deviations (outliers).

Because the *Bluetooth*® subsystem of the 750-644 I/O module has already been optimized for maximum performance, any effect on the timing in “Real-Time Communication” mode is marginal. Therefore, it is recommended that you keep the factory setting MBX_BQM_NONE.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETCONNECTIONQOS							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							
3	MBX_QOS_SETTINGS							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE_AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7)
MBX_QOS_SETTINGS	MBX_BQM_NONE (0x01)	No QoS activated (standard)
	MBX_BQM_BEST_EFFORT (0x02)	QoS is enabled in “Best Effort” mode
	MBX_BQM_GUARANTEED (0x03)	QoS is enabled in the “Guaranteed” mode

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETCONNECTIONQOS							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	An index greater than 6 was used.
	MBX_CMD_INVALID_ARG	No valid target table was used or an invalid value for MBX_QOS_SETTINGS was selected.

12.3.6.11 Reading Back the Time Setting – Between Two Attempts to Establish a Connection

GetReconnectionTimePeriod, 0x8A

This command is used to read back the time interval between two attempts to reestablish the connection to a 750-644 I/O module.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETRECONNECTIONTIMEPERIOD							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETRECONNECTIONTIMEPERIOD							
1	T	MBX_RESULT						
2	MBX_RECONNECTTIME (LSB)							
3	MBX_RECONNECTTIME (MSB)							

Return value

Parameters	Value	Description
MBX_RECONNECTTIME	[0 … 65535]	Minimum time interval in seconds between two attempts to reestablish a connection when the previous attempt has failed (value 0: no waiting between two attempts).

12.3.6.12 Setting the Time Setting – Between Two Attempts to Establish a Connection

SetReconnectionTimePeriod, 0x8B
This command is used to set the waiting time for the master between two attempts before a new attempt to establish a connection with a slave is undertaken. The settings only take effect when the 750-644 I/O module is switched to the role of master.

When establishing a network in “Communication” mode, the master first attempts to connect all allowed slaves. If unsuccessful, it begins the data exchange, first with the devices that could be connected. It then scans again for the devices originally not found at a configured time interval. A similar scenario applies for the failure of slaves; in this case, the master first attempts to reconnect immediately and repeats these attempts periodically if it does not succeed. 750-644 I/O modules configured as slaves continuously attempt to connect in “Communication” mode.

Note

Communication drops
When connecting to slaves, the master is not available for data exchange. If allowed slaves have failed continuously, the remaining network members experience communication interruption times within the time interval of the “ReconnectionTimePeriod” until the failed device is ready again. In “Real-Time Communication” mode, 750-644 I/O modules provide information on such interruption times through the cyclical and acyclical diagnostic function. For time-critical applications, it is possible to temporarily “eject” failed slaves by applying the function “UnbindRemoteDevice” to them in “Communication” mode.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETRECONNECTIONTIMEPERIOD							
1	T	-						
2	MBX_RECONNECTTIME (LSB)							
	MBX_RECONNECTTIME (MSB)							

Arguments

Parameters	Value	Description
MBX_RECONNECTTIME	Time in seconds	Minimum time in seconds between two attempts to reestablish a connection when the previous attempt has failed.

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETRECONNECTIONTIMEPERIOD							
1	T	MBX_RESULT						

Return value

Parameters	Value	Description
-	-	-

12.3.6.13 Reading the User-Friendly Name of an Allowed Device

GetUserfriendlyName, 0x8C

This query can be used to read the user-friendly name to an entry in the list of allowed devices. If the name is too long for the actual size of the Mailbox, then the first [Mailbox size -3] characters are output. The value from MBX_NAME_LENGTH returns the actual length of the name.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	(●)	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETUSERFRIENLYNAME							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE_AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICENAME							
1	T	MBX_RESULT						
2	MBX_NAME_LENGTH							
3	CHAR1							
…	…							
17	CHAR15							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	An index greater than 6 was used.
	MBX_CMD_INVALID_ARG	No valid target table has been chosen.
MBX_NAME_LENGTH	[0 … 15]	Number of characters of the complete name
CHARn	[0 … 255]	Characters of the device name in ASCII code Example: ABC A = CHAR1 = 0x41 B = CHAR2 = 0x42 C = CHAR3 = 0x43

12.3.6.14 Writing the User-Friendly Name of an Allowed Device

SetUserfriendlyName, 0x8D
This command can be used to add any alias to an entry to the list of allowed devices. The user can specify an intuitive name for the respective remote device, e.g. “Pump_001”, “Gate4” or “Busnode_002”. The name does not affect the *Bluetooth*® device name of the remote device because the alias is saved in the local 750-644 I/O module.
By converting to ASCII characters, the name is simple to read and facilitates administration of the *Bluetooth*® SPP device.

The name can be a maximum of [Mailbox size -3] characters long. If the name does not completely fill up the Mailbox, it ends with the first null byte.

Note

Alias has no direct relationship to the *Bluetooth*® device name!
The name entry is independent of the entered device (MAC ID). The user-friendly name has no direct relation to the *Bluetooth*® name of the remote device that can be read with “GetRemoteDeviceName”. Characters following a null byte are ignored.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	(●)	●	●	-	-	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETUSERFRIENLYNAME							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							
3	CHAR1							
…	…							
17	CHAR15							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE_AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)
MBX_NAME_LENGTH	[0 … 15]	Number of characters of the complete name
CHARn	[0 … 255]	Characters of the device name in ASCII code. 0x0 terminate the string. Example: ABC A = CHAR1 = 0x41 B = CHAR2 = 0x42 C = CHAR3 = 0x43 End of the name = CHAR4 = 0x00

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICENAME							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	An index > 6 (for 750-644 I/O modules) > 5 (for *Bluetooth*® SPP devices) was selected.
	MBX_CMD_INVALID_ARG	No valid target table has been chosen.

12.3.7 Diagnostics

12.3.7.1 Reading the Status of the Local I/O Module

GetLocalDeviceStatus, 0xD0
This call returns the type of 750-644 I/OI module, the operating mode and a general diagnostic status.

Note

Type and operating mode
The type of I/O module is set by using the command “SetLocalDeviceRole” (see appendix “Setting the Role of a Local Device”) or read back by using the command “GetLocalDeviceRole” (see appendix “Reading the Role of a Local Device”).
The operating mode is set using “SetLocalOperationMode” (see appendix “Setting the Local Operating Mode”) or read using “GetLocalOperationMode” (see appendix “Read the Local Operating Mode”).

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICESTATUS							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLOCALDEVICESTATUS							
1	T	MBX_RESULT						
2	MBX_DEVICE_ROLE							
3	MBX_OPMODE_ID							
4	MBX_COMMROFILE_ID							
5	MBX_DIAGNOSTIC_STATE							

Return values

Parameters	Value	Description
MBX_DEVICE_ROLE	MBX_ROLE_COORDINATOR (0x01)	Device role: Master
	reserved (0x02)	Reserved
	MBX_ROLE_ENDDEVICE (0x03)	Device role: Slave
MBX_OPMODE_ID	MBX_CM_OPMODE_CONF (0x01)	"Configuration" mode
	MBX_CM_OPMODE_COMM (0x02)	"Communication" mode
MBX_COMMROFILE_ID	MBX_CM_OPPROFILE_REALTIME (0x01)	"Real-Time Communication" mode
	MBX_CM_OPPROFILE_ADHOC (0x03)	"Ad-hoc Communication" mode
	MBX_CM_OPPROFILE_CONFIG (0x02)	"Configuration" mode
MBX_DIAGNOSTIC_STATE	OK (0x00)	No error, no warning
	MBX_WARNING (0x01)	Warning, details query necessary
	MBX_ERROR (0x02)	General error, details query necessary
	MBX_CRITICAL_ERROR (0x04)	Critical error, details query necessary

12.3.7.2 Reading the Status of the Wireless Network

GetNetworkStatus, 0xD1
Call returns information on the status of the wireless network. Information about 750-644 I/O modules and *Bluetooth*® SPP devices is recorded.

Note

In "Configuration" mode, no connections are established!
Since no radio connections are established when in "Configuration" mode, MBX_NETWORK_FAILED (0x01) is always returned. Since no connections are established in "Configuration" mode, the other arguments of this command in this mode always return "0".

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETNETWORKSTATUS							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETNETWORKSTATUS							
1	T	MBX_RESULT						
2	MBX_NETWORK_STATE							
3	W7	W6	W5	W4	W3	W2	W1	W0
4	E7	E6	E5	E4	E3	E2	E1	E0

Return values

Parameters	Value	Description
MBX_NETWORK_STATE	MBX_NETWORK_FAILED (0x01)	Configured network not established (e.g. in “Configuration” mode).
	MBX_NETWORK_OK (0x02)	Configured network successfully established.
	MBX_NETWORK_INCONSISTENT (0x03)	At least one, but not all configured connections are established.
W0 … W6	Assigned bit = 1	Assigned 750-644 I/O module from the table of WAGO devices is linked and connected.
	Assigned bit = 0	Assigned 750-644 I/O module from the table of WAGO devices is not connected.
W7	0	Reserved
E0 … E5	Assigned bit = 1	Assigned *Bluetooth*® SPP device from the table of external devices is linked and connected.
	Assigned bit = 0	Assigned *Bluetooth*® SPP device from the table of external devices is not connected.
E6, E7	0	Reserved

Note

Correspondences
W0 to W6 correspond to WAGO 750-644 I/O modules 0x20 to 0x26 with MBX_TARGET_TABLE_AND_INDEX.
E0 to E5 correspond to *Bluetooth*® SPP devices 0x10 to 0x16 with MBX_TARGET_TABLE_AND_INDEX.

12.3.7.3 Reading Diagnostic Information

GetStatusMessage, 0xD2
The command returns diagnostic information on occurring errors and warnings from the local 750-644 I/O module. When querying, a concrete object identification MBX_OBJECT_ID must be indicated. The response then always contains the same MBX_OBJECT_ID plus a defined status report MBX_STATE_MESSAGE. If the object identification remains unknown, the

system returns the information byte for executing the command MBX_CMD_RESULT and the value MBX_CMD_INVALID_ARG.

Each defined MBX_OBJECT_ID is always uniquely assigned a current status report (usually “OK”). If an event occurs, the status report is changed in such a way that it mirrors the most recently occurring event (error/warning). The status report of an individual MBX_OBJECT_ID is always overwritten with the next more recent event as long as it is not “OK”. The prioritization of error message before warning message must always be observed.

Note

Errors/warnings
In the cyclical status report (C/S byte, LED activation), errors/warnings are only displayed as long as the interrupted status lasts. The status report, on the other hand, remains until it is overwritten (new message for the same Object ID occurs). Errors always have a higher priority than warnings in the display. Only the status of 750-644 I/O modules is recorded.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	●	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETSTATUSMESSAGE							
1	T	-						
2	MBX_OBJECT_ID (LSB)							
3	MBX_OBJECT_ID_LO							

Arguments

Parameters	Value		Description
MBX_OBJECT_ID	**MBX_OBJECTID_GROUP_MASK**	**0xF000**	
	MBX_OBJECTID_GROUP_SYSTEM	0x0000	Status of entire system
	MBX_OBJECTID_GROUP_WIRELESS	0x1000	Status of wireless connections
	MBX_OBJECTID_GROUP_TIMING	0x2000	Status of time monitoring
	MBX_OBJECTID_GROUP_PA	0x3000	Status of process image
	MBX_OBJECTID_GROUP_ISC	0x4000	Status of intersystem communication
	MBX_OBJECTID_GROUP_CONFIG	0x5000	Status of configuration
	MBX_OBJECTID_TARGET_MASK	**0x0FFF**	
	Target-ID	0x000 through 0x0007	Target object in the group

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETSTATUSMESSAGE							
1	T	MBX_RESULT						
2	MBX_OBJECT_ID (LSB)							
3	MBX_OBJECT_ID (MSB)							
4	MBX_STATE_MESSAGE (LSB)							
5	MBX_STATE_MESSAGE (MSB)							

Return values

Parameters	Value		Description
MBX_CMD _RESULT	MBX_CMD_DENIED_NOT_IMPLEMENTED		Non-implemented MBX_OBJECT_ID
	MBX_CMD_DENIED_BUSY		Another MBX command actively being processed
	MBX_CMD_INVALID_ARG		Invalid object ID
MBX_OBJECT _ID	See section "Establishment of the Object ID".		
MBX_STATE _MESSAGE	MBX_STATE_OK	0x0000	No error
	MBX_STATE_OK_CONFIG_CHANGED	0x0001	Configuration changed, but not yet saved
	Error messages		
	MBX_STATE_ERROR_UNSPECIFIED	0x1000	Not specified
	MBX_STATE_ERROR_WATCHDOG	0x1001	Watchdog
	MBX_STATE_ERROR_CREATE_LINK	0x1002	Connection errors
	MBX_STATE_ERROR_LOST_LINK	0x1003	Connection interrupted
	MBX_STATE_ERROR_PASIZE_WRONG	0x1004	PI defective
	MBX_STATE_ERROR_SYSTEMBUS_JAM	0x1005	Internal communication overallocated
	MBX_STATE_ERROR_SYSTEMBUS _INTERRUPTED	0x1006	Interruption in internal communication
	MBX_STATE_ERROR_MAILBOX _COMMAND	0x1007	Error in Mailbox communication
	MBX_STATE_ERROR_NETWORK _CONFIG	0x1008	Error in the network configuration
	Warnings		
	MBX_STATE_WARNING_UNSPECIFIED	0x2000	Not specified
	MBX_STATE_WARNING_WATCHDOG	0x2001	Watchdog
	MBX_STATE_WARNING_LESSTHEN_54 _CHANNELS	0x2002	Less than 54 channels available
	MBX_STATE_WARNING_LESSTHEN_39 _CHANNELS	0x2003	Less than 39 channels available
	MBX_STATE_WARNING_BER_MEDIUM	0x2004	BER is moderate
	MBX_STATE_WARNING_BER_HIGH	0x2005	BER is high
	MBX_STATE_WARNING_REMOTE _MAILBOX	0x2006	A remote Mailbox is active; data in the PI may be obsolete.

12.3.7.4 Establishment of the Object ID

The object ID is composed of a group ID and a target ID.

- The group ID identifies the functional group for which the status is to be queried.
- The target ID indicates the target for which the status is to be queried. Either all existing connections (0x0000) or individual connections (0x0001 to 0x0007) can be chosen.

A maximum of one connection exists in one slave; therefore, only the target IDs 0x0000 and 0x0001 are valid in this case as well.

For a master, the target IDs 0x0000 and 0x0007 are valid.

If the maximum possible 7 devices have been configured, the query of a target ID for which no device has been configured returns the value MBX_STATE_ERROR_UNSPECIFIED (0x1000).

If a connection has been configured but could not be established, the query of the corresponding target ID always returns the value MBX_STATE_ERROR_CREATE_LINK (ox1002).

To calculate the group ID and target ID from an existing object ID and vice versa, the following logical links must be used:

- Group_ID = Object_ID ∧ 0xF000
- Target_ID = Object_ID ∧ 0x0FFF
- Object_ID = Group_ID ∨ Target_ID

Group ID	**Target ID**	**Description**
0x0000	0x0000	The query of the group status always returns MBX_STATE_OK. An error in the overall system would mean failure of the module.
Wireless (status of the wireless connections), 0x1000	0x0000	If not all devices*) are connected or if the 750-644 I/O module is in “Configuration” mode, MBX_STATUS_ERROR_UNSPECIFIED is returned as the group status, otherwise MBX_STATE_OK.
	Slave: 0x0001 Master: 0x0001 to 0x0007	If no device*) is linked for the corresponding table space, MBX_STATE_OK is always returned. For existing connections, the following warnings are generated for connection quality: - MBX_STATE_WARNING_BER_HIGH - MBX_STATE_WARNING_BER_MEDIUM - MBX_STATE_WARNING_LESSTHEN_39_CHANNELS - MBX_STATE_WARNING_LESSTHEN_54_CHANNELS A master can also return information about the connection status: - MBX_STATE_OK If the respective slave is connected or is connecting. - MBX_STATE_ERROR_CREATE_LINK If the slave could not connect, but further attempts to connect are carried out (device*) is configured, but not accessible).
Timing 0x2000	0x0000 and Slave: 0x0001 Master: 0x0001 to 0x0007	For connected 750-644 I/O modules: - MBX_STATE_ERROR_WATCHDOG The time since the last packet is greater than 60 x (number of active radio channels+ 2 ms). - MBX_STATE_WARNING_WATCHDOG The time since the last packet is greater than 20 x (number of active radio channels + 2 ms). - MBX_STATE_OK The time since the last packet is less than 20 x (number of active radio channels + 2 ms). For connected *Bluetooth*® SPP devices: - Always MBX_STATE_OK

Group ID	Target ID	Description
Process image 0x3000	0x0000	- MBX_STATE_ERROR_PASIZE_WRONG The sum of all set cut-offs is greater than the PI size -2. - MBX_STATE_WARNING_REMOTE_MAILBOX The Mailbox is active for least one remote device*).
	0x0001 to 0x0007	- MBX_STATE_ERROR_PASIZE_WRONG The set cut-off is greater than the PI size -2. - MBX_STATE_WARNING_REMOTE_MAILBOX The Mailbox is active for the remote device*), the data in the PI could be obsolete.
Intersystem communication 0x4000	-	Reserved
Configuration 0x5000	0x0000	The group status returns MBX_STATE_OK_CONFIG_CHANGED if the configuration has changed; otherwise, MBX_STATE_OK.

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.7.5 Reading the Connection Quality

GetLinkQuality, 0xD5

The connection quality (“Link Quality” LQ) is indicated by the bit error rate of the radio connection. The LQ value is converted into the current bit error rate based on the following characteristic:

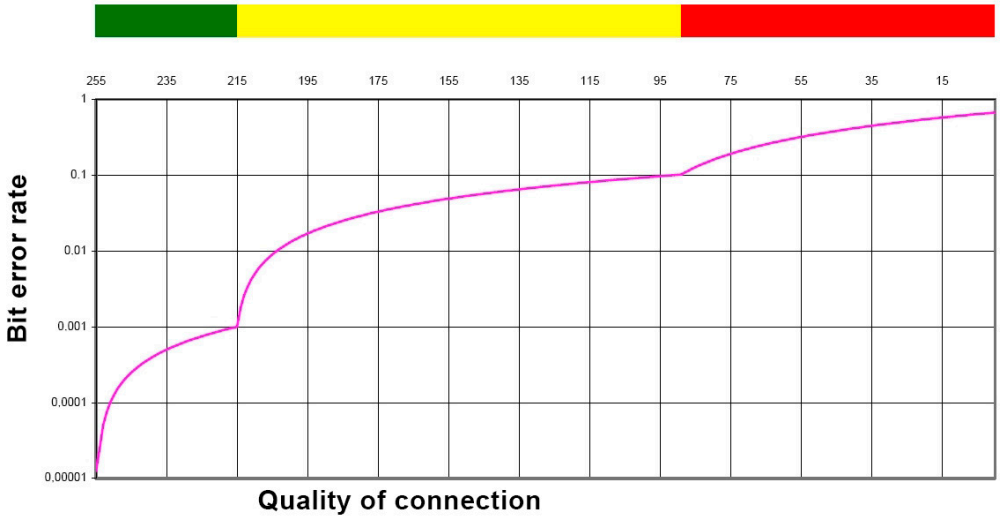


Figure 61: LED indication and connection between connection quality and bit error rate

The connection quality (see bars over the table) is indicated by LEDs:

- Green: Indicates a low bit error rate of $< 10^{-3}$
- Yellow: Indicates a bit error rate ranging from 10^{-2} to 10^{-3}
- Red: Indicates a bad transmission channel with a bit error rate of $< 10^{-2}$

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	-	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLINKQUALITY							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE_AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLINKQUALITY							
1	T	MBX_RESULT						
2	MBX_LQ_VALUE							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	The index used is too large.
	MBX_CMD_GENERAL_ERROR	The device*) is not connected.
	MBX_CMD_INVALID_ARG	The device*) is not connected or no valid target table was selected.
MBX_LQ_VALUE	[0 … 255]	Value of the connection quality for the requested connection.

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.7.6 Reading the Signal Strength for a Connection

GetLinkSignalStrength, 0xD7

The RSSI value displays possible overrides of the *Bluetooth*® receiver. It returns “0” if the strength of the received signal lies within the tolerance range. If the received signal is stronger than the upper limiting value of the tolerance range, a value > “0” is returned. If the received signal is weaker than the lower limiting value, a value < “0” is returned.

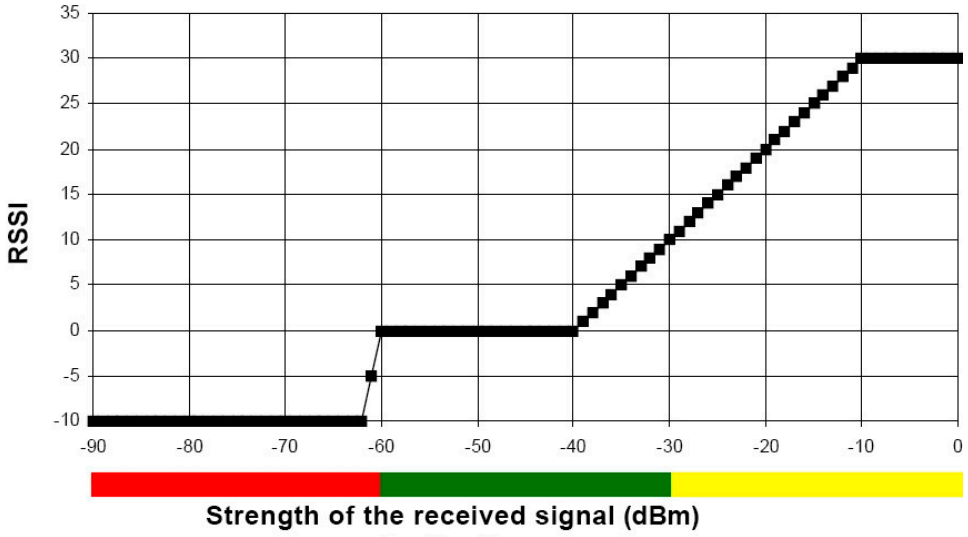


Figure 62: Connection between RSSI value and LED color

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	-	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLINKSIGNALSTRENGTH							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETLINKSIGNALSTRENGTH							
1	T	MBX_RESULT						
2	MBX_RSSI_VALUE							

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	The index used is too large.
	MBX_CMD_GENERAL_ERROR	The device*) is not connected.
	MBX_CMD_INVALID_ARG	The device*) is not connected or no valid target table was selected.
MBX_RSSI_VALUE	-128 … +127 (two´s complement)	RSSI value for the requested connection.

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.7.7 Reading Available Hopping Channels

GetAvailableChannelMap, 0xD8

The call returns information on the status of the environment (i.e. the status of the wireless medium) for a connection channel. For *Bluetooth*® the channels available for hopping are indicated. There are 79 channels with 1 MHz available. The channels are numbered serially from 0 through 78. The frequency of each channel is based on the channel number:

Frequency of the channel = 2402 + channel number MHz

The 750-644 I/O module supports AFH (adaptive frequency hopping). If individual frequency ranges are recognized as defective (for example, if other wireless technologies with higher signal strength in this range are sending), the corresponding channels of its own transmission are excluded. This reduces interference and improves the connection quality for the *Bluetooth*® network as well as for the third-party system. A positive side effect is the possibility of making connections through third-party activity in the 2.4 GHz ISM band using the list of the channels masked in this manner. The rule of thumb is: The greater the number of channels available for hopping, the better the status of the wireless medium.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
-	●	●	-	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETAVAILABLECHANNELMAP							
1	T	-						
2	MBX_TARGET_TABLE_AND_INDEX							

Arguments

Parameters	Value	Description
MBX_TARGET_TABLE _AND_INDEX	Bit 0 … 3	Table index Index 0 … 6 for 750-644 I/O modules of slots 1 … 7 Index 0 … 5 for *Bluetooth*® SPP devices of slots 8 … 12
	Bit 4 … 7	Target table “2” for WAGO_DEVICE (slots 1 … 7) “1” for EXTERNAL_DEVICE (slots 8 … 12)

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_GETAVAILABLECHANNELMAP							
1	T	MBX_RESULT						
2	MBX_AFH_CHANNEL_MAP (LSB)							
3	MBX_AFH_CHANNEL_MAP							
4	MBX_AFH_CHANNEL_MAP							
5	MBX_AFH_CHANNEL_MAP							
6	MBX_AFH_CHANNEL_MAP							
7	MBX_AFH_CHANNEL_MAP							
8	MBX_AFH_CHANNEL_MAP							
9	MBX_AFH_CHANNEL_MAP							
10	MBX_AFH_CHANNEL_MAP (MSB)							

Return value

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OUT_OF_RANGE	The index used is too large.
	MBX_CMD_GENERAL_ERROR	The device*) is not connected.
	MBX_CMD_INVALID_ARG	The device*) is not connected or no valid target table was selected.
MBX_AFH _CHANNEL _MAP	Each *Bluetooth*® channel is represented by one bit: Bit x = 0: Channel x is not available for hopping (because otherwise busy); Bit x = 1: Channel can be used for channel hopping for the requested connection. Channel numbers correspond to the significance of the bits: Bit 0 (bit with the lowest significance) in the LSB = channel 0 (2402 MHz) Bit 1 in the LSB = channel 1 (2403 MHz) … Bit 6 in the MSB = channel 78 (2480 MHz) Bit 7 (bit with the highest significance) in the MSB = channel 79 (always 0; does not exist)	

*)750-644 I/O module or *Bluetooth*® SPP device

12.3.7.8 Setting an LED

SetLED, 0xD9

Call sets the color and blink code of a defined LED. This can be used to test the functionality of the LED.

Note

750-644 I/O module reboot required!

To reinstate normal status information on the LEDs, the 750-644 I/O module must be rebooted. This can be triggered by the corresponding Mailbox command or by briefly switching off the power.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	-	-	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLED							
1	T	-						
2	MBX_LED_NUMBER							
3	MBX_LED_COLOR							
4	MBX_LED_BLINK							

Arguments

Parameters	Value	Description
MBX_LED_NUMBER	[0 … 7]	Selection of the LED, top left LED 0, to the right of that LED 1, etc.
MBX_LED_COLOR	MBX_LEDOFF (0x00)	LED off
	MBX_LEDRED (0x01)	LED color red
	MBX_LEDGREEN (0x02)	LED color green
	MBX_LEDYELLOW (0x03)	LED color yellow
MBX_LED_BLINK	MBX_LEDSTATIC (0x00)	LED statically switched on
	MBX_LEDBLINK (0x01)	LED blinks

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_SETLED							
1	T	MBX_RESULT						

Return values

Parameters	Value	Description
MBX_RESULT	MBX_CMD_OK	No error occurred
	MBX_CMD_DENIED_NOT_APPLICABLE	Not available in “Real-Time Communication” or “Ad-hoc Communication” mode.
	MBX_CMD_OUT_OF_RANGE	An invalid LED number was given.
	MBX_CMD_INVALID_ARG	An invalid color or behavior was indicated.

12.3.7.9 Mirroring a Mailbox for Test Purposes

MirrorMailboxCommand, 0xDA
This command causes the 750-644 I/O module to immediately copy the contents of the Mailbox query to the full extent of the Mailbox in the contents of the response. The command can be executed to test acyclic communication between the application and the local 750-644 I/O module.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
●	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_MIRRORMAILBOXCOMMAND							
1	T	-						
2	MBX_CONTENT_1							
…	…							
17	MBX_CONTENT_16							

Arguments

Parameters	Value	Description
MBX_CONTENT_n	Any payload	The number (n) of bytes is limited by the current Mailbox size -2.

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_MIRRORMAILBOXCOMMAND							
1	T	MBX_RESULT						
2	MBX_CONTENT_0							
…	…							
17	MBX_CONTENT_15							

Return value

Parameters	Value	Description
MBX_CONTENT_n	The value for MBX_CONTENT_n transmitted in the query	The number (n) of bytes is limited by the current Mailbox size -2.

12.3.7.10 Reading the Operating Time of the I/O Module

GetLocalUpTime, 0xDB
Call can read the operating time of the 750-644 I/O module since the last reboot.

Note

Not designed for time measurement over longer periods of time!
This function serves as an aid for troubleshooting, e.g. to test power failures. The accuracy of the time measurement is not designed to enable precise time measurement over longer periods of time.

Conditions

Mailbox size			Mode			Device role		Save config.	Reboot
6	12	18	Config.	Real-time	Ad-hoc	Master	Slave		
(●)	●	●	●	●	●	●	●	-	-

Request

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_UPTIME							
1	T	-						

Response

Byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	MBX_UPTIME							
1	T	MBX_RESULT						
2	MBX_MINUTES							
3	MBX_HOURS							
4	MBX_DAYS (LSB)							
5	MBX_DAYS							
6	optional	MBX_DAYS						
7	optional	MBX_DAYS (MSB)						

Return values

Parameters	Value	Description
MBX_MINUTES	[0 … 59]	Minute portion of the operating time
MBX_HOURS	[0 … 24]	Hour portion of the operating time
MBX_DAYS	Mailbox size 6: [0 … 65,535] Mailbox size > 6: [0 … 4294967295]	Number of days the 750-644 I/O module has been operating. The two higher value bytes are only available with a Mailbox > 6.

12.4 Extended Register Structure (Configuration Block)

The extended register structure represents the entirety of all configuration parameters. It is created as 512 byte block from 128 registers per each 4 bytes and can be read and written by means of block transfer (DLD_Start, DLD_Cont, DLD_End commands). If the structure is read from the 750-644 I/O module, the fields marked as “reserved” are populated with the value “0”. If the structure is written to the 750-644 I/O module, the 750-644 I/O module ignores the values entered in the fields marked as "reserved".

In addition, some fields have parameters that cannot be written (e.g. local MAC address). These fields are also not accepted by the 750-644 I/O module during a write operation. If the 750-644 I/O module should be configured when writing the extended register structure, we recommend the following procedure:

- Reading the structure from the 750-644 I/O module
- Making the desired changes in the respective fields
- Writing the modified structure to the 750-644 I/O module

Offset (byte)	Register No.	Length (bytes)	Data Type Definition	Description
0	0 … 2	9	Device status: Byte 1 local status Byte 2 status of master, slave 0 Byte 3 status of slave 1,2 Byte 4 status of slave 3,4 Byte 5 status of slave 5,6 Byte 6 status of remote device*) 0,1 Byte 7 status of remote device*) 2,3 Byte 8 status of remote device*) 4,5 Byte 9 status of remote device*) 6,7	Local status: 0 if all configured, but at least one 750-644 I/O module is connected. Otherwise always 1. Status of master, slaves or external devices*), 4 bits per device: 0 – not connected 1 – connection established 2 – connection 3 – device*) has been “parked”
		1	Version of main configuration	Version of the configuration (see appendix “Reading the Host Firmware Version”)
		2	Version of the subconfiguration	
12	3	4	Configuration key	Identification of the configuration must have the value 0x1E55F15E.
16	4	1	Process image size	Size of the I/O module process image in bytes.
		1	Size and type of the Mailbox interface	Bit 0 … 14: Size of the Mailbox. Bit 15: Must have value “1” (Mailbox type can be “faded in”)
		1	*Reserved*	
		1	Wireless technology	1 – *Bluetooth*®
20	5 … 7	6	Local MAC address	Local address of the 750-644 I/O module, LSB first (see appendix “Reading a Local MAC ID”).
		1	Profiles supported	Bit field for the supported profiles; the individual values are linked binarily using OR: 1 – Proprietary real-time profile 2 – SPP
		1	Local device role	See appendix “Reading the Role of the Local Device” and “Setting the Role of the Local Device”.
		1	ID of the communication profile	See appendix “Reading the Local Operating Mode” and “Setting the Local Operating Mode”.
		3	*Reserved*	
32	8	1	WAGO device class	See appendix “Reading the Local WAGO Device Class” and “Writing the Local WAGO Device Class”.
		1	WAGO device subclass	See appendix “Reading the Local WAGO Device Class” and “Writing the Local WAGO Device Class”.
		1	Encryption mode	See appendix “Reading the Local Encryption Mode” and “Setting the Local Encryption Mode”.
		1	Use “Link Key”	0 – No authentication or PIN 1 – Authentication with “Link Key”. See appendix “Reading the Local Authentication Mode” and “Setting the Local Authentication Mode”.
36	9 … 12	16	Local device name	See appendix “Reading Local Device Names” and “Writing Local Device Names”.
52	13, 14	8	*Reserved*	

Offset (byte)	Register No.	Length (bytes)	Data Type Definition	Description
60	15 … 18	16	Password	See appendix "Reading the Local *Bluetooth*® Password" and "Writing the Local *Bluetooth*® Password".
76	19	1	Inquiry time	Maximum duration of a query. The exact time results from: Inquiry time x 1.28 sec.
		1	*Reserved*	
		2	Reconnection time	Time between two connection attempts, LSB first (see appendix "Reading Back Time Setting – Between Two Attempts to Establish a Connection" and "Setting the Time Setting – Between Two Attempts to Establish a Connection").
80		12	*Reserved*	
92	22 … 23	1	MaxTXPower	Restriction of the maximum transmission power, value in dBm, value range [0 … 20]
		1	DefaultTXPower	Transmission power used for paging and inquiry, value in dBm, value range [0 … 20]
		6	*Reserved*	
100	24 … 114 13*7 Register	364 13*28 Reg.	Slave configuration for 13 devices*), 28 bytes per device*): Byte 1 … 6: MAC ID Byte 7: Bind/Unbind Byte 8: Max. process data length Byte 9: PI size of the remote device*) Byte 10: RFCOMM_CHANNEL Byte 11: *Reserved* Byte 12: Supported profiles Byte 13 … 28: UserFriendlyName	Configuration for 13 remote devices*): Local address of the 750-644 I/O module, LSB first (see appendix "Reading the MAC ID of a Remote Device") 1 – Device*) was connected 0 – other (see appendix "Granting Access Permission to a Device" and "Deleting Access Permission for a Device") Configured width of the slots in the PI (size of the available process data) PI of the remote device*) (see appendix "Determining the Size of a Slot for Data Transfer in the Master PI") RFCOMM channel to which the connection is established Received value for the bit field of the supported profiles. The individual values are linked binarily using OR: 1 – Real-time 2 – SPP User-friendly name (see appendix "Reading the User-Friendly Name of an Allowed Device").
464	115 … 128	48	*Reserved*	

*)750-644 I/O module or *Bluetooth*® SPP device

13 Glossary

A

Ad-hoc
(lat. for “to this”)
An ad-hoc network refers to a wireless network topology in information technology between two or more end devices that manage without a fixed infrastructure.
For this I/O module: “Ad-hoc Communication” mode is one of 3 operating modes that the 750-644 I/O module supports. In “Ad-hoc Communication” mode, communication is also possible with *Bluetooth*® SPP devices from other manufacturers.

AFH
(Adaptive Frequency Hopping)
The adaptive frequency hopping process is used to temporarily skip faulted or occupied sections of the total available frequency band and use other channels.

Allowed device
Device with which a connection can be established. Devices not included in this category cannot connect.

Application
An application is a computer program developed by the user.

ASCII
(American Standard Code for Information Interchange)
The basis for character set coding. Character encoding defines 128 characters, consisting of 33 non-printable and 95 printable characters. Each character is assigned a bit pattern consisting of 7 bits. Since each bit can have two values, a total of 27 (= 128) different bit patterns exist, which can be interpreted as whole numbers 0–127 (hexadecimal 00–7F).

Authentication
Authentication is a process for testing the identity transmitted by a communication partner.

Available device
Device, that is physically in range and ready to connect and ready for use.

B

Baseband
A baseband is referred to if the desired signal is transmitted in an unaltered frequency range. In wireless communication systems, wireless transmission does not usually take place in the baseband, but rather by modulation of a significantly higher frequency carrier signal with the baseband signal.

Binding
Bluetooth® specific process in which a connection between devices is enabled to exchange data.

Bit
A bit is the smallest unit of information. Its value can either be 1 or 0.

Bit error rate
Generally: Frequency of bit errors in the data transmission.
Bluetooth® context: Information in percentage on recognized bit errors during baseband transmissions. As a rule, packets recognized as defective can be automatically repaired. If this is not possible, the defective data is automatically discarded.

Bit rate
Number of bits transmitted within a time unit.

Block
For the purpose of this I/O module: A block is a large area of data that belongs together in which all configuration data is stored and can be accessed.

Block transfer
Configuration data of a block can be downloaded or uploaded using a block transfer.

***Bluetooth*®**
Industry standard according to IEEE 802.15.1 for data radio transmission between devices over short distances. This includes non-connected and connected point-to-point transmission and so-called piconets. *Bluetooth*® provides an interface through which both mobile miniature devices such as mobile phones and PDAs and computers and peripheral devices can communicate with one another. The main purpose behind *Bluetooth*® is to do away with the need for wired connections between devices.

***Bluetooth*® SPP device**
Bluetooth® SPP denote *Bluetooth*® devices from other manufacturers that support the “Serial Port Profile” (SPP).

Bootloader
The bootloader is software in the first block of the bootable system. It is loaded and executed by the firmware and, in turn, starts other parts of the operating system. The version status of the bootloader can be separately queried.

Byte
(Binary-Yoked-Transfer-Element)
A data element larger than a bit and smaller than a word. A byte generally contains 8 bits. A byte may contain 9 bits in 36-bit computers.

C

Channel
A channel is a mechanism or resource that enables data transmission over space or time.

Channel map
The channel map defines for a quantity of available channels the portion thereof that is to be used or not used. This makes it possible to reserve certain frequencies for transmission by other devices or technologies.

Checksum
The formation of a checksum is used to recognize errors in data transmission or storage. There are different methods of forming a checksum. Usually, redundant bits calculated from the report itself are attached to a report. These bits are calculated again after data transmission and compared to the checksum.

CoD
(Class of Device)
The *Bluetooth*® Class of Device (CoD) is a 24-bit field that indicates to what standard device type (such as mobile phone or hands-free calling kit) a *Bluetooth*® device corresponds. Vendor-specific types may also be used in addition to standard types.
The complete CoD for the WAGO *Bluetooth*® RF Transceiver 750-644 is 000000000010000011111000bin or 0x0020F8hex.

Communication
For this I/O module: In “Communication” mode of the 750-644 I/O module, data is exchanged cyclically with connected *Bluetooth*® devices.

Configuration
Operating mode of the 750-644 I/O module, in which the I/O module can be configured. In this mode, no data is exchanged with other devices. However, *Bluetooth*® devices within wireless range can be scanned.

Connection
Presence of at least one transmission channel between devices that communicate with each other.

Control byte
For this I/O module: A specific byte of the cyclic process image output (PIO) containing protocol information for acyclic services (register communication, parameter channels).

C/S byte
(**C**ontrol/**S**tatus **byte**)
See “Control byte” and “Status byte”

Cut-off
Generally: Limitation to a specific size of the portion of cyclical process data allocated to a device.
Bluetooth® context: A *Bluetooth*® master only considers segments of the process image of connected *Bluetooth*® slaves. The *size of these segments* is referred to as the cut-off. Such a cut-off is set using WAGO-I/O-*CHECK* in “Configuration” mode of the *Bluetooth*® master. The current data to be read and written, which is assigned to slots in the *Bluetooth*® master's process image, remains.

Command
Instruction to perform specific actions

D

Data exchange
Transfer of data between communication partners

Device name
The *Bluetooth*® name of a device. This name can be queried by other *Bluetooth*® devices via a radio link.

Device role
Bluetooth® context: Difference between the function as master or slave.

dBi (isotrop)
Unit for antenna gain with respect to the isotropic radiator

DC/DC
“Direct Current” (DC) is the English name for continuous current. A direct current power controller (DC/DC transformer) is a self-controlled converter that periodically switches to generate a different voltage at the output. An area of application, for example, is electrical drive technology.

Diagnostics
Diagnostic information provides information on the system status, particularly disturbances or error conditions. Cyclic diagnostic information is provided by the LED displays and the content of the status bytes. Acyclic diagnostic information can be queried using the mailbox interface.

E

EDR
(**E**nhanced **D**ata **R**ate)
EDR characterizes newer *Bluetooth*® versions that allow data transmission rates of several Mbit/s.

EIRP
(**E**quivalent **i**sotropically **r**adiated **p**ower)
Product of the radiated power passed to a transmitting antenna multiplied by the antenna gain.

Encryption
Encryption converts sensitive data to illegible/unusable data by using a key. The raw data can only be obtained from encrypted data if the key is known.

F

FCC
(**F**ederal **C**ommunications **C**ommission)
Approval authority for communications devices in the USA

Firmware
For this I/O module: Software of the microcontroller used. The following versions can be queried separately: host controller and baseband controller.

Flag
An indicator for identifying certain conditions. A flag is represented by one bit. A certain status is represented by a certain bit value.

Fresnel zone
Predictable spatial regions between the transmitting and receiving antenna for a radio transmission.

Function block
Function blocks are used for IEC-61131 programming and stored in libraries for repeated utilization. A function block is a structured module, which has a name and contains input and output variables, but also local variables.

H

HCI
(**H**ost **C**ontroller **I**nterface)
An interface in the *Bluetooth*® protocol suite through which higher layers can directly act on the baseband protocol.

Hexadecimal
In a hexadecimal numbering system, the numbers are represented in a base 16 place value system.

Host controller
Host controllers are microcontrollers with different software statuses that can be queried.

I

IEEE
Institute of Electrical and Electronic Engineers

Ingress Protection acc. to IEC 60529
Ingress Protection classes (IP classes) define, on the one hand, the measures for protection against hazardous voltages (touch protection) and, on the other, protection measures against moisture.

Inquiry
An “Inquiry” (request/information), in *Bluetooth*® technology, is a process in which *Bluetooth*® devices within range are sought.

Internal data bus
With WAGO, an internal data bus is the internal bus of the WAGO-I/O-SYSTEM 750/753.

ISM
(**I**ndustrial, **S**cientific, and **M**edical Band)
ISM bands are frequency bands that can be used license-free with the observation of certain criteria. In addition to *Bluetooth*® other wide-spread wireless technologies such as WLAN use the ISM band at 2.45 GHz according to IEEE 802.11.

L

Latency
The latency of the data transmission indicates how long after transmission a data packet to a local interface that same data packet is available to a remote interface.

LED
(**L**ight-**E**mitting **D**iode)

Library
Collection of modules available to the programmer in the WAGO-I/O-*PRO* CAA programming tool for creating control programs in accordance with IEC 61131-3.

Link Key
Connection key issued using device information and (optionally) a PIN, which allows a secure Authentication of other devices.

Local I/O module
For this I/O module: A 750-644 I/O module that can be reached through a local interface (for example, linked by wire through a fieldbus).
Example: Configuration via WAGO-I/O-*CHECK*.

M

MAC ID
(**M**edia **A**ccess **C**ontrol **Id**entification)
The MAC ID of a device is its hardware address. *Bluetooth*® MAC IDs enable worldwide, unique identification of a specific *Bluetooth*® device.

Mailbox
I/O modules with Mailbox functionality have an acyclic communication channel (Mailbox) in the process image. The data exchange between I/O module and application can be significantly expanded over this channel without enlarging the process image. Depending on the module function, the remaining cyclic data is valid and available during mailbox communication.

Mailbox interface
The Mailbox interface is an interface for executing acyclic services using the process image (PI).

Master
In a device network, the master performs administrative tasks. For this I/O module: The 750-644 I/O module that is assigned the role of master, organizes the piconet and connections of up to 7 slaves.

Mirroring
For this I/O module: Received data is returned without change, permitting a simple function test of the interface.

N

Net Forming
Generally: “Net forming” is the configuration or establishment of a network. For this I/O module: All steps including the device configuration that are necessary for the successful establishment of connections between devices.

Node
A node is a contiguous structure of one or more input/output units that can be addressed via a local header-end (e.g. a WAGO fieldbus coupler/controller) as a network.

O

Offset
For this I/O module: Offset in the process image beginning with the 3rd byte D0 (after the control/status byte and internal byte).

Opcode
For this I/O module: “Opcode” is the abbreviated form of “operation code”. The opcode is part of a Mailbox command (1 byte in length). The complete command is formed by the opcode along with its arguments.

P

PI
(**P**rocess **I**mage)
The process image (PI) is an area of the memory in which the process data for and from modules/couplers/controllers is stored. The allocation and meaning of the process data are module-/fieldbus-specific.

PIO
(**P**rocess **I**mage of the **O**utputs/Output data)
Output process image

PII
(**P**rocess **I**mage of the **I**nputs/Inputs Data)
Input process image

Packet
For this I/O module: A data/wireless packet consists of user data and header data that are transmitted together.

Parameter channel
A parameter channel is an interface for parameterization of an I/O module. It is an acyclic communication channel between the application and I/O module with 2 bytes of protocol information and 2 bytes of data (255 addressable data sets).

Parity
A parity bit is a simple check bit in a data word. The parity check is employed to detect erroneously transmitted data words. Here, a data word refers to a sequence of bits. “Parity” denotes the number of bits assigned as 1 in the data word, and is designated “even” when the number of bits is even; otherwise “odd”.

Password
General: Data is protected from unauthorized users by a password. If the password is known, the rights secured by it are guaranteed. If the password is the sole means of securing against trespassers, special measures should be taken to keep it secret.
Bluetooth® context: The password is a character string that can be determined by the user for protection from unauthorized access. *Bluetooth*® SPP devices use a password to calculate “link keys” with additional information that forms the basis for authentication and encryption.

Piconet
A *Bluetooth*® network consisting of a master and up to 7 slaves is called a piconet. Communication may run directly and bi-directionally between master and slaves. However, communication between slaves is only possible indirectly through the master.

PIN
(**P**ersonal **I**dentification **N**umber)
The PIN is used to authenticate *Bluetooth*® devices with one another, acting as the basis for encryption.
See also "Password"

PLC
(**P**rogrammable **L**ogic **C**ontroller)
A PLC is a device used to control a machine or system and is programmed on a digital basis, such as a WAGO fieldbus controller.

Port
A port is an internal or external interface.

Process image mapping
Subdivision of a process image into independent parts and allocation of these parts to specific slots for data transmission.

Process Data
Process data lays within certain (established) areas of the memory and can be sent or received, for example, from the physical process of a control. The entirety of the process data forms the process image on the control level.

Q

QoS
(**Q**uality-**o**f-**S**ervice)
Quality of a communication service from the view of the user. The user can define his or her requirements regarding the communication service through the QoS.

R

Real-time capability
Devices have real-time capability if their time behavior is deterministic; i.e., the observations of guaranteed maximum times in each operating condition and the notification of timeouts as errors. For example, a device has real-time capability with regard to data exchange if a maximum delay for the transmission of data packets is never exceeded and errors or disturbances that occur are reported to the next higher entity.

Real-time
For this I/O module: “Real-Time Communication” mode is one of three operating modes that the 750-644 I/O module supports. It is especially suited for time-critical applications.

Reconnection time
The “reconnection time” is the time interval in which a non-connected device attempts to establish connections to other devices.

Register communication
Via register communication, an acyclic interface to the parameterization and configuration data of an I/O module with 1 byte of protocol information and 2 bytes of data (64 addressable data sets) is configured. In register communication, process data is not exchanged and the mailbox is masked.

Remote device
A device that is not connected through a local interface (e.g. fieldbus coupler/ controller), rather the connection is wireless only. The physical distance can range from centimeters to kilometers.

Return value
A return value is returned after the execution of a function or a confirmed service. It contains, for example, a performance result.

RSSI
(**R**eceived **S**ignal **S**trength **I**ndication)
The RSSI is an algorithm for determining signal strength between wireless participants. RSSI values allow, for example, the diagnosis of distances between wirelessly connected devices that are too small or too large. RSSI values are measured over a certain time span and can be derived from an existing communication. They range from 0 to 106.

R&TTE directive
(**R**adio and **T**elecommunications **T**erminal **E**quipment Directive)
This directive applies to radio equipment and for all devices that can be connected to public telecommunications networks.

S

Scan
See „Inquiry“

Scatternet
Group of independent and asynchronous piconets that contains at least one *Bluetooth*® device in common.

SDP
(**S**ervice **D**iscovery **P**rofile)
The SDP is used for identification of existing profiles.

Segment
For this I/O module: Part of the process image (PI). The process image is divided into up to 10 different segments. In addition to segments of a fixed size, e.g. C/S byte, HS byte and null byte, there are also segments of configurable size: the slots. The size of a segment is referred to as cut-off.

Signal strength
The signal strength is an indicator of reception quality. The higher the signal strength, the better the reception.

Slave
A slave is a device that transports data in a *Bluetooth*® device network (piconet). It does not accept any administrative tasks. Opposite: Master.

Slot
For this I/O module: A slot represents a part of the process image (PI) that is reserved for data exchange with a specific remote device.

SMA
(**S**ub-**M**iniature-**A**)
SMA refers to a special design for coaxial connectors. SMAs are used, for example, to connect external antennas.

SN
Serial Number

SPP
(**S**erial **P**ort **P**rofile)
Bluetooth® profile for emulation of a serial port.

SSP
(**S**ecure **S**imple **P**airing)
Based on *Bluetooth*® standard 2.1 or higher, SSP is used for secure authentication.

Status byte
For this I/O module: A specific byte of the cyclic process image input (PII) that provides information on the system status at run-time.

Subsystem
Part of a whole system with which it is connected over defined interfaces.

T

TCP/IP
TCP is a connection-oriented network protocol for the transport layer (Layer 4) of the ISO/OSI model provided with relatively secure transmission mechanisms.

Transmission channel
See "Channel"

U

UserFriendlyName
A name (labeling) for slots chosen by the user that is stored in the local 750-644 I/O module.

W

WAGO-I/O-*CHECK*
WAGO-I/O-*CHECK* software configures local I/O modules (network configuration/process image mapping).

List of Figures

List of Tables

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Phone: +49/5 71/8 87 – 0
Fax: +49/5 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: http://www.wago.com